Web是一种基于超文本和HTTP的分布式图形信息系统，为浏览者提供了图形化的、易于访问的只管界面。

Web资源分为**静态Web资源**（如html页面）：人们浏览到的页面内容是不会改变的。**动态Web资源**：指Web页面中人们浏览到的内容是由程序产生的，不同时间点看到的内容可能会有不同。静态Web资源开发技术有：html、css等。动态Web资源开发技术有：JavaScript、JSP/Servlet、ASP、PHP等。

Java Web是用Java技术来解决Web互联网领域的技术栈。在Java中，动态Web资源开发技术统称为Java Web。

### Java Web开发发展

**静态页面**

最早的网站都是静态的，这里的静态指的是网页内容是提前编辑好的，写死的不会改变的。当你输入一个网址，会得到一个固定的内容。

于是问题就是，访问的资源必须是实现写好的，否则就访问不到，而动态展示也不能实现。

**CGI**

为了实现动态展示，出现了CGI。CGI是一个可执行的程序或脚本（比如python、C、shell）。这时候访问一个页面，该页面会向服务器请求CGI程序，服务器启动CGI进程处理数据并返回结果，最后交由浏览器显示。

于是新的问题就是，每个请求都会启动一个CGI，启动慢并且资源消耗太大。

**Servlet**

Java Servlet是运行在服务器上的程序，作为来自Web浏览器或其他HTTP客户端的请求和HTTP服务器上的数据库或应用程序之间的中间层。Servlet可以收集来自网页表单的用户输入，呈现来自数据库或者其他来源的记录，还可以动态创建网页。

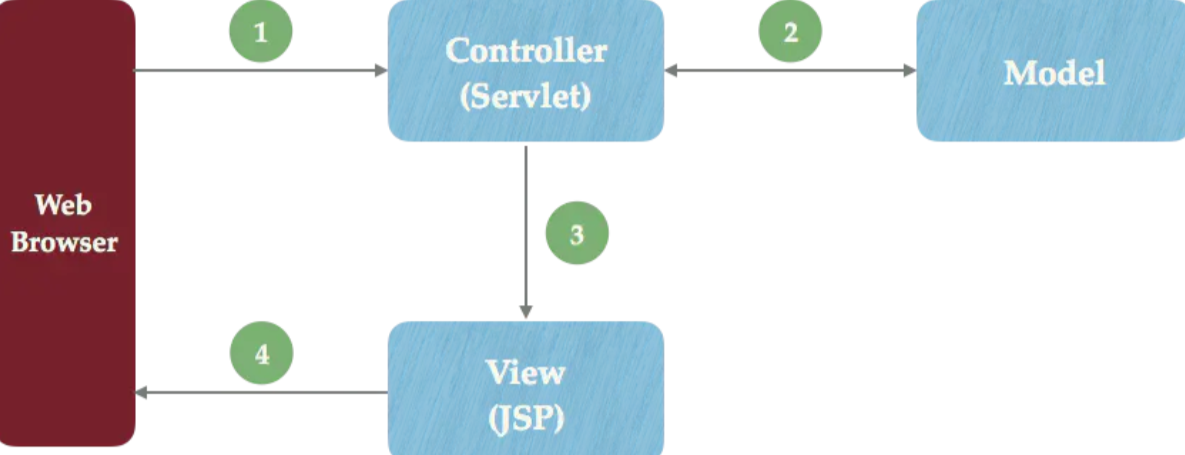
性能比起CGI好了很多，能做的事情也很多。可是将html和java代码混杂着写，对于前端人员来说，似乎不是太友好。

**JSP**

Servlet是在Java代码中写html代码，而JSP是在html代码中写Java。虽然这样方便了前端程序员，但是后端程序员并不好过。因此使用JSP开发的效率依然不高。

**MVC**

后来发现，Servlet适合逻辑处理（因为主要是Java代码），而JSP非常适合页面展示（主要是HTML代码），于是在结合了Servlet和JSP各自的优缺点之后，诞生了架构射击模式：MVC

 MVC模式是软件工程中的一种软件架构模式，把软件系统分为三个基础部分：模型（Model：业务功能编写、数据库射击以及数据存取操作）、视图（View：负责界面显示）、控制器（Controller：负责转发请求，对请求进行处理）

**框架**

有了Servlet、JSP和MVC之后，已经为网页的开发提供了很大便利，但是还不够，为了能让开发变得更加简单，后续诞生了许多框架。

### Java Web主流开发框架

框架，是一个提供可重用的公共结构半成品，它为构建新的应用程序提供了极大便利。“框架的强大之处不是源自他能让你做什么，而是它不能让你做什么”。框架能将混乱的内容变得结构化，保证了程序结构风格的统一。

Java的Web框架虽然各不相同，但基本也都是遵循特定的路数的：使用Servlet或者Filter拦截请求，使用MVC的思想设计架构，使用约定，XML或 Annotation实现配置，运用Java面向对象的特点，面向对象实现请求和响应的流程，支持Jsp，Freemarker，Velocity等视图。

**Spring**

Spring框架是一个轻量级的框架，是由于软件开发的复杂性而创建的，渗透了Java EE技术的方方面面，是一个开源框架。

* 目的：解决企业应用开发的复杂性
* 目标：Java EE技术更容易使用，并促进良好编程习惯的养成
* 功能：使用基本的JavaBean代替EJB，并提供更多的企业应用功能
* 范围：任何Java应用

