

SPA 设计与架构

理解单页面Web应用

SPA Design and Architecture:
Understanding Single-page Web Applications

[美] Emmit A. Scott 著
卢俊祥 译



SPA 设计与架构

理解单页面Web应用

SPA Design and Architecture:
Understanding Single-page Web Applications

[美] Emmit A. Scott 著
卢俊祥 译



电子工业出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

SPA 开发技术的运用是当今 Web 开发领域的热门趋势,但真正全面掌握该技术的开发者并不多。本书详尽阐述单页面 Web 应用(SPA)开发技术,从 SPA 构建基础入手,通过 MV*、模块化编程、路由、模块间通信、服务器端交互等概念的阐述,全面介绍 SPA 的设计与架构,帮助读者正确掌握 SPA 开发的各方面知识要素。同时,书中还讨论了 SPA 的单元测试及客户端任务自动化,覆盖了从开发到部署的一系列任务,让读者在阅读完本书之后能够打下扎实的 SPA 开发基础。

本书的重点是帮助读者正确、全面地掌握 SPA 开发概念,这些概念都是通用的。但为了让内容更全面、具体,本书将通过 Knockout、Backbone.js 及 AngularJS 这三种不同风格的 MV* 框架来进行比较性讨论,这是本书的一大特色。同时在涉及具体 MV* 框架知识点时,书中会提供相应介绍。书中示例丰富具体,并提供完整源代码下载。

本书适合前端及对 SPA 技术感兴趣的开发者阅读。读者只需掌握 JavaScript、HTML 和 CSS 基本知识,就可以阅读本书。

Original English Language edition published by Manning Publications, USA. Copyright © 2016 by Manning Publications. Simplified Chinese-language edition copyright © 2016 by Publishing House of Electronics Industry. All rights reserved.

本书简体中文版专有版权由 Manning Publications 授予电子工业出版社。未经许可,不得以任何方式复制或抄袭本书的任何部分。专有版权受法律保护。

版权贸易合同登记号 图字:01-2016-1162

图书在版编目(CIP)数据

SPA 设计与架构:理解单页面 Web 应用 / (美)埃米顿·A.斯科特(Emmit A. Scott)著;卢俊祥译.
北京:电子工业出版社,2016.11

书名原文:SPA Design and Architecture: Understanding Single-page Web Applications
ISBN 978-7-121-30091-2

I. ①S… II. ①埃… ②卢… III. ①网页制作工具—程序设计 IV. ①TP393.092

中国版本图书馆CIP数据核字(2016)第247063号

策划编辑:张春雨

责任编辑:李云静

印 刷:北京天宇星印刷厂

装 订:北京天宇星印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编:100036

开 本:787×980 1/16 印张:19 字数:375 千字

版 次:2016 年 11 月第 1 版

印 次:2016 年 11 月第 1 次印刷

定 价:79.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式:010-51260888-819 faq@phei.com.cn。

译者序

当第一次使用 Gmail 时，我被它那流畅的原生桌面般体验迷倒，赞叹天底下居然有如此令人惊艳的 Web 应用。之后，SPA 的概念逐渐盛行，越来越多关于 SPA 的介绍、实践分享进入开发者的眼帘。然而，在现实开发世界里，我发现很多开发者对 SPA 开发技术的整体概念是模糊的，往往以为只需靠 Ajax 技术，就能很好地实现 SPA。

在我大致浏览了本书的内容后，立刻感受到这是一本非常不错的 SPA 开发书籍。事实上，这本书在 Amazon 网站中获得了非常不错的读者评价。在翻译过程中，我自己也强化了不少 SPA 开发概念，并受益匪浅。

Web 前端开发技术可谓是当今变化最为频繁的软件开发技术，新的开发理念、新的框架层出不穷，同时 ES 6 在语法上带来了诸多变化，这些都迫使我们不断去适应新趋势的发展。但本书着力于 SPA 开发技术的基本原理。掌握了这些基础知识，就能够做到相当程度的以不变应万变，这也是本书吸引我的地方。

岁月如梭，能够沉浸在技术创造的乐趣中是一件让我无限期待的事情。编写出优美的代码，构建出极致的应用，是每个热爱创造的开发者共同的追求目标。真心期待本书能够带给你不一样的收获。

