**架构是业务数据与业务逻辑之间的折衷关系**

分层：根据模块相似性，横向划分层次。

分割：根据子模块差异性，纵向切分。

集群：相同的模块部署在不同的机器上，构成一个集群，通过负载均衡向外服务。(冗余)

分布式：不同的模块部署在不同的机器上，远程调用协同。(分工合作)

Performance

1. 缓存(cache)

场景：读多写少、很少变化的数据，80%业务集中在20%数据上，空间换时间。

应用：本地缓存、CDN

1. IO多路复用(multiplexing)

场景：传输任务数 > 信道数

应用：线程池、HTTP/TCP链接

1. 压缩(compression)

场景：压缩开销 < 传输开销

应用：git

High Availability

1. 无状态化服务冗余，状态化数据备份，针对容错性问题

Scalability

1. 向集群中增加机器，针对业务量增长的问题

Business Extensibility

1. 业务解耦划分，针对新增业务的问题

Security

-----------------------------------------------------------------------------------

Micro Service

微服务开发流程：

1. 设计：接口、模型

2. 编码：代码版本库

3. 流水线构建：编译打包、部署环境、用例验证

4. 发布版本

微服务集成：

服务容器化、路由转发、分层管理、容错机制