

## Множественная очередь

Следующий очевидный шаг: события могут быть разные и какой-то процесс ждет ответа жесткого диска, что медленно, но всё равно довольно быстро по сравнению с человеком, а какой-то ждет человека. И то, что они находятся в одной очереди – это тоже не очень эффективно, поэтому логично разделить эту очередь на несколько типов (по типу события или даже по конкретному событию).

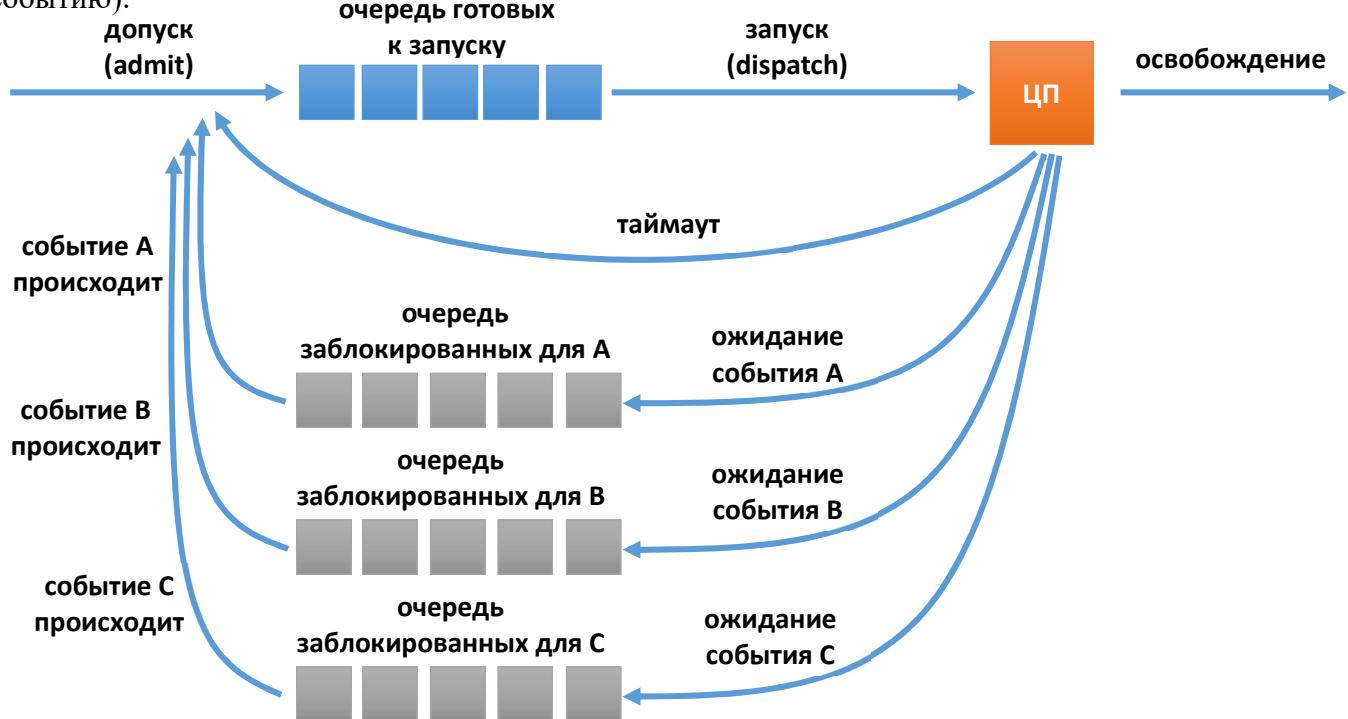


Рисунок 6. Множественная очередь

Можно сказать, «Пусть будет одна очередь, которая ждет какого-то ответа от жесткого диска, пусть будет вторая очередь, которая ждет ответа от сетевого интерфейса, пусть будет третья очередь, которая ждет ответа от человека». Мы добавили сложности, мы добавили больше работы операционной системы, потому что всё, что здесь вы видите – это ответственность операционной системы (она должна содержать эти очереди, она должна «думать» что куда поставить, что куда перенести, она должна следить за этими событиями, сами процессы ничего этого делать не умеют), но это оказывается более эффективным в общем, потому что мы используем ресурсы компьютера, мы ждем меньше, мы работаем больше.

В идеале процессор должен быть сто процентов времени занят, он должен постоянно что-то делать, потому что это будет эффективным путем. Если процессор отдыхает – значит он не используется эффективно. Тогда можно задать вопрос: «Почему тогда процессор не занят всегда на сто процентов, это всё еще плохая операционная система? Почему, когда мы пользуемся компьютером и заходим в какой-нибудь диспетчер задач и видим, что процессор чаще всего работает где-то на 10-15 процентов, если только мы не делаем какую-то очень сложную операцию конвертирования видео или что-нибудь такое?». Процессор не работает на сто процентов, потому что нет у него работы на сто процентов. Процессор – это очень мощный инструмент и когда он работает на сто процентов – это максимальная эффективность, но то как мы используем компьютеры, чаще всего не требует стопроцентного использования процессора постоянно. Нельзя сказать, что это неэффективно в данной ситуации, потому что всё, что нам нужно – всё происходит. Если бы процессор не использовался на сто процентов и при этом какие-то процессы постоянно ждали бы его (например, есть процесс, который стоит в очереди, он хочет сделать что-то и процессор не занят на сто процентов, он может сделать это, но этого не происходит) – вот тогда было бы всё очень неэффективно.