# 网络框架的发展

1. 单一应用框架

当网站流量很小时，只需一个应用，将所有功能都部署在一起，以减少部署节点和成本，此时，用于简化增删改查工作量的数据访问框架（ORM）最为关键。

### **垂直应用架构**

当访问量逐渐增大，单一应用增加机器带来的加速度越来越小，提升效率的方法之一是将应用拆成互不相干的几个应用，以提升效率。此时，用于加速前端开发的Web框架（MVC）是关键。

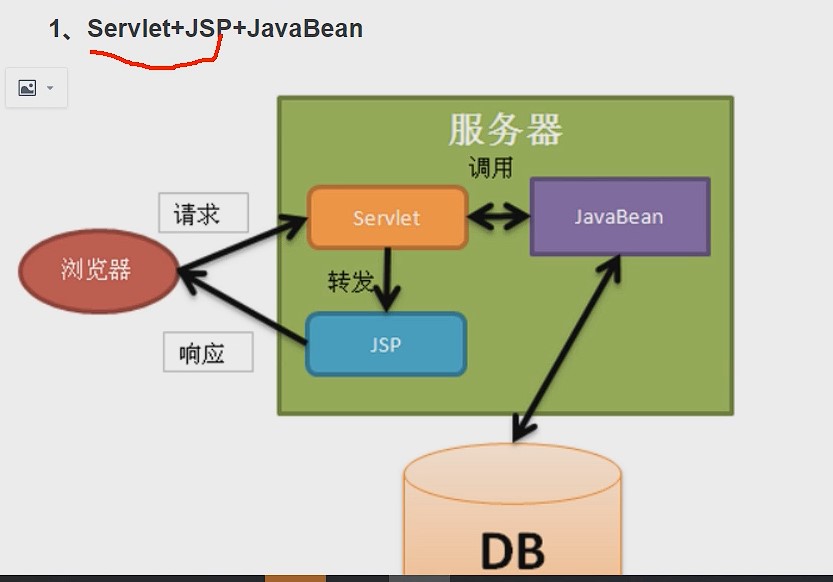
1. **分布式服务架构**

当垂直应用越来越多，应用之间交互不可避免，将核心业务抽取出来，作为独立的服务。逐渐形成稳定的服务中心，使传统应用能更快速响应多变的市场需求。此时，用于提高业务复用及整合的分布式服务框架（PRC）是关键。

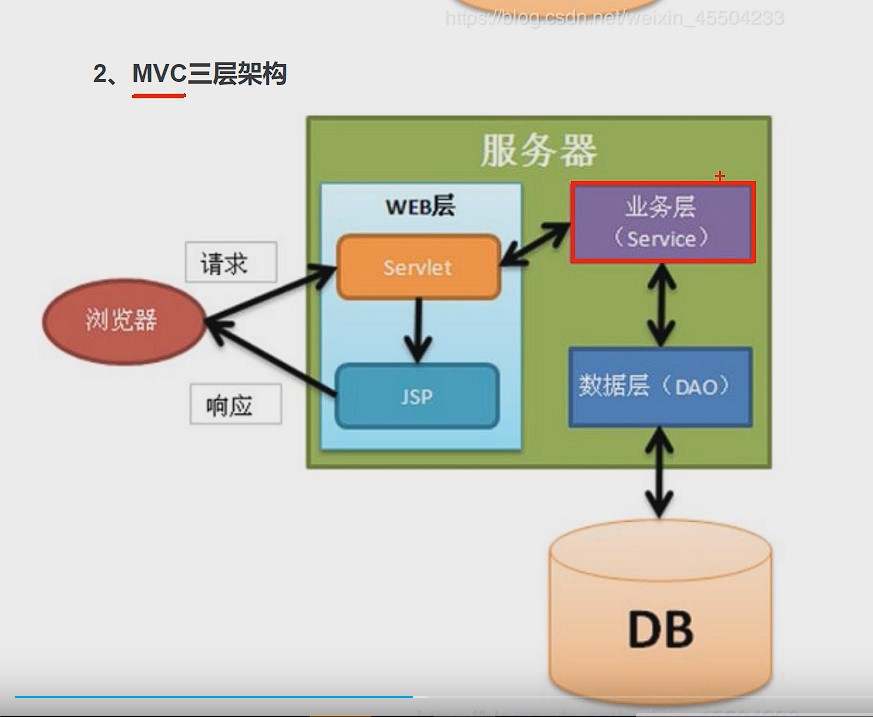
1. **流动计算架构**

当服务越来越多，容量的评估，小服务资源的浪费问题逐渐显现，此时需增加一个调度中心基于访问压力实时管理集群容量，提高集群利用率。此时，用于提高机器利用率的资源调度和治理中心（SOA）最关键。

1. **Java主流架构技术演变之路**
2. Servlet+JSP+JavaBean



1. MVC三层架构



（3）使用EJB进行应用的开发，但是EJB是重量级框架（在使用的时候，过多的接口和依赖，侵入性强），在使用上比较麻烦

（4）Struts1/Struts2+Hibernate+Spring(俗称SSH慢慢会被淘汰但是如今仍有公司再用，像银行，金融机构等)

（5）SpringMVC+Mybatics+Spring

（6）SpringBoot开发，约定大于配置

6.Spring的介绍

官网地址：<http://spring>.io/projects/spring-framework#overview

Spring(chinese introduction)：<http://waylau.gitbooks.io/spring-framework-4-reference/>

源码地址：<http://github.com/spring-projects/spring-framework>

Spring是一个轻量级的java框架目的是解决企业级应用开发的业务逻辑层和其他各层的耦合问题。他是一个分层的javaSE、JavaEE full-stack(一站式)轻量开发框架，为开发java应用程序提供全面的基础架构支持。Spring负责基础架构，因此java开发者可以更专注的进行应用开发。

Spring最根本的使命是解决企业级应用开发的复杂性，及简化java开发。

Spring可以做很多事情，他为企业级提供了丰富的功能，但是这些功能的底层都依赖于他的两个核心特性，也就是依赖注入（dependency injection,DI）和面向切面编程（aspect-oriented programming,AOP）.

简略核心解释:

1. Spring 是轻量级的开源JavaEE框架
2. Spring可以解决1企业应用开发的复杂性
3. Spring有两个核心部分：IOC和AOP
4. IOC：控制反转，把创建对象过程交给Spring进行管理
5. AOP：面向切面，不修改源代码进行功能增强
6. Spring特点：
7. 方便解耦，简化开发
8. AOP编程文件
9. 方便程序测试
10. 方便和其他框架进行整合
11. 降低JavaEE API的使用难度
12. Java源码是经典学习范例
13. 方便进行事务的管理