# 大型网站架构

## 大型网站系统软件的特点

1. **高并发，大访问量**、

Google 日均PV 35亿，日均ip 3亿

2. **高可用**

7\*24 小时不间断服务

3. **海量数据**

4. **用户分布广泛，网络情况复杂**

大型网站为全球提供服务，各地网络情况复杂

5. **需求变更快速，发布频繁**

6. **渐进式发展**

从小网站然后到大网站

## 大型网站架构演化发展历史

1. **初始阶段的网站架构**

LAMP环境，文件系统，应用程序，数据库在一台服务器上如图:

文件

应用

数据库

应用服务器

1. **应用服务和数据服务分离**
2. 访问用户越来越多，网站性能越来越差
3. 数据越来越多，存储空间不足

数据库服务器，硬盘速度

文件服务器,硬盘大小

应用服务器:偏重CPU

1. **使用缓存改善网站性能**
2. 本地缓存: 更快，但是大小有限制
3. 分布式缓存服务器

把经常使用的数据使用分布式缓存服务器缓存到内存中 ，从而减少数据库的访问压力。

例: memcached

1. **使用应用服务器集群改善网站的并发处理能力**

通过负载均衡调度服务器，将来自用户浏览器的请求分发到应用服务器集群中的任何一台服务器上，如果有更多用户，就在集群中加入更多的应用服务器

1. **数据库读写分离**

应用程序在写数据的时候，访问主数据库，主数据库通过主从复制将数据更新同步到从数据库，这样应用服务器读数据的时候，就可以通过从数据库获得数据

1. **使用反向代理和CDN加速网站响应**
2. 用户分布广泛，网络环境复杂，速度差别极大
3. CDN和反向代理的基本原理都是缓存，区别.CDN部署在网络提供商的机房，使用户请求网站时，可以从距离自己网络最近的网络提供商获取数据。反向代理则部署在网站的中心级房，当用户请求到达中心机房后，首先访问的服务器是反向代理服务器，如果反向代理服务器中缓存这用户请求的资源，就将其直接返回给用户。
4. **使用分布式文件系统和分布式数据库系统**
5. **使用NoSQL和搜索引擎**

业务越来越复杂，对数据存储和检索的需求越来越复杂，需要采用非关系型数据库如NoSQL和非数据库查询技术如搜索引擎

1. **业务拆分**

将整个网站业务分成不同的产品线，分归不同的业务团队负责。具体到技术上，将一个网站分成不同的应用，每个应用独立部署维护，应用之间通过超链接建立联系，也可以通过消息队列进行数据分发，最多的还是通过访问同一数据存储系统来构成一个关联的完整系统

1. **分布式服务**

把相同共用的业务提取出来，独立部署

分布式数据库服务器

分布式缓存服务器

分布式文件服务器

B应用服务器

分布式服务器

CDN服务器

NoSQL服务器

搜索引擎服务器

A应用服务器务器

消息队列服务器服务器

负载均衡调度服务器

反向代理服务器