

WEB 前端工程化方案浅析

杨帆

(中电福富信息科技有限公司 福建 福州 350003)

【摘要】本文阐述了如何进行 web 前端工程化的建议以及方案。

【关键词】web 前端 工程化 动化 模块化

1. 引言

随着移动互联网的大潮一路汹涌高歌,web 前端开发日新月异,同时变的越来越复杂。从早期的桌面衍生到移动端、app,需要适配不同类型不同尺寸的设备。从早期作为后端 MVC 框架的简单一层,演进成有自己的 MVC 或 MVVM 等开发框架。从早期只是简单的 html、javascript、css 演变成如今有大量的内部或外部的组件和模块。从早期只是简单的源码发布,到现在需要预处理、压缩、合并等一系列复杂的打包操作。前端以前只是后端工程师的一个任务,现在在很多公司和团队内已经成为一个岗位,需要大量的人进行协同开发。因此为了降低开发成本和复杂性,提高效率,实现更好的协同,前端工程化已经成为一个越来越迫切的事情。本文主要从规范化、组件化、模块化、自动化这四个方面来探讨前端工程化的实践方案。

2. 规范化

所谓规范化即制定统一的开发规范来约束团队成员,便于协作和代码维护。规范并没有普适性的标准,每个团队可以建立自己的一套符合自身的规范体系。

对于 web 前端来说,开发规范主要应考虑以下几个方面:

- * 制定统一的目录结构规范:保证源码和编译输出目录分开,同时区分外部框架、组件和内部组件,区分静态资源、页面、配置文件和逻辑代码等

- * 统一的编码规范:包括通用规则如文件资源命名规范、资源协议、文本缩进、注释、代码检查等。HTML 规范如文档类型、标签、脚本加载、语义化等。javascript 规范如名称空间、变量、函数、闭包、继承等。css 规范如类命名、选择器、缩写属性、单位、hacks 等。如果有使用其他语言如 Emcascript、typescript 或开源框架如 angularjs、vue 等也应该制定相应的规范。

- * 前后端接口规范:包括通用的 rest 规范如 http 方法、url、响应状态码等。请求格式规范如 json、表单编码格式等。响应格式如 json、jsonp、html 等。还有其他如安全、缓存、实体定义等的统一定义

- * 文档规范:主要包括需求文档、概要设计文档、详细设计文档、操作手册、维护手册等,需要根据每种类型制定统一的模板,保证文档具有针对性、精确性、清晰性、完整性、灵活性和可追溯性。

- * 其他规范:如组件开发管理规范、版本管理(git、svn)规范、版本提交规范、代码检查规范、图标规范等。

3. 组件化

对一些可复用的功能、页面元素进行封装后形成组件,以组件为基础自由搭配组合出不同的功能称之为组件化开发。组件化最重要的思想在于分而治之,对于所有的软件开发框架来说组件化都是一个不可缺少的手段,而 web 前端的组件化与其

的区别在于把图形和非图形各种逻辑和交互抽象成一个统一的实体。组件化主要应该从以下两个方面来考虑。

- * 从 UI 角度来设计组件:首先组件必须满足业务逻辑,不能单单只是为了视觉效果,交互过程及逻辑的合理性对于用户更为重要。其次要保持视觉风格的统一,对于相同的功能和目的的组件使用相同视觉手段。最后要保持交互的一致性,比如错误的提示方式、进度条等待框的交互方式、导航栏的样式等等都要相同的交互风格。

- * 从开发角度设计组件:对于研发来说组件其实是按模板(html)+样式(css)+逻辑(javascript)三个组合起来进行面向对象的抽象。通常来说可分为通用组件如日期选择控件、表格控件,业务组件如搜索建议框、组织树等。首先组件必须为复用按职责进行设计,具有明确的组件接口。其次组件必须尽量减少耦合度,即减少对其他组件的依赖和影响。再次组件应该考虑可扩展性,包括逻辑上的继承、样式的扩展、模板的定制等等。最后组件还应考虑减少冗余、性能优化、日志跟踪排错等其他方面的问题。

4. 模块化

简单来说,将一个大的功能拆分成几个相互关联的小功能,再进行统一的加载和拼装,称之为模块化开发。只有模块化了,才能拆分大的功能,进行多人的协作开发。模块化的关键点在于高内聚低耦合。对于 web 前端来说,至少需要解决以下三种类型的模块化:

- * javascript 的模块化:由于 javascript 一直没有现成的模块化方案,这对开发大型复杂的应用造成了很大的障碍。因此开源社区制定了多种的模块加载方案,如 CMD、AMD、COMMON-JS 等,并且有对应的加载组件如 requireJs、seaJs 等。某些开源框架也有自己的模块系统,如 angularjs、vue 等。虽然百花齐放,但是毕竟没有统一的标准。随着 EmcaScript6 的出台,在语言层面直接植入了模块系统,完全可以替代开源的方案,而且使用简单,具有很多特性。因此使用 EmcaScript6 来进行开发已渐渐成为趋势,但是由于目前很多的浏览器并不能直接支持 EmcaScript6,因此需要一个编译、加载的过程,目前业界主流主要是通过 webpack+Babel 的工具来实现

- * CSS 的模块化:目前有很多的 css 预处理器如 sass、less、PostCSS 等,解决了 css 动态能力弱的问题,但是依然没有解决全局污染、命名混乱、依赖管理不彻底等模块化关心的问题。为了避免冲突,各种开源都制定了自己的命名风格,但这是一种比较弱的约束,并不能消灭上面的问题。只要真正的自动化,才能解决问题,因此出现了 Shadow DOM、CSS in JS、CSS Modules 三种解决方案。Shadow Dom 由于是 Web Components 的标准,目前很多浏览器不支持。而 CSS in JS 则是完(下转第 157 页)