同时，林长瑞、吴桐、朱建宝、周荣华、吴胜华、叶铭辉、李禧强、姚建峰、郑秀玲亦不同程度地参与了本书的翻译工作。

感谢我的妻子和娃，你们给了我很大的支持，小宝贝还给我带来了许许多多的乐趣。同时还要感谢本书的策划编辑张春雨，在你的鼓励下，我的翻译过程充满愉悦。

卢俊祥

2016年10月

本书献给我的三位可爱宝贝：Ana Carolina、David 和 Sofia。

感谢你们所给予的全部微笑、拥抱和无私的爱。

你们永远是爸爸的甜心。

序

1991 年的时候，Tim Berners-Lee 推出了全球首个网站，这个网站运行在一个他命名为“WorldWideWeb”的程序上。两年后，他发布了 WorldWideWeb 的源代码，世界从此发生了天翻地覆的变化。迄今仍可在 info.cern.ch 浏览人类历史上的第一个 Web 页面。

自 1991 年起，Web 得到了空前的发展。在它 24 岁的时候¹，其仍是 IT 世界里使用最广泛的技术。Web 以某种形式运行于各种操作系统、硬件平台及绝大部分的移动设备之上。让这一切成为可能的软件就是万能的 Web 浏览器。

传统上，Web 浏览器是简单的中间人角色。其从服务器端获取数据，展示数据，再将数据发回服务器端，然后又获取更多的数据并展示。而今天的 Web 浏览器，虽仍坚守初心，但其复杂程度已远非当年可预见。

当年简陋的浏览器已经发展成为各式各样应用程序的运行时成熟环境。无须安装，就能随处访问、运行这些应用程序。这就是开发者的“屠龙技”。部署四处运行且即时更新的代码库——这种魅力实在是难以抗拒。还没有其他任何一种技术能够做到如此得意。

站在 Web 平台成功之巅的是无处不在的 JavaScript——一门 10 天内发明出来的语言，它大概是目前世界上使用最广泛的编程语言了。开发者们已经接纳了

¹ 本书著于 2015 年。——译者注

JavaScript，它帮我们打开了崭新应用类型的大门，之前做梦都想不到这一切会发生在 Web 浏览器中。

这些新型应用程序，我们通常称之为单页面应用程序（SPA），几乎完全在浏览器中运行，其引入了一套全新的规则、模式及问题。Web 的广泛吸引力带来了层出不穷的 JavaScript 和 CSS 框架；框架如此之多，以至于要做出合适的挑选犹如大海捞针。

亲爱的读者，这就是本书如此重要的原因。

在过去的 4 年里，我作为一名开发者在 Telerik 工作，我积极倡导对 Kendo UI JavaScript 库的关注。我看到过太多的 JavaScript 框架起起灭灭。当某个流行框架的炒作达到临界状态时，下一个“大事件”¹就随之而来，只留下那些在所谓“时髦”框架之上实际构建解决方案的开发者们仍在收拾残局。这让我总想知道它何时能够消停下来，这样我们就能够专注于“正确之道”，以构建新一代的富客户端应用程序。

残酷的现实却是：做任何事情都不存在所谓的“正确之道”。唯一的办法就是为你的项目和技能栈而战。这是让你更具生产力并最终成功的唯一姿势。

为了在 SPA 开发世界里杀出一条路，理解 SPA 概念之下的基本原理是很有必要的。掌握一门框架还远远不够，因为这样的话，最终仍会让你感到贫瘠和匮乏。深度理解成功 SPA 构筑之法的核心理念，使你得以从容决策，在借助 JavaScript 框架完成了 80% 功能之后，懂得如何构建剩下的 20%。

本书就是你的指导手册——不管你是专家还是新人。在读它的时候，我发现自己在恶补过往不求甚解的基础，并对之前感觉甚好而实际上只是部分理解（甚至理解错误）的术语有了新认识。这些认识和解释埋藏在理论结合实践的字里行间，并在讨论 SPA 框架的时候教你如何构建 SPA、正面处理现实世界需求。

那些我通常持怀疑态度的书籍，都试图解释跟 SPA 一样大的概念，但是这部书籍的亮点却在于——另辟蹊径设法将复杂主题化解为易于理解与消化的部分。

我毫无保留地倾力推荐这本书——每一页，每一句。

Burke Holland

Telerik 开发者关系主管

1 比如发布各种眼花缭乱的、破坏兼容性的新特性。——译者注

前言

我经历过的许多项目，都需要花费一年甚至更多的时间来持续构建。当然，之后还有各种更新和意想不到的事情在等着我们处理。由于这些类型的项目耗时长久，而在这期间，技术的发展突飞猛进；因此，在我准备开始下一个项目的时候，却发现不得不重新评估我的技术栈，因为事物的变化早已今非昔比。

