

# 并发简介

## 并发的定义

在操作系统中，是指一个时间段中有几个程序都处于已启动运行到运行完毕之间，且这几个程序都是在同一个处理机上运行，但是任一个时刻点上只有一个程序在处理机上运行。

并发环境下，由于程序的封闭性被打破，出现了新的特点：

1. 程序与计算不再一一对应，一个程序副本可以有多个计算；
2. 并发程序之间有相互制约关系：直接制约体现为一个程序需要另一个程序的计算结果，间接制约体现为多个程序竞争某一资源，如处理机、缓冲区等；
3. 并发程序在执行中是走走停停，断续推进的。

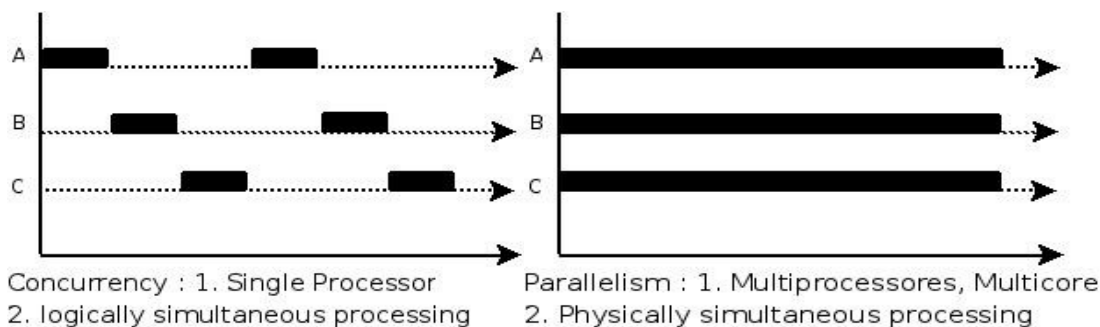
## 和并行的区别与联系

**并发性 (Concurrency)：**指两个或两个以上的事件或活动在**同一时间间隔内**发生。

**并行性 (Parallelism)：**指两个或两个以上事件或活动在**同一时刻**发生。

### • 两者的区别

并发和并行的区别就是一个**处理器同时处理多个任务**和**多个处理器（或者是多核的处理器）同时处理多个不同的任务**。前者是**逻辑上**的同时发生，而后者是**物理上**的同时发生。



### • 两者的联系

并行的事件或活动一定是并发的，但反之并发的活动未必是并行的。

并行性是并发性的特例，而并发性是并行性的扩展。

## 表现形式

### • 多进程并发

当客户有请求时，服务器用一个子进程来处理客户请求，父进程继续等待其它客户的请求。这种方法的优点是当客户有请求时，服务器能及时处理客户，特别是在客户服务器交互系统中。对于一个 TCP 服务器，客户与服务器的连接可能并不马上关闭，可能会等到客户提交某些数据后再关闭，这段时间服务器端的进程会阻塞，所以这时操作系统可能调度其它客户服务进程。

- **多线程并发**

多线程并发是对多进程并发的改进，由于多进程并发服务器在创建进程时要消耗较多的系统资源，所以用线程来取代进程，这样服务处理程序可以较快的创建（据统计，创建线程比创建进程要快 10100 倍，所以又把线程称为“轻量级”进程）。

- **I/O 复用并发**

I/O 复用技术是为了解决进程或线程阻塞到某个 I/O 系统调用而出现的，使进程不阻塞于某个特定的 I/O 系统调用。它也可用于并发服务器的设计，但很多情况下它是与多线程和多进程一起使用。

这种方法的优点是：由于只有一个进程或线程，系统资源消耗较小。缺点是要处理完一个客户后，才能处理其它客户的请求，而且处理客户请求时，不能让它阻塞到其它的 I/O 系统调用上，否则就成了一种迭代服务器。对于某些处理客户请求的时间较短、实时性要求不是很高的网络服务器，I/O 复用可以较好的应用。

I/O 复用能让一个或多个 I/O 条件满足（例如，输入已经准备好被读，或者描述字可以承接更多的输出）时，我们就被通知到。I/O 复用由 select 和 poll 支持，较新的 Posix.1g 也支持 (pselect)。I/O 复用技术的应用面十分广泛，总结一下主要有以下几方面：

- 客户程序需要同时处理交互式的输入以及与服务器之间的网络连接。
- 客户端需要同时对多个网络连接作出反应。
- TCP 服务器需要同时处理处于监听状态和多个连接状态的套接字。
- 服务器需要处理多个网络协议套接字。
- 服务器需要同时处理不同的网络服务和协议。