

Планирование и диспетчеризация процессов и задач

Операционные системы и среды
ВМ-1 3 курс

Потребность в распределении ресурсов

- Очевидно, что на распределение ресурсов влияют конкретные потребности тех задач, которые должны выполняться параллельно. Другими словами, можно столкнуться с ситуациями, когда невозможно эффективно распределять ресурсы с тем, чтобы они не простаивали. Например, пусть всем выполняющимся процессам требуется некоторое устройство с последовательным доступом. Но поскольку, как мы уже знаем, оно не может разделяться между параллельно выполняющимися процессами, то процессы вынуждены будут очень долго ждать своей очереди, то есть недоступность одного ресурса может привести к тому, что длительное время не будут использоваться многие другие ресурсы.

планирование вычислительных процессов.

- Если же мы возьмем такой набор процессов, что они не будут конкурировать между собой за неразделяемые ресурсы при своем параллельном выполнении, то, скорее всего, процессы смогут выполняться быстрее (из-за отсутствия дополнительных ожиданий), да и имеющиеся в системе ресурсы, скорее всего, будут использоваться более эффективно. Таким образом, возникает **задача подбора такого множества процессов, которые при своем выполнении будут как можно реже конфликтовать за имеющиеся в системе ресурсы.** Такая задача называется *планированием вычислительных процессов.*

Диспетчеризация

- Задача планирования процессов возникла очень давно — в первых пакетных операционных системах при планировании пакетов задач, которые должны были выполняться на компьютере и по возможности бесконфликтно и оптимально использовать его ресурсы. В настоящее время актуальность этой задачи стала меньше. На первый план уже очень давно вышли задачи динамического (или краткосрочного) планирования, то есть текущего наиболее эффективного распределения ресурсов, возникающего практически по каждому событию. **Задачи динамического планирования стали называть диспетчеризацией.** *Очевидно, что планирование процессов осуществляется гораздо реже, чем текущее распределение ресурсов между уже выполняющимися задачами. Основное различие между долгосрочным и краткосрочным планировщиками заключается в частоте их запуска, например: краткосрочный планировщик может запускаться каждые 30 или 100 мс, долгосрочный — один раз в несколько минут (или чаще; тут многое зависит от общей длительности решения заданий пользователей).*

Долгосрочный планировщик

- Долгосрочный планировщик решает, какой из процессов, находящихся во входной очереди, в случае освобождения ресурсов памяти должен быть переведен в очередь процессов, готовых к выполнению. Долгосрочный планировщик выбирает процесс из входной очереди с целью создания неоднородной мультипрограммной смеси. Это означает, что в очереди готовых к выполнению процессов должны находиться в разной пропорции как процессы, ориентированные на ввод-вывод, так и процессы, ориентированные преимущественно на активное использование центрального процессора.

