

私塾在线 《高级软件架构师实战培训 阶段二》

跟着cc学架构系列精品教程

高并发基础

n 何谓高并发

所谓高并发指的是：在同时或极短时间内，有大量的请求到达服务端，每个请求都需要服务端耗费资源进行处理，并做出相应的反馈。

n 从服务端视角看高并发

服务端处理请求需要耗费服务端的资源，比如能同时开启的进程数、能同时运行的线程数、网络连接数、cpu、I/O、内存等等，由于服务端资源是有限的，那么服务端能同时处理的请求也是有限的

高并发问题的本质就是：资源的有限性

n 高并发带来的问题

服务端的处理和响应会越来越慢，甚至会丢弃部分请求不予处理，更严重的会导致服务端崩溃。

n 高并发问题并不是互联网应用独有

n 高并发问题的层面

比如：前端请求、Web服务器、Web应用、数据库等

高并发处理的基本思路-1

n 高并发处理的基本思路

一：从客户端看

- 1: 尽量减少请求数量，比如：依靠客户端自身的缓存或处理能力
 - 2: 尽量减少对服务端资源的不必要耗费，比如：重复使用某些资源，如连接池
- 客户端处理的基本原则就是：能不访问服务端就不要访问

二：从服务端看

- 1: 增加资源供给，比如：
 - (1) 更大的网络带宽
 - (2) 使用更高配置的服务器
 - (3) 使用高性能的Web服务器
 - (4) 使用高性能的数据库
- 2: 请求分流，比如：
 - (1) 使用集群
 - (2) 分布式的系统架构

高并发处理的基本思路-2

3: 应用优化, 比如:

- (1) 使用更高效的编程语言
- (2) 优化处理业务逻辑的算法
- (3) 优化访问数据库的SQL

服务端的处理基本原则是: 分而治之, 并提高单个请求的处理速度

高并发处理的基本手段-1

n 是手段而非具体的方案

下面按照每个层面来谈谈处理的具体手段，注意不是高并发处理的方案，方案是要结合具体的应用，综合分析，选择合适的手段组合起来。

另外还要注意，下面提到的手段都是“包含但不限于”

n 客户端发出请求层面，常见的手段有：

- 1: 尽量利用浏览器的缓存功能，减少访问服务端，比如：js、css、图片等
- 2: 可以考虑使用压缩传输的功能，减少网络流量，也会提高传输速度
- 3: 考虑使用异步请求，分批获取数据

n 前端接收客户端请求层面，常见的手段有：

一：比如Nginx这一层

- 1: 动静分离，部分静态资源可以直接从Nginx返回
- 2: 按请求的不同，分发到不同的后端进行处理，比如：负载均衡、业务拆分访问等
- 3: 前面再加上一层来做多个Nginx的负载均衡，比如：LVS、F5等
- 4: 还可以在更前面使用CDN服务

高并发处理的基本手段-2

二：比如Varnish这一层

- 1：还可以对动态内容进行缓存，尽量减少访问后端服务
- 2：使用页面片断缓存技术，比如ESI (Edge Side Includes)

n Web服务器层面，常见的手段有：

- 1：使用最新的JVM，并进行配置优化
- 2：合理选择服务器的运行模式，比如有些服务器有Client和Server之分
- 3：对Web服务器进行配置优化，比如：调整内存数量、线程数量等
- 4：提供多个能提供相同服务的Web服务器，以实现负载均衡
- 5：仔细规划Web服务器上部署的应用规模
- 6：对Web服务器进行集群
- 7：提供专门的图片、文件、视频等静态资源服务器

高并发处理的基本手段-3

n Web应用层面，常见的手段有：

这个可能全在Web服务器里面，也可能有单独的服务层

- 1: 动态内容静态化
- 2: Java开发优化
- 3: 优化处理业务逻辑的算法
- 4: 合理高效的利用缓存
- 5: 优化访问数据库的Sql，可以考虑利用存储过程等数据库的能力
- 6: 合理使用多线程，加快业务处理
- 7: 部分业务可以考虑内存数据库，或者是进行纯内存处理
- 8: 尽量避免远程调用、大量I/O等耗时的操作
- 9: 合理规划事务等较为耗资源的操作
- 10: 合理使用异步处理
- 11: 对部分业务考虑采用预处理或者预计算的方式，减少实时计算量
- 12: 内部系统间的业务尽量直接调用、直接处理，减少WebService、工作流等

高并发处理的基本手段-4

n 数据库层面，常见的手段有：

- 1: 合理选择数据库的引擎，比如Mysql 的InnoDB与MyISAM引擎
- 2: 进行配置优化
- 3: 可以考虑使用存储过程来处理复杂的数据逻辑
- 4: 数据库集群，进行读写分离
- 5: 合理设计数据库的表结构、索引等
- 6: 分库、分表，降低单库、单表的数据量
- 7: 合理使用NoSql