



马哥教育
www.magedu.com



2016全新Linux运维实战+python实战班震撼上线！
先就业，后付款！年薪24W！学徒班还有少量名额！

首页
最新文章
开发
IT技术
职场
业界
极客
创业
访谈
在国外

- 导航条 - ▾

伯乐在线 > 首页 > 所有文章 > Web前端 > 前后端分离的思考与实践（二）

前后端分离的思考与实践（二）

2014/04/18 · [Web前端](#), [开发](#) · [1 评论](#) · [Node.js](#), [Web开发](#)

分享到：

18

慕课网技术沙龙之前端专场
PS入门教程——新手过招
ps入门教程 II -进阶篇-图层、矢量、路径
Java实现权限管理（下）

原文出处：[淘宝UED - Herman](#)

基于前后端分离的模版探索

前言

在做前后端分离时，第一个关注到的问题就是 渲染，也就是 View 这个层面的工作。

在传统的开发模式中，浏览器端与服务器端是由不同的前后端两个团队开发，但是模版却又在这两者中间的模糊地带。因此模版上面总不可避免的越来越多复杂逻辑，最终难以维护。

而我们选择了NodeJS，作为一个前后端的中间层。试图藉由NodeJS，来梳理 View 层面的工作。

使得前后端分工更明确，让专案更好维护，达成更好的用户体验。

本文

渲染这块工作，对于前端开发者的日常工作来说，佔了非常大的比例，也是最容易与后端开发纠结不清的地方。

回首过去前端技术发展的这几年，View 这个层面的工作，经过了许多次的变革，像是：

1. Form Submit 全页刷新 => AJAX局部刷新
2. 服务端渲染 + MVC => 客户端渲染 + MVC

3. 传统换页跳转 => 单页面应用

可以观察到在这几年，大家都倾向将 **渲染** 这件事，从服务器端移向了浏览器端。

而服务器端则专注于 **服务化**，提供数据接口。

浏览器端渲染的好处

浏览器端渲染的好处，我们都很清楚，像是

- a. 摆脱业务逻辑与呈现逻辑在Java模版引擎中的耦合与混乱。
- b. 针对多终端应用，更容易以接口化的形式。在浏览器端搭配不同的模版，呈现不同的应用。
- c. 页面呈现本来就不仅是html，在前端的渲染可以更轻易的以组件化形式 (html + js + css) 提供功能，使得前端组件不需依赖于服务端产生的html结构。
- d. 脱离对于后端开发、发布流程的依赖。
- e. 方便联调。

浏览器端渲染造成的坏处

但是在享受好处的同时，我们同样的也面临了 **浏览器端渲染** 所带来的坏处，像是：

- a. 模版分离在不同的库。有的模版放在服务端 (JAVA)，而有的放在浏览器端 (JS)。前后端模版语言不相通。
- b. 需要等待所有模版与组件在浏览器端载入完成后才能开始渲染，无法即开即看。
- c. 首次进入会有白屏等待渲染的时间，不利于用户体验
- d. 开发单页面应用时，前端Route与服务器端Route不匹配，处理起来很麻烦。
- e. 重要内容都在前端组装，不利于SEO

反思前后端的定义

其实回头想想，在我们把渲染的工作从 **服务端(Java)** 抽出来到 **浏览器端(JS)** 的时候，我们的目的只是明确的前后端职责划分，并不是非浏览器渲染不可。

