Theory: Synchronous, asynchronous, parallel

11 minutes 0 / 5 problems solved

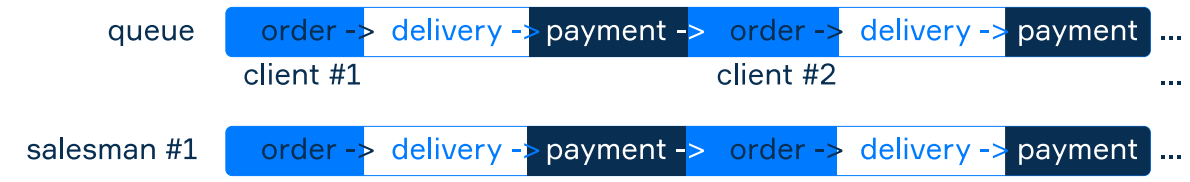
When we are considering some complex process, let's call it **workflow**, various parts of it may run differently. Sometimes actions go one by one, sometimes they go in random order overlapping each other, and sometimes things go simultaneously and in parallel. The workflow can evolve differently. There are three sorts of workflow executions sequence: synchronous, asynchronous, and parallel.

> Many terms related to computer program processing are not just technical ones. They describe a wide variety of real-world phenomena. In some sense, the processes taking place inside a computer are not that different from those in real life. Moreover, on some level of abstraction, they are practically identical. So, let's try to use them and explore their base concepts using a real-life example.

An appropriate example of a complex process is customer service. Let's use it to study some basic types of workflow from the point of view of the sequence of execution.

§1. Synchronous workflow

There are many models to manage customer flows. The simplest approach is one shop with one seller. The seller deals with each client from the beginning to the end of each sale *and* performs all the roles from storekeeper to cashier.

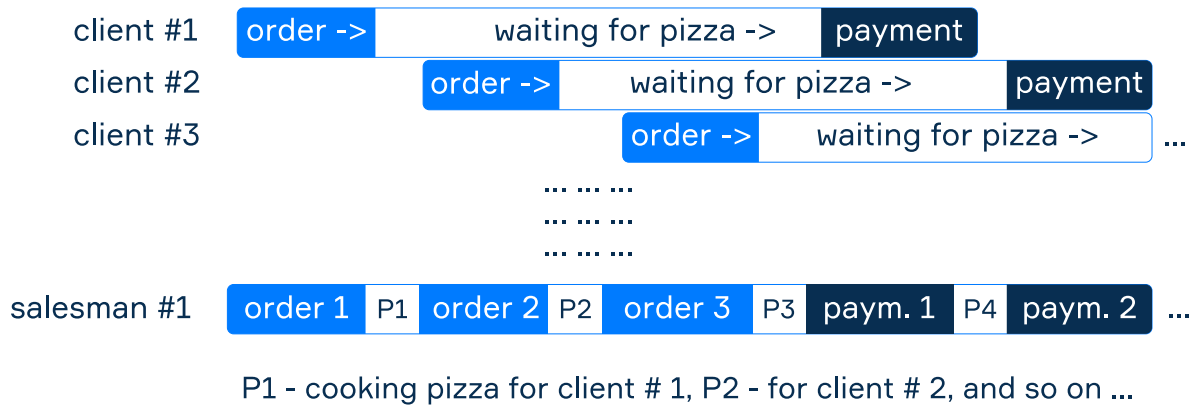


When there are many customers at the same time, this approach is very far from perfect as the seller can deal only with one client per time, while others have to wait in the line. They serve each client separately one by one which means starting to serve the next client only after finishing with the current. We name this type of action a **synchronous** one.

Synchronous workflows are very common. Most of the activities should go synchronously if their goal is to achieve some specific results. The number of examples is enormous. Scenes in a movie plot, car assembling, words in a sentence, cooking, you name it.

§2. Asynchronous workflow

Let's imagine our old shop becomes fancier, this is a pizza shop now. After the first client has ordered their pizza, they need to wait for it to be cooked. At this point, the seller leaves the first customer alone for a while, and now the second one can make their order, then the third and so on.



P1 - cooking pizza for client # 1, P2 - for client # 2, and so on ...

Current topic:

Synchronous, asynchronous, parallel

Topic is required for:

Concurrency and parallelism

Processes and threads

Table of contents:

When the first client's pizza is ready, the seller returns to them to complete the sale. That's how this story will be repeating again and again.

Our old friend seller can serve several customers simultaneously in overlapping periods. We call such behavior **asynchronous**.

Operations of this kind often emerge when there is a need for waiting. Imagine you are reading on an aircraft while flying, or you do the dishes while something is cooking; those pairs of activities are asynchronous.

§3. Parallel processing

As the pizza shop sales are growing, now one worker is not enough for the whole business. So, we should hire several. If each seller has a separate compact oven for preparing exactly one pizza at a time, then we can divide the queue of buyers among the sellers.

Now each of them works independently, and this is a case of **parallel** processing. Each task in parallel processing is running in a continuous period as a whole unit process. The parallel execution is possible only if there is more than one executor. Cashiers in a supermarket are one of the examples of parallel processing in everyday life, as well as highways.

§4. Conclusion

So, there are three types of workflow processing. The first is synchronous, the second is asynchronous, and the third is parallel.

- Synchronous: one task at a time, the next starts when the previous is done.
- Asynchronous: multiple tasks at the same time in overlapping periods, executing by little parts.
- Parallel: multiple or one task split into parts being executed continuously by different executors in parallel.

Report a typo

140 users liked this theory. **5** didn't like it. **What about you?**

Start practicing

Comments (3) Hints (0) Useful links (0) Show discussion