远程访问服务”角色。默认情况下,在安装完 Windows Server 2008 之后,需要手工添加这一角色,步骤如下:

- (1)“开始”—“管理工具”—“服务器管理器”;
- (2)“添加角色”—“添加角色向导”;
- (3)“选择服务器角色”—“网络策略和访问服务”—“下一步”;
- (4)“网络策略和访问服务简介”—“下一步”;
- (5)“选择角色服务”—“路由和远程访问服务”—“下一步”;
- (6)“确认”—“安装”—“进度”—“正在启动安装”;
- (7)“安装结果”—“成功”—“关闭”。

3.3.2 安装“路由和远程访问服务器”

添加“路由和远程访问服务”角色后,接着安装“路由和远程访问服务器”,步骤如下:

- (1)“开始”—“管理工具”—“路由和远程访问”—“路由和远程访问服务器安装向导”—“下一步”;
- (2)“路由和远程访问服务器安装向导”—“自定义配置”—“下一步”;
- (3)“路由和远程访问服务器安装向导”—“LAN 路由”—“下一步”;
- (4)“正在完成路由和远程访问服务器安装向导”—“完成”;
- (5)“路由和远程访问”—“启动服务”。

3.3.3 设置静态路由

选择 3.3.2 节的步骤(5)中“IPV4”—“静态路由”右击,按照图 3 所示的方法分别添加两条静态路由如图 4 所示:



图 3 设置静态路由

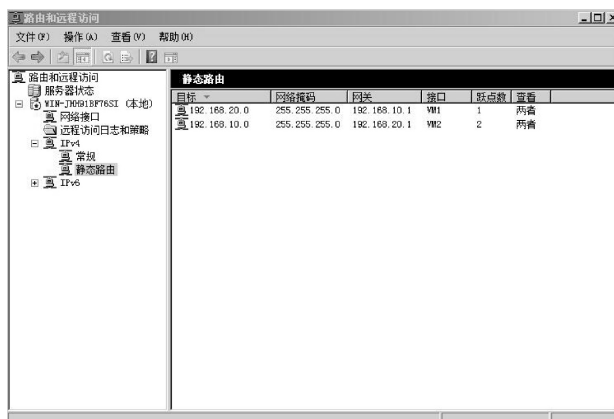


图 4 添加静态路由

3.4 测试

路由器配置好之后,克隆 1 和克隆 2 可以通信。在网络 Vnet1 上的计算机克隆 1 上用“ping”网络 Vnet2 上的计算机克隆 2 的 IP 地址 192.168.20.2 能“ping”通。

同理,在网络 Vnet2 上的计算机克隆 2 上用“ping”网络 Vnet1 上的计算机克隆 1 的 IP 地址 192.168.10.2 也能“ping”通。

4、结束语

综上所述,虚拟机技术极大方便了计算机网络专业的实践教学。VMware 虚拟机技术可以在不额外添加硬件设备的前提下,充分挖掘现有硬件设备的潜力,极大提高了硬件设备的使用率,增加了学生网络实践动手操作的机会。因此,在以后的计算机网络专业教学活动中,作为计算机网络专业老师的我们应充分利用虚拟机技术来改进自己的日常教育教学工作,提高网络实训教学效果。

参考文献:

- [1] 赖伟骏.VMware 虚拟机技术在 Windows 2000 软路由实验中的应用[J].黎明职业大学学报,2005.
- [2] 李佳等.采用 VMware 构建虚拟网络实验室[J].重庆广播电视大学学报,2007(3).
- [3] 谢鹏.使用 VMware 技术实现 Active Directory 组网实验[J].黑龙江科技信息,2010(24):86-134.

(上接第 152 页)

全使用 js 来替代 css,不能利用现有的 css 技术。CSS Modules 仍然使用 css,但是使用 js 来关联依赖,从目前来说是最优的解决方案

* 资源的模块化:即其他资源如图片、xml、json、视频等等的模块化。资源模块化会带来依赖关系单一、处理集成化、项目结构清晰的好处。目前通常使用 webpack 等工具来实现。

5. 自动化

对于软件开发来说,无论怎么强调自动化都不过分,能够让机器干的活绝对不要叫人手去做。web 前端主要要实现以下几种自动化

* 自动构建:包括预编译、依赖打包、资源嵌入、文件压缩、hash 指纹、代码审查等。

* 自动测试:包括单元测试、UI 界面测试。目前 UI 界面测

试还不是很成熟,主要 Selenium、phantomjs 等工具来实现。

* 持续集成:对 web 前端来说持续集成一种良好的软件开发实践。每次集成通过自动化平台串联各个开发环节,实现和沉淀工作的自动化,从而快速的发现错误,防止版本分支大幅偏离主干,让产品可以快速迭代,同时还能保持高的质量。目前通常使用 gitlab、Jenkins、webpack 等工具来实现各个环节的自动化。

6. 后记

本文大致介绍了 web 前端工程化的方案。目前该方案在我司正在积极的实践并且取得良好的效果。前端领域发展很快,各种新技术新思想不断涌现,在整个 web 开发体系中,前端工程师的角色越来越重要,但是前端工程化还远远不够成熟,需要在后续的实践中继续不断的完善。