只是因为传统的开发模式中，出了服务器就到了浏览器，所以前端的工作内容只能被限制在浏览器端。

也因此很多人认定了 后端 = 服务端 前端 = 浏览器端，就像下面这张图。

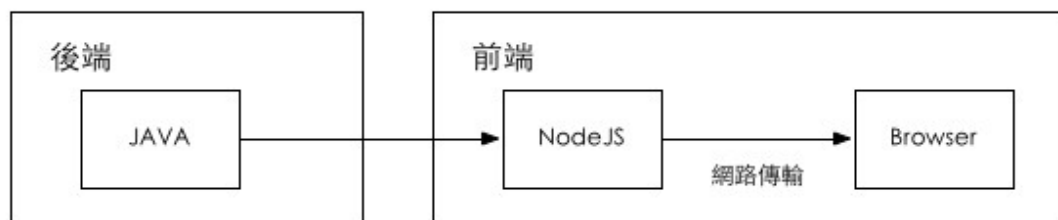


而在淘宝UED目前进行的 **中途岛Midway** 项目中，藉由在 **JAVA - Browser**中间搭建一个NodeJS中间层，试图把这个前后端的分割线，重新针对 **工作职责** 去区分，而分针对硬体环境去区分（服务器 & 浏览器）。

因此我们有机会做到模版与路由的共享，也是一个前后端分工中最理想的状态。

淘宝中途岛 Midway

在中途岛项目中，我们把前后端分界的那条线，从浏览器端移回到了服务器端。



藉由一个由前端 轻松掌控 且 与浏览器共通 的Nodejs层，可以更清晰的完成了前后端分离。

也可以让前端开发针对不同的情况，自行决定 最适当的解决方案。而不是所有事情 都在浏览器端来处理。

职责划分

中途岛并不是前端试图抢后端饭碗的项目，目的只是把 模版 这个模糊地带切割清楚，取得更明确的职责划分。

- 后端 (JAVA)，专注于
 1. 服务层
 2. 数据格式、数据稳定
 3. 业务逻辑
- 前端，专注于
 1. UI层
 2. 控只逻辑、渲染逻辑
 3. 交互、用户体验

而不再拘泥于服务端或浏览器端的差异。

模版共享

在传统的开发模式中，浏览器端与服务器端是由不同的前后端两个团队开发，但是模版却又在这两者中间的模糊地带。因此模版上面总不可避免的越来越多复杂逻辑，最终难以维护。

有了NodeJS，后端同学可以在JAVA层专注于业务逻辑与数据的开发。而前端同学则专注于控制逻辑与渲染的开发。并且自行选择这些模版是要在 服务端 (NodeJS) 或是 浏览器端 做渲染。

用著一样的模版语言 *XTemplate*，一样的渲染引擎 *JavaScript*

在 不同的渲染环境 (Server-side、PC Browser、Mobile Browser、Web View、etc.) 渲染出 一样的结果。

路由共享

也因为有了NodeJS这一层，可以更细致的控制路由。

假如需要在前端做浏览器端路由时，可以同时配置服务器端的路由，使其在 浏览器端换页 或是 服务端换页，都可以得到一致的渲染效果。

同时也处理了SEO的问题。

模版共享的实践

通常我们在浏览器端渲染一份模版时，流程不外乎是

1. 在浏览器端载入模版引擎 (xtmpleate, juicer, handlerbar, etc.)

2. 在浏览器端载入模版档案，方法可能有
 - 使用 `<script type="js/tpl"> ... </script>` 印在页面上
 - 使用模块载入工具，载入模版档案（KISSY, requireJS, etc.）
 - 其他
3. 取得数据，使用模版引擎产生html
4. 将html插入到指定位置。

從以上的流程可以觀察到，要想要做到模版的跨端共享，重点其实在 一致的模块选型 这件事。

市面上流行很多种模块标准，例如 KMD、AMD、CommonJS，只要能將NodeJS的模版档案透过一致模块规范输出到NodeJS端，就可以做基本的模版共享了。

而后续的系列文章会针对Model的proxy与共享，做进一步的探讨。

案例探讨

因为有了中途岛这中间层，针对过往的一些问题都有了更好的解答，例如说

案例一 复杂交互应用（如购物车、下单页面）

- 状况：全部的HTML都是在前端渲染完成，服务端仅提供接口。
- 问题：进入页面时，会有短暂白屏。
- 解答：
 1. 首次进入页面，在NodeJS端进行 全页渲染，并在背景下载相关的模版。
 2. 后续交互操作，在浏览器端完成 局部刷新
 3. 用的是 同一份模版，产生一样的结果