当我和团队在准备最近一次的 SPA 项目时，我萌生了编写这本书的想法。我的主管允许我们探研项目需要的“最佳”技术栈。因此，我们开始评估各种解决方案并创建小型概念验证应用程序。

结束评估工作之后，我对在海量信息中筛选出方向感到心有余悸。对于那些构建 SPA 应用的新手而言，我的艰辛历程同样也是他们要面对的困难。

因此我打算写一本书，不仅会总结构建 SPA 所需知识点，还将介绍构建 SPA 相关的一些第三方库或框架。此外，我希望这本书的内容简单直接并容易消化，同时还能够提供足够的技术细节，争取让读者在读完内容之后，具备实际构建 SPA 应用的能力。

感谢你跟我一起出发。我希望你最终能够发现这本书是一部不可或缺的 SPA 开发指南。

致谢

这是我写的第一本书，因此你可以想象到个中的艰辛。然而我非常幸运，有三个朋友始终在我身后给予写作上的大力相助：Dan Maharry——我的项目编辑；Joel Kotarski——我的技术编辑；还有 Andrew Gibson——我的技术校对。他们是了不起的团队，非常感谢他们！

非常感谢文字编辑 Sharon Wilkey 的细致，还要感谢 Jean-François Morin 在 Java/Spring 方面的技术校对工作。

同时也要感谢 Manning 的其他工作人员，他们让这本书得以顺利出版：Marjan Bace、Michael Stephens、Bert Bates、Maureen Spencer、Kevin Sullivan、Mary Piergies、Candace Gillhoolley、Rebecca Rinehart、Ana Romac、Toni Bowers 和 Linda Recktenwald，以及许多付出汗水帮助我实现目标的无名英雄。

特别的感谢送给 Burke Holland，他撰写了本书序。我是 Burke 的忠实“粉丝”，他能为本书写序让我倍感荣幸。

感谢那些编写过程中的内容审核者：Alain Couniot、Anirudh Prabhu、Bruno Sonnino、David Schmitz、Fernando Monteiro Kobayashi、Johan Pretorius、John Shea、Maqbool Patel、Philippe Vialatte、Rajesh Pillai、Shrinivas Parashar、Trevor Saunders、Viorel Moisei 以及 Yogesh Poojari。

同时感谢我的家人：我的妻子 Rosalba、我的女儿 Ana Carolina 和 Sof、我的儿子 David，以及我的妈妈 Lucy。言语已难以表达我的感激之情——编写过程中你们赋予我如此多的关爱、支持和鼓励。

我的朋友和同事，无论我们是否在一起，我都要感谢你们的鼓励和支持。

关于本书

本书引导你如何创建单页面应用程序。不仅介绍 SPA 构建所需的框架和技术，同时还介绍单元测试、客户端开发与构建任务的自动化。

由于 SPA 构建的过程细节决定着技术栈的选择，因此本书会使用当今主流的 JavaScript 框架来比较多种方式。为什么会存在这么多框架呢？原因之一就是应用构建没有标准方式。通过比较不同框架，可以更好地决定哪种方案更适合你的项目。

本书中的每章都包含了一个完整可行的应用示例。我尽量保持内容的有趣和简单。我并不喜欢那种用一页页源代码填充了半本书的大型示例，这让人读起来太费力了。因此，我决定为每章创建独立的项目。与此同时，尽可能保持每个示例短小精悍，紧扣当前章节的概念主题并不失趣味性。

路线图

第 1 部分：基础知识

- 第 1 章介绍 SPA 整体概念。一开始你将了解到重要概念及 SPA 应用与传统 Web 应用间的差异。我会给出一个清晰简明的定义，并简要说明 SPA 的各个部分是如何结合起来的。
- 第 2 章会继续深入一层，介绍 JavaScript 框架的一个“流派”——MV* 框架，

同时介绍它们在单页面应用程序创建中的作用。本章还会探讨框架之间的共性以及差异性。章末示例使用三种 MV* 框架来分别创建，让你在实践中感受不同的构建风格。

- 在第 3 章中，你将了解模块化编程。这里会通过示例直观了解应该在 SPA 中使用模块的原因。我们还将分解模块模式的语法，并一步步解释它。本章最后介绍模块加载及 AMD 模块的内容。