Краткосрочный планировщик

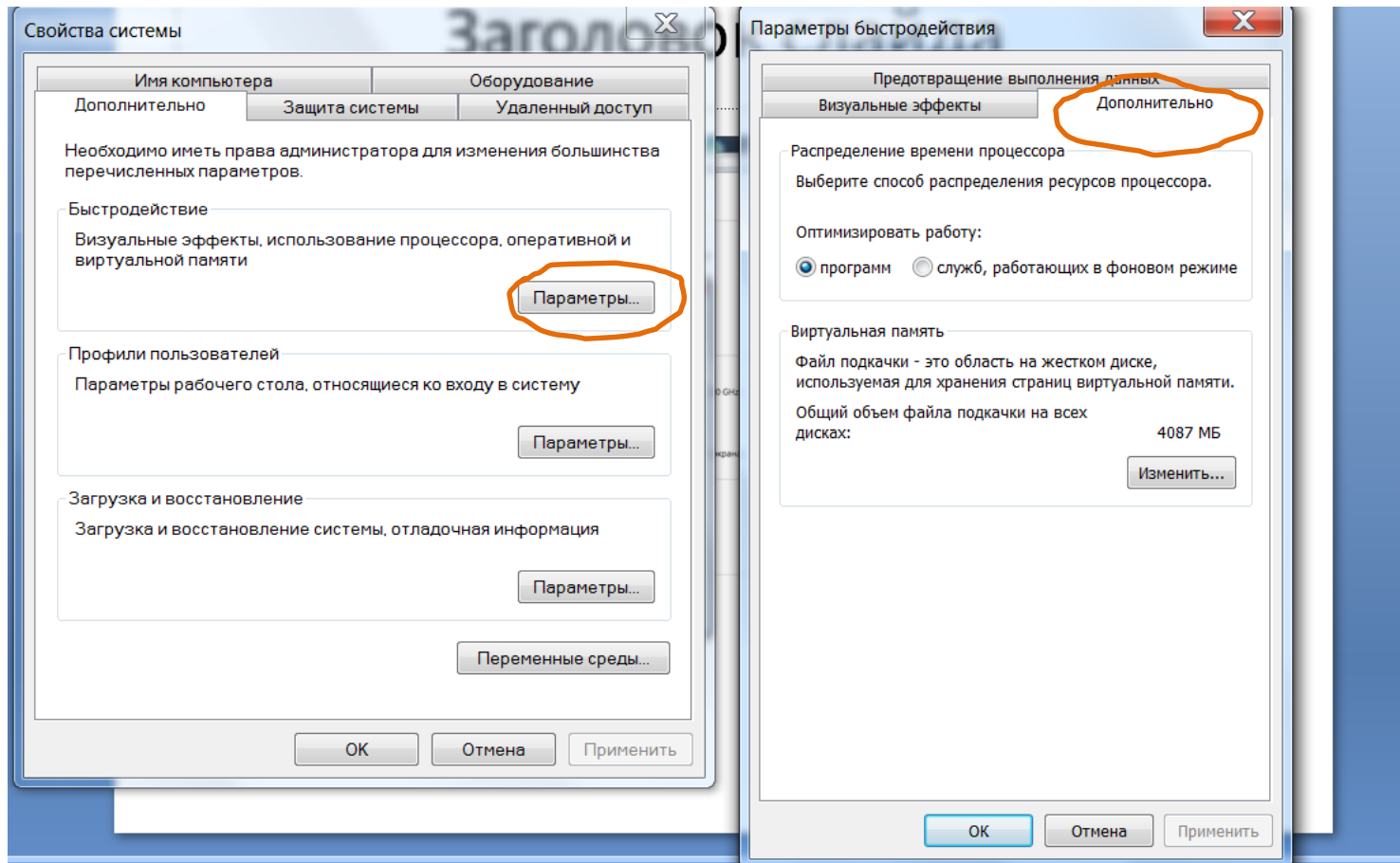
- Краткосрочный планировщик решает, какая из задач, находящихся в очереди готовых к выполнению, должна быть передана на исполнение. В большинстве современных операционных систем, с которыми мы сталкиваемся, долгосрочный планировщик отсутствует.

Стратегии планирования

Стратегия планирования определяет, какие процессы мы планируем на выполнение для того, чтобы достичь поставленной цели. Известно большое количество различных стратегий выбора процесса, которому необходимо предоставить процессор. Среди них, прежде всего, можно выбрать следующие:

- по возможности заканчивать вычисления (вычислительные процессы) в том же самом порядке, в котором они были начаты;
- отдавать предпочтение более коротким вычислительным задачам;
- предоставлять всем пользователям (процессам пользователей) одинаковые услуги, в том числе и одинаковое время ожидания.

Выбор стратегии диспетчеризации задач, в соответствии с которой приоритет на получение процессорного времени будут иметь задачи пользователя, а не фоновые служебные вычисления в Windows 7.



Дисциплины диспетчеризации

- Известно большое количество **дисциплин диспетчеризации**, то есть правил формирования очереди готовых к выполнению задач, в соответствии с которыми формируется эта очередь (список). Иногда их называют **дисциплинами обслуживания**, опуская тот факт, что речь идет о распределении процессорного времени. Одни дисциплины диспетчеризации дают наилучшие результаты для одной стратегии обслуживания, в то время как для другой стратегии они могут быть вовсе неприемлемыми. Известно большое количество дисциплин диспетчеризации.
- Прежде всего, различают два больших класса дисциплин обслуживания: **бесприоритетные и приоритетные**. При неприоритетном обслуживании выбор задач производится в некотором заранее установленном порядке без учета их относительной важности и времени обслуживания. При реализации приоритетных дисциплин обслуживания отдельным задачам предоставляется преимущественное право попасть в состояние исполнения.

дисциплина *FCFS*(*First Come First Served*)

- Самой простой в реализации является дисциплина *FCFS*(*First Come First Served* — первым пришел, первым обслужен), согласно которой задачи обслуживаются «в порядке очереди», то есть в порядке их появления.

SJN(Shortest Job Next — следующим выполняется самое короткое задание)

- *Дисциплина обслуживания SJN(Shortest Job Next — следующим выполняется самое короткое задание) требует, чтобы для каждого задания была известна оценка в потребностях машинного времени. Необходимость сообщать операционной системе характеристики задач с описанием потребностей в ресурсах вычислительной системы привела к тому, что были разработаны соответствующие языковые средства.*

Дисциплина *SRT (Shortest Remaining Time)*

- Дисциплина *SRT (Shortest Remaining Time)* — следующим будет выполняться задание, которому осталось меньше всего выполняться на процессоре.