案例二 单页面应用

- 状况：使用Client Side MVC框架，在浏览器换页。
- 问题：渲染与换页都在浏览器端完成，直接输入网址进入或f5刷新时，无法直接呈现同样的内容。
- 解答：
 1. 在浏览器端与NodeJS端共享 同样的Route 设定
 2. 浏览器端换页时，在浏览器端进行Route变更与 页面内容渲染
 3. 直接输入同样的网址时，在NodeJS端进行 页面框架 + 页面内容渲染
 4. 不管是浏览器端换页，或直接输入同样的网址，看到的内容都是一样的。
 5. 除了增加体验、减少逻辑复杂度外。更解决了SEO的问题

案例三 纯浏览型页面

- 状况：页面仅提供资讯，较少或没有交互
- 问题：html在服务端产生，css与js放在另外一个位置，彼此间有依赖。
- 解答：
 1. 透过NodeJS，统一管理html + css + js
 2. 日后若需要扩展成复杂应用或是单页面应用，也可以轻易转移。

案例四 跨终端页面

- 状况：同样的应用要在不同端点呈现不同的介面与交互
- 问题：html管理不易，常常会在服务端产生不一样的html，浏览器端又要做不一样的处理
- 解答：
 1. 跨终端的页面是渲染的问题，统一由前端来处理。
 2. 透过NodeJS层与后端服务化，可以针对这类型复杂应用，设计最佳的解决方案。

总结

过去的AJAX、Client-side MVC、SPA、Two-way Data Binding 等技术的出现，都是试图要解决当时的前端开发遇到的瓶颈。

而NodeJS中间层的出现，也是在试图解决现今前端被侷限在浏览器端的一个限制。

这边文章专注于前后端模版共享，也希望能抛砖引玉，与大家一起讨论如何在NodeJS中间层这个架构下，我们可以怎样的改善我们的工作流程，怎样的跟 后端配合，来把 前端 这个工作做得更好。

 赞

 9 收藏

 1 评论



相关文章

- [基于 JavaScript 的操作系统你听说过吗？](#)
- [Web 开发在 2015 年及未来的发展趋势](#)
- [前后端分离的思考与实践（三）](#)
- [前后端分离的思考与实践（一）](#)
- [前后端分离的思考与实践（六）](#)
- [我为什么向后端工程师推荐Node.js](#)

可能感兴趣的话题


- [\[毕业就失业\]Finally It comes to me](#)
- [女生搞开发能干一辈子吗？有没有发展前景啊？ · 44](#)
- [中秋节不发福利也就算了，晚上加班测试，中秋节当天也要来加班，我也是醉了... · 56](#)
- [预算 6K 以内，求推荐开发用笔记本电脑 · 73](#)
- [昨晚加班到4点多，4.30左右才到家，好久没发帖了，来发个帖聊聊 · 51](#)
- [后端转前端，转型期真心痛苦 · 12](#)

登录后评论

新用户注册

直接登录     

最新评论



05/27




并且自行选择这些模版是要在 服务端 (NodeJS) 或是 浏览器端 做渲染。

这句话是什么意思？

还有
浏览器端渲染
是说通过ajax取得服务端的数据填入模板吗？

而服务器端渲染指的是 服务器端将数据填入模板，然后渲染成html输出到浏览器端？

是这样的吗？

 赞  回复 

文章

输入搜索关键字

搜索



- [本周热门文章](#)
- [本月热门文章](#)
- [热门标签](#)

0 [无休止加班的成因](#)

1 [我见过最有趣的代码注释，都在这里了...](#)

2 [HTTPS 科普扫盲帖](#)

3 [PM 叫你去改一个 Bug，后来.....](#)