Spring框架的用途不仅限于服务器端的开发，从简单性、可测试性和松耦合性角度而言，绝大部分Java应用都可以从Spring框架中受益。

Spring框架是一个轻量级控制反转（loC：在spring中，对象的属性是由对象自己创建的，就是正向流程；如果属性不是对象创建，而是由spring来自动进行装配，就是控制反转）和面向切面（AOP：以功能进行划分，对服务顺序执行流程中的不同位置进行横切，完成各服务共同需要实现的功能）的容器框架，主要为依赖注入容器和AOP实现存在，还提供了声明式事务、对DAO层的支持等简化开发的功能。

Spring 框架可以很方便地与 Spring MVC、Struts 2、MyBatis、Hibernate 等框架集成，其中大名鼎鼎的 SSM 集成框架指的就是基于 Spring MVC + Spring + MyBatis 的技术框架，使用这个集成框架能使应用程序更加健壮、稳固、轻巧和优雅，这也是当前流行的 Java Web 技术框架。

**Spring MVC**

Spring MVC 框架属于 SpringFrameWork 的后续产品，已经融合在 Spring Web Flow 中，是结构清晰的 MVC Model2 的实现。

Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块，并且拥有高度的可配置性，支持多种视图技术。它还可以进行定制化开发，使用相当灵活。

此外，Spring 框架整合 Spring MVC 框架是无缝集成，这是一个高性能的架构模式，已越来越广泛地应用于互联网应用的开发中。当使用 Spring 框架进行 Web 开发时，可以选择 Spring MVC 框架或集成其他 MVC 的开发框架

优点：

* + 对Spring MVC覆盖绑定（overriding binding）、验证（validation）等提供生命周期管理
  + 与许多表示层技术/框架无缝集成：JSP/JSTL、Tiles、Velocity、FreeMarker、Excel、XSL、PDF 等
  + 便于测试——归功于IoC

缺点：

* + 大量的XML配置文件
  + 太过灵活——没有公共的父控制器
  + 没有内置的Ajax支持

**Hibernate 框架**

Hibernate 框架不仅是一个优秀的持久化框架，也是一个开放源代码的对象关系映射框架。它对 JDBC 进行了轻量级的对象封装，将 POJO 与数据库表建立映射关系，形成一个全自动的 ORM 框架。

Hibernate 框架可以自动生成 SQL 语句，且自动执行，使 Java 程序员可以随心所欲地使用对象编程思维来操纵数据库。

Hibernate 框架还可以应用在任何使用 JDBC 的场合：

* 可以在 Java 的客户端程序使用；
* 也可以在 Servlet/JSP 的 Web 应用中使用；
* 最具革命意义的是，Hibernate 框架可以在应用 EJB 的 Jave EE 架构中取代 CMP，以完成数据持久化的重任。

Hibernate 框架已经成为当前主流的数据库持久化框架，并被广泛应用。

**MyBatis 框架**

MyBatis 框架是一个优秀的数据持久层框架，可在实体类和 SQL 语句之间建立映射关系，是一种半自动化的 ORM 实现。Mybatis 的封装性要低于 Hibernate 框架，且性能优异、简单易学，因此应用较为广泛。

持久层： 可以将业务数据存储到磁盘，具备长期存储能力，只要磁盘不损坏，在断电或者其他情况下，重新开启系统仍然可以读取到这些数据。

优点： 可以使用巨大的磁盘空间存储相当量的数据，并且很廉价

缺点：慢（相对于内存而言）

**Struts 2 框架**

Struts 2 框架以 WebWork 的优秀设计思想为核心，吸收 Struts 框架的部分优点，提供了一个更加简洁的基于 MVC 设计模式实现的 Web 应用程序框架，它本质上相当于一个 Servlet。

在 MVC 设计模式中，Struts 2 框架作为控制器（Controller）来建立模型与视图的数据交互。

Struts 2 框架是 Struts 的下一代产品，是在 Struts 1 和 WebWork 技术的基础上进行合并的创新。它采用拦截器的机制来处理用户的请求，可使业务逻辑控制器与 Servlet API 完全脱离开，所以也可以理解是 WebWork 的更新产品。

Struts 2 框架充分利用了其他 MVC 框架的经验和教训，使整个框架更加清晰和灵活。

优点：

* + 架构简单——易于扩展
  + 标记库很容易利用FreeMarker或者Velocity来定制
  + 基于控制器或
  + 者基于页面的导航

缺点：

* + 文档组织得很差
  + 对新特征过分关注

### 发展现状和发展趋势

随着时间的推演，后面又出现企业级JavaBean EJB等大型框架，再到我们熟悉的Spring、SpringBoot。而Spring诞生之初，主要目的是用来替代更加重量级的企业级技术 尤其是EJB，相对于EJB来说Spring提供了更加轻量级和简单的编程模型。它增强了简单老式的Java对象POJO的功能，使其具备了之前只有EJB和其他企业级Java规范才有的功能。

就Spring和EJB的比较以及Spring对于EJB的提升优点等不作过多解释，但从前面的种种发展历程来看，我们现在使用的Spring等都相对来说简单、方便了很多，框架的产生也对Javaweb开发影响极深，也是发展的必然趋势。

Java Web的发展应该还是很快的，今天一个技术，明天一个框架。我们应该学会每个框架其中的原理，以及他存在的原因。