第 2 部分：核心概念

- 第 4 章快速介绍客户端路由的知识点。你将了解路由的运行机制，以及各种框架如何处理路由。本章也会介绍客户端路由影响应用程序状态的方式。
- 第 5 章介绍 SPA 的布局设计与视图合成。一开始先了解简单设计，然后借助复杂路由逐步深入到复杂设计。此外，我们还将涉及高级的布局主题，如嵌套视图与并列视图。
- 第 6 章讨论模块间通信。如果创建的模块不能相互通信，那还有什么意义呢？！在本章中，你不仅会看到模块间通信的不同方式，也将了解模块化应用设计之道。
- 第 7 章将阐述 SPA 环境中的服务器端处理。尽管客户端仍是我们关注的焦点，但也应该了解 SPA 中的服务器端通信，以及如何处理服务器端调用的返回结果。本章会讨论通过回调函数与 Promise 方式处理返回结果。你还将看到 MV* 框架如何帮助我们完成这些任务。本章最后将简短介绍 REST 及 SPA 应用程序如何调用 RESTful 服务。
- 第 8 章是一个关于 JavaScript 应用程序单元测试的概览，特别针对 SPA 应用。如果你从未接触过客户端单元测试，也不用太过担心。在这里我们将放缓脚步、降低难度，一步步了解相关的基础知识。
- 最后是第 9 章，在这里会讨论客户端任务自动化在开发和构建过程中带给我们的帮助。同时从整体上了解针对各种场景的最常用任务类型，并在本章最后的示例项目中实践这些任务。

附录是章节的补充。附录 A 解析第 2 章示例项目（三个版本）的完整源代码。附录 B 和附录 C 对第 7 章做补充。附录 B 的内容为 XMLHttpRequest API 概览，附录 C 是该章示例项目服务器端调用部分的总结。尽管第 7 章的内容组织也考虑到让你能够使用自己的服务器端语言，但附录 C 仍额外包含了一个 Spring MVC 代码指南，这些代码包含在下载源文件里。附录 D 是关于 Node.js 和 Gulp.js 安装的简单指南，这些内容是实践第 9 章代码所需的。

阅读对象

本书假设你至少具备 JavaScript、HTML 和 CSS 方面的基本知识。如果你有一定的 Web 开发经验，那也是很有帮助的，但这不是必需的。然而，这本书的目标人群是那些在 SPA 方面欠缺经验甚至根本没经验的开发者，以及在构建 SPA 过程中未使用过本书所涵盖技术的开发者。

代码约定与下载

程序清单中的源代码，以及在正文中为了区别于普通文字的代码，都以等宽字体方式呈现。程序清单中还包含代码注解，以突出重点概念。

本书示例源代码可以通过出版社网站下载：<https://www.manning.com/books/spa-design-and-architecture>。

软硬件要求

如果你使用最新的 Mac OS X 或 Windows，以及现代浏览器（如 Firefox、Safari 或 Chrome），应该可以正常运行书中示例。对于特定软件要求，可以在相关章节或附录中找到答案。

由于大多数示例都动态获取 HTML 文件，如果你在本地（而非服务器）运行书中示例，可能需要在浏览器中设定某些权限。请参考下载源文件里的项目 `readme.txt` 文档以获取具体信息。

如果你想运行第 7 章示例，则需要用到 Web 服务器及服务器端语言。每位开发者都有自己的技术栈，因此你可以决定你自己的技术路线。我使用的是 Java 和 Spring MVC，并在书中提供了一个对应的简短设置指南。如果你使用不同的软件，书中也从概念上描述了服务器端调用及相关对象，以便你能够通过你自己的技术栈重建示例应用。

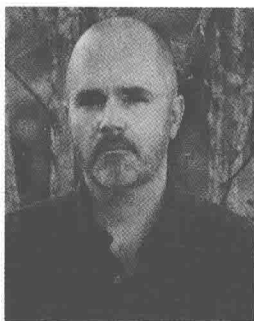
作者在线

购买本书的读者可以免费访问 Manning 出版社维护的专用论坛，在论坛里读者可以评论本书、提出技术问题并得到作者和其他开发者的帮助。要访问论坛并订阅信息请访问 <https://www.manning.com/books/spa-design-and-architecture>。该页面提供了注册后如何访问论坛的指引、各种帮助信息以及论坛行为准则。