4 [在谷歌招工程师，我看重的是这些](#)

5 [HTTPS 工作原理和 TCP 握...](#)

6 [TCP/IP之TCP协议：流量控制...](#)

7 [LOL设计模式之「策略模式」](#)

8 [.Net 高效开发之不可错过的实用工具](#)

9 [跟我一起写shell补全脚本（Zsh篇）](#)



业界热点资讯

[更多 »](#)



[甲骨文准备将NetBeans交给Apache管理](#)

11 小时前 · 2



[让计算机崩溃的简短代码](#)

3 天前 · 20



[PHP 7.1 新特性一览](#)

14 小时前 · 3



[redis3.2新功能 - GEO地理位置命令介绍](#)

1 天前 · 4



[JS 又是第一编程语言：GitHub 2016 年度报告](#)

3 天前 · 33 · 3

精选工具资源

[更多资源 »](#)



[android-remote-notifications: 从远程JSON文件拉取...
Android, 通知](#)



[PHP CPP: 一个开发PHP扩展的C++库
PHP, 扩展](#)



[Apache Mahout: 经典机器学习算法库
Java, 机器学习](#)






[WilliamChart: 优美和直观的图表库
Android, 图表](#)



[Singularity: 方便部署和操作的Mesos框架
Java, 集群管理](#)

最新评论

-  Re: [我见过最有趣的代码注释，都在这...](#)
12不是很信
-  Re: [PM 叫你去改一个 Bug，后来...](#)
现在初三，因为这个原因，理想就是不受雇于人，有一家自己的（小）公司。这种活法太恶心
- 

Re: [你应该知道的计算机网络知识](#)

同意, 因为一般一台机子只配有一个网络适配器(网卡), 所以也就默认一个MAC地址对应一台唯一的电脑...



Re: [133 行代码实现质感地形](#)

应该感谢博主搜集了这么好的资源!! 辛苦啦--



Re: [133 行代码实现质感地形](#)

谢谢提醒。已经在正文更新了 :)



Re: [PM 叫你去改一个 Bug, 后来...](#)

很赞的文章, 翻译得也非常赞。解决客户的问题, 这个是最大的价值, 当然也不必要放弃原则:
P



Re: [133 行代码实现质感地形](#)

你的连接错啦!! 现在修改了地址的 <http://demos.playfuljs.com/terrai...>



Re: [加入伯乐在线专栏作者, 扩大知名...](#)

MD 的教程, 网络上有的, 你自己搜搜。

关于伯乐在线博客

在这个信息爆炸的时代, 人们已然被大量、快速并且简短的信息所包围。然而, 我们相信: 过多“快餐”式的阅读只会令人“虚胖”, 缺乏实质的内涵。伯乐在线内容团队正试图以我们微薄的力量, 把优秀的原创文章和译文分享给读者, 为“快餐”添加一些“营养”元素。

快速链接

[网站使用指南»](#)
[问题反馈与求助»](#)
[加入我们»](#)
[网站积分规则»](#)
[网站声望规则»](#)

关注我们

新浪微博: [@伯乐在线官方微博](#)

RSS: [订阅地址](#)

推荐微信号



程序员的那些事



UI设计达人



极客范

合作联系

Email: bd@jobbole.com

QQ: 2302462408 (加好友请注明来意)

更多频道

[小组](#) - 好的话题、有启发的回复、值得信赖的圈子
[头条](#) - 分享和发现有价值的内容与观点
[相亲](#) - 为IT单身男女服务的征婚传播平台
[资源](#) - 优秀的工具资源导航
[翻译](#) - 翻译传播优秀的外文文章
[文章](#) - 国内外的精选文章
[设计](#) - UI, 网页, 交互和用户体验
[iOS](#) - 专注iOS技术分享
[安卓](#) - 专注Android技术分享
[前端](#) - JavaScript, HTML5, CSS

