

## 第四周总结

---

首先是课程笔记：

### 4.1 系统架构：系统技术挑战与方案

1. 挑战：高并发高可用；海量数据；用户分布广泛，网络情况复杂；安全环境恶劣；需求快速变更，发布频繁；渐进式发展 -> 最重要的还是高并发带来的挑战

2. 解决方案：垂直伸缩；水平伸缩

### 4.2 分布式架构演化

系统的演化：单服务器 -> 数据，文件，应用分离 -> 使用缓存 -> 应用程序集群 -> 数据库读写分离 -> CDN，反向代理，负载均衡 -> 分布式文件系统和数据库 -> NoSql和搜索引擎 -> 业务拆分 -> 微服务和中台化 -> 大数据与智能化

### 4.3 架构模式与要素

1. 定义：互联网架构模式就是试图去描述那些为解决互联网系统高性能、高可用、易扩展、可伸缩、安全等目标，被很多互联网应用重复使用的一些解决方案，这些解决方案是互联网软件的重要组成部分

2. 架构模式：

2.1 分层：不同的层级部署在不同的服务器集群，然后通过上层对下层依赖调用的方式组成一个完整的系统

2.2 分割：纵向层面把服务分割成一个个相对独立的高内聚低耦合的模块单元，分别进行实现和部署

2.3 集群：服务在进行分层分割后，还要把用户访问比较多的模块集群化，比如网站首页。通过负载均衡对外提供服务。

2.4 缓存：CDN，反向代理，本地缓存，远程缓存

2.5 异步：没有直接的调用耦合，通过消息队列共享数据，从而完成服务的逻辑。这可以提高系统的可用性，加快网站的响应速度，消除并发访问高峰。

2.6 冗余：针对服务器的宕机问题，将服务本身进行冗余互备，将数据进行冗余备份。

2.7 自动化：主要集中在运维方面，在无人值守的时候依然可以正常运行。

2.8 安全：登录认证，加密，验证码，防止XSS攻击和SQL注入，风控，垃圾信息敏感信息过滤等

3. 架构指标：

3.1 高性能：互联网架构设计的关键点

3.2 高可用：针对服务器宕机，服务依然可用。这是一个比较硬的指标。

3.3 可伸缩性：是否可以用多台服务器构成一个集群，可以方便地加减服务器。

NoSql伸缩性远远好于分布式数据库

3.4 可扩展：功能性要求，需求迭代更为容易。

3.5 安全

### 4. 互联网技术一览

4.1 前端架构：App及Web开发技术；浏览器及HTTP优化技术；CDN；动静分离；图片服务；反向代理；DNS

4.2 网关及应用层架构：网关架构；负载均衡；动态页面静态化；业务拆分

4.3 服务层架构：微服务框架；分布式消息队列；分布式缓存；分布式一致性（锁）服务

4.4 存储层架构：分布式文件；分布式关系数据库；NoSql数据库

4.5 后台架构：大数据平台；搜索引擎；推荐引擎；数据仓库

4.6 运维与安全：数据采集与展示；数据监控与报警；攻击与防护；数据加密与解密

#### 4.4 维基百科技术架构

维基百科是全球第六大流量互联网，仅次于百度，只有十几个人开发和维护，用的都是捐献来的低质量服务器

---

#### 个人感想：

大型网站架构的方法和模式有很多，并不是每一种好的方法和模式都适合任何公司，技术是为业务服务的，根据公司发展阶段的不同，对于技术的投入是完全不一样的。做架构需要能够根据公司的需要和发展，合适的进行决策，以最小的代价满足公司发展的需求。

另外，根据技术架构的设计，还可以把各个模块翻译成岗位，这样，技术部的组织架构大致上也可以确定了。