# 大型网站架构模式

## 网站架构模式

目标: 高性能，高可用，易伸缩，可扩展，安全

## 分层

分层架构

1. 应用层: 负责具体业务和视图展示，如网站首页及搜索输入和结果展示
2. 服务层: 为应用层提供服务支持，如用户管理服务，购物车服务
3. 数据层: 提供数据访问服务，如数据库，缓存，文件，搜索引擎等

注：

1. 合理规划层次边界和接口

2. 严格遵守分层架构的约束，禁止跨层调用，逆向调用

二级分层

应用层:

1. 视图层
2. 业务逻辑层

服务层:

1. 数据接口层
2. 逻辑处理层

## 分割

将不同的功能和服务分割开来，包装成高内聚低耦合的模块单元。将不同的业务进行分割，由独立团队负责，部署在不同的服务器上。

## 分布式

目的: 分层和分割后的模块便于分布式部署，将不同的模块部署在不同的服务器上，通过远程调用协同工作。

缺点:

1. 通过网络链接协同工作，造成性能的影响
2. 服务器越多宕机的可能性越大，使网站可用性降低
3. 分布式环境保持数据一致性非常困难
4. 分布式网站错综复杂，不易维护

分布式方案

1. 分布式应用和服务：将分层和分割后的应用和服务分布式部署。改善网站性能和并发，加快开发和发布速度，较少数据库链接资源，使不同应用复用，共用服务
2. 分布式静态资源：JS,CSS,图片等资源独立分布式部署，并采用独立的域名，动静分离。减轻应用服务器的负载压力
3. 分布式数据和存储
4. 分布式计算

## 集群

多台服务器部署相同的应用构成一个集群，通过负载均衡设备共同对外提供服务。

避免宕机造成的故障

## 缓存

1. CDN: 内容分发网络，用户的网络请求先到达他的网络服务商，这里缓存网站的一些静态资源，就近以最快的速度返回给用户

2 反向代理：数据前端架构的一部分，部署在网站的前端，当用户请求到达网站的数据中心时，最先访问到的就是反向代理服务器，这里缓存网站的静态资源

1. 本地缓存：在应用服务器本地缓存热点数据，应用程序可以在本机内存中直接访问数据，而无需访问数据库
2. 分布式缓存：