Atitit mvc的原理与概论attilax总结

[1.1. 客户端mvc vs服务端mvc 1](#_Toc8784)

[1.2. Mvc知识点 1](#_Toc14442)

[1.3. Mvp mvvm 1](#_Toc26601)

[1.4. Atitit.web三大编程模型 Web Page Web Forms 和 MVC.doc 2](#_Toc28024)

[1.5. Mvc与组件化 2](#_Toc9300)

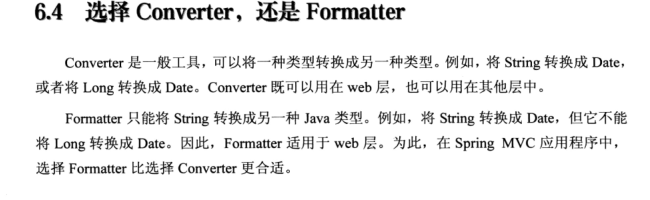
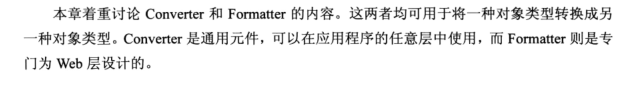
[1.6. 常用mvc框架，angular struts spring mvc 2](#_Toc24087)

[1.6.1. Spring MVC学习指南 ([美]戴克 2](#_Toc14073)

## 客户端mvc vs服务端mvc

## Mvc知识点

|  |  |  |  |
| --- | --- | --- | --- |
| mvc |  |  | 服务端控件体系 |
| mvc |  |  | mvc基础 |
| mvc |  |  | angular |
| mvc |  |  | mvvm |
| mvc |  |  | 双向绑定 |
| mvc |  |  | [拦截器 2](" \l "_Toc18559) |
| mvc |  |  | 过滤器filter |
| mvc |  |  | 数据格式化 |
| mvc |  |  | html指令 |
| mvc |  |  | 输入验证 |
| mvc |  |  | include |
| mvc |  |  | ajax |
| mvc |  |  | ognl |
| mvc |  |  | Web Page Web Forms |
| mvc |  |  | mvp |
| mvc |  |  | Converter和Formatter 91 |
| mvc |  |  | JSR 303 Validator |
| mvc |  |  | 表达式语言（EL） |
| mvc |  |  | 国际化 |
| mvc |  |  | JSTL |
| mvc |  |  | 数据绑定 |



## Mvp mvvm

## Atitit.web三大编程模型 Web Page Web Forms 和 MVC.doc

## Mvc与组件化

MVC横切割了开发过程中的代码，从服务器到浏览器层层分离，层次之间耦合度很低，因为它是顺着底层的开发脉络进行封装，所以有利于开发者对整个程序过程流转的理解。但是MVC有一个非常大的缺点，这个缺点是和整个软件发展思路相背离的，那就是它无法封装、无法封装所以无法被重用。有谁看到过mvc下面的组件？有的只是一个个现成的案例，然后拿来修改。因为一个组件肯定牵涉到控制和显示，但是mvc的开发这两个层次是分离的。

WebForm是微软前面一套web开发的机制。它纵向切割了代码，控制和显示是封装在一起的。它从开发者思维逻辑上而不是实际情况上对代码进行封装，开发webform容易上手的原因也就在此了.. WebForm一出现后，随之而来的是大量的组件诞生，这是mvc模式下看不到的。而组件化，构件化是提升开发效率的极大因素

webform生产关系是比mvc先进的

## 常用mvc框架，angular struts spring mvc

Atitit.struts2体系结构大总结 - attilax的专栏 - 博客频道 - CSDN.NET.html

### [Spring MVC学习指南 ([美]戴克](https://www.baidu.com/link?url=Fm-P5kJxKM-6r3hkxI6CFa6vixVd6X1_dURYI3bdFviFsygSaX0tp2KdvsdBJf-G&wd=&eqid=a1431818000006a60000000658333c9a" \t "https://www.baidu.com/_blank)