Manning 以尽责的态度提供一个读者间、读者与作者间互动的空间。Manning 无法承诺作者的参与程度，其对论坛的贡献基于自愿而免费原则。我们建议你尽量向作者提一些富有挑战性的问题，以保持作者的热情！

只要书籍得以出版，你就可以通过出版社网站访问作者在线论坛以及早期的讨论归档内容。

关于作者



Emmitt A. Scott 是一名有 17 年 Web 应用构建经验的高级软件工程师和架构师。他为教育、银行和通信领域开发过大型应用程序。他的爱好包括阅读（特别喜欢 Jim Butcher 的小说）、吉他（想当年他可是一位摇滚乐手）以及尽可能多陪陪孩子。

译者简介

卢俊祥

译者，书迷；关注 Web 技术趋势，热衷 App 开发、Web 开发、数据分析、架构设计以及各类编程语言；陈氏太极拳五十六式爱好者；佛禅人生，缘散缘聚。

微博：@2gua

个人网站：<http://www.2gua.info/>

知乎专栏：<https://zhuanlan.zhihu.com/guagua/>

关于封面图画

本书封面图画为“1700 年年轻土耳其贵族服饰”。这幅图出自 Thomas Jefferys 的《古今各国服饰集》(*A Collection of the Dresses of Different Nations, Ancient and Modern*)——1757 至 1772 年间出版于伦敦，扉页标明这些画是手工上色的铜版雕刻印刷品，用阿拉伯树胶加固。Thomas Jefferys (1719—1771)，被称为“乔治三世的地理学家”。他是一位英国制图大师，乃当时顶尖的地图制作大师。他为官家镌刻及印制地图，制作了大量商业的地图和地图集，尤其是北美地域的作品。Thomas Jefferys 作为一名地图制作大师，其调查及绘制之地的服饰也引起了他的兴趣，而成果就出色地体现在了这四卷册中。

向往远方而游历，在 18 世纪后期还是一个新景象，当时这样的卷册极受欢迎，其向远游者和“神游旅行者”介绍了不同国家的居民。Jefferys 卷册里丰富的画面生动地讲述了大约 200 年前世界各国的独特魅力。自那时起着装发生了变化，同时那个年代丰富的地域多样性也逐渐褪去。现在要分辨出不同大陆的居民已非常困难。也许应该尽量乐观地看待这种现象，我们失去文化及视觉多样性的同时，却换来了多样化的个人生活、更加多样有趣的知识和科技生活。

正当计算机书籍内容的同质化现象日益突出的今天，Manning 借助 Jeffreys 的图画将昨日重现，参考两个世纪前地域生活的丰富多样性来设计图书封面，谨表达对计算机领域开创精神的赞誉。

目录

第 1 部分 基础知识	1
1 单页面应用程序介绍	3
1.1 SPA 简述	4
1.1.1 无须刷新浏览器	7
1.1.2 表现逻辑位于客户端	7
1.1.3 服务器端事务处理	7
1.2 更进一步	8
1.2.1 以 Shell 页面开始	8
1.2.2 从传统页面到视图	9
1.2.3 视图的产生	10
1.2.4 实现无刷新的视图切换	11
1.2.5 贯穿动态更新过程的流畅性	12
1.3 SPA 应用相较传统 Web 应用的优势	12
1.4 温故知新	13
1.5 优秀 SPA 应用的构成	15
1.5.1 组织项目	15

1.5.2	创建可维护的松耦合 UI	17
1.5.3	使用 JavaScript 模块	18
1.5.4	执行 SPA 导航	19
1.5.5	创建视图组成与布局	19
1.5.6	模块通信	20
1.5.7	与服务器端通信	20
1.5.8	执行单元测试	20
1.5.9	客户端自动化技术	20
1.6	小结	21
2	MV* 框架介绍	22
2.1	MV* 概念	24
2.1.1	传统 UI 设计模式	25
2.1.2	MV* 和浏览器环境	27
2.2	MV* 基础概念	28
2.2.1	框架	29
2.2.2	我们的 MV* 项目	30
2.2.3	模型	32
2.2.4	绑定	36
2.2.5	模板	40
2.2.6	视图	44
2.3	为什么要用 MV* 框架	44
2.3.1	关注分离	45
2.3.2	简化日常任务	46
2.3.3	提升生产率	47
2.3.4	标准化	47
2.3.5	可扩展性	48
2.4	框架选择	48
2.5	挑战环节	50
2.6	小结	50
3	JavaScript 模块化	52
3.1	模块概念	53
3.1.1	模块模式概念	53