Дисциплина обслуживания, называемая *карусельной (Round Robin, RR)*

- Дисциплина обслуживания RR предполагает, что каждая задача получает процессорное время порциями или, как говорят, *квантами времени (time slice) q* . После окончания кванта времени q задача снимается с процессора, и он передается следующей задаче. Снятая задача ставится в конец очереди задач, готовых к выполнению.

Карусельная дисциплина диспетчеризации

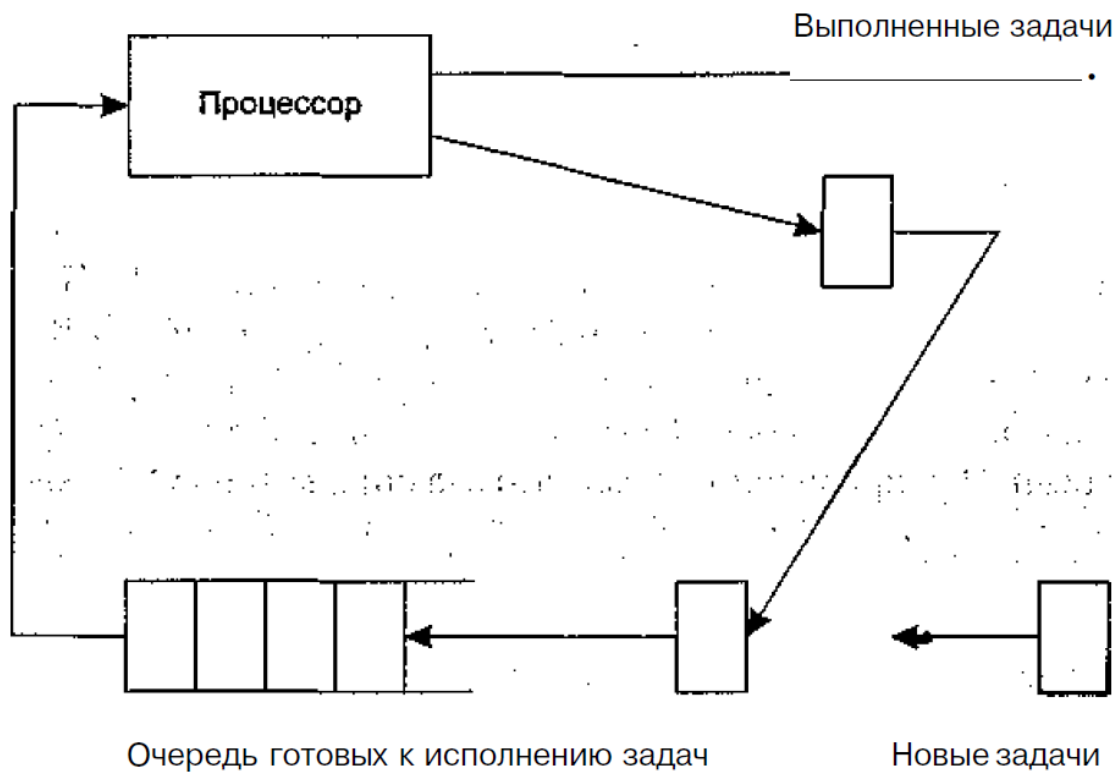


Рис. 2.3. Карусельная дисциплина диспетчеризации

Вытесняющие дисциплины обслуживания

- Есть дисциплины, в которых процессор принудительно может быть отобран у текущей задачи. Такие дисциплины обслуживания называют *вытесняющими*, поскольку *одна задача вытесняется другой*. Другими словами, возможно *принудительное перераспределение процессорного времени между выполняющимися задачами*. Оно осуществляется самой операционной системой, *отбирающей периодически процессор у выполняющейся задачи*.

Не вытесняющие дисциплины обслуживания

- А есть дисциплины диспетчеризации, в которых ничто не может отобрать у задачи процессор, пока она сама его не освободит. Освобождение процессора в этом случае, как правило, связано с тем, что задача попадает в состояние ожидания некоторого события.
- Итак, диспетчеризация без перераспределения процессорного времени, то есть *не вытесняющая (non-preemptive multitasking), или кооперативная, многозадачность (cooperative multitasking)*, — это такой способ диспетчеризации задач, при котором активная задача выполняется до тех пор, пока она сама, что называется «по собственной инициативе», не отдаст управление диспетчеру задач для того, чтобы тот выбрал из очереди другой, готовый к выполнению процесс или поток.

Контрольные вопросы и задачи

- Перечислите и поясните основные функции операционных систем, которые связаны с управлением задачами.
- В чем заключается основное различие между планированием процессов и диспетчеризацией задач?
- Что такое стратегия обслуживания? Перечислите известные вам стратегии обслуживания.
- Какие дисциплины диспетчеризации задач вы знаете? Поясните их основные идеи, перечислите достоинства и недостатки.
- Расскажите, какие дисциплины диспетчеризации следует отнести к вытесняющим, а какие — к не вытесняющим.