

版权注意事项：

- 1、书籍版权归作者和出版社所有
- 2、本PDF仅限用于个人获取知识，进行私底下的知识交流
- 3、PDF获得者不得在互联网上以任何目的进行传播
- 4、如觉得书籍内容很赞，请购买正版实体书，支持作者
- 5、请于下载PDF后24小时内删除本PDF。

Front-End Tooling

with Gulp, Bower and Yeoman

【奥】斯特凡·鲍姆加特纳 著
(Stefan Baumgartner)
谈博文 译

Web

前端自动化构建

Gulp、Bower和Yeoman开发指南



机械工业出版社
China Machine Press

本书重点关注的是广义的开发工作流，涵盖了开发者开发和维护项目时将会面临的三个阶段，分别是创建、开发和部署。对于每个阶段，本书都详细解释了这一阶段的需求和概念，并且介绍了一个合适的工具来实现这些需求。

全书共10章。第1章展现了前端开发者日常面临的挑战，并且介绍了如何用三种类型的工具建立工作流以克服这些挑战。第2章介绍了第一种工具，也就是构建工具Gulp。第3章介绍依赖和执行链。第4章引入依赖管理的概念，并选择Bower作为依赖管理的工具。第5章是前端工作流的最后一步，介绍如何通过脚手架工具 Yeoman来复用第2章到第4章中搭建好的流程。第6章介绍如何利用Gulp的特性来改变Gulp在不同开发环境中的输出结果。第7章深入解释 Gulp 的底层技术——文件对象流。第8章介绍了Gulp 的插件生态系统，并且告诉读者在选择新插件作为构建管道时，如何避免冗余的或者质量不好的插件。第9章介绍了模块的概念，并且展示几个不同的模块规范之间的差异。第10章以脚手架工具Yeoman来结束全书。

Web前端自动化构建

Gulp、Bower和Yeoman开发指南

[奥] 斯特凡·鲍姆加特纳 (Stefan Baumgartner) 著

谈博文 译

Front-End Tooling

with Gulp, Bower and Yeoman



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

Web 前端自动化构建: Gulp、Bower 和 Yeoman 开发指南 / (奥) 斯特凡·鲍姆加特纳 (Stefan Baumgartner) 著; 谈博文译. —北京: 机械工业出版社, 2017.9
(Web 开发技术丛书)

书名原文: Front-End Tooling with Gulp, Bower and Yeoman

ISBN 978-7-111-57883-3

I. W… II. ①斯… ②谈… III. 超文本标记语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2017) 第 209489 号

本书版权登记号: 图字: 01-2017-0731

Stefan Baumgartner: *Front-End Tooling with Gulp, Bower and Yeoman* (ISBN 978-1617292743).

Original English edition published by Manning Publications Co., 209 Bruce Park Avenue, Greenwich, Connecticut 06830.

Copyright © 2017 by Manning Publications Co.

All rights reserved.

Simplified Chinese translation edition published by China Machine Press.

Copyright © 2017 by China Machine Press.

本书中文简体字版由 Manning 出版公司授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

Web 前端自动化构建: Gulp、Bower 和 Yeoman 开发指南

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 和 静

责任校对: 李秋荣

印 刷: 三河市宏图印务有限公司

版 次: 2017 年 9 月第 1 版第 1 次印刷

开 本: 186mm×240mm 1/16

印 张: 13.75

书 号: ISBN 978-7-111-57883-3

定 价: 59.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

The Translator's Words 译者序

遥想当年用 jQuery 刀耕火种的时代，如今的前端开发者幸福太多了。从 UI 组件到前端框架，从语法糖到预处理语言，前端生态圈的变化可谓是日新月异。但是别看现在的开发者熟稔 React、Vue、ES6/ES7 等技术，假如让他从零开始开发一个项目，他可能就无从下手、不知所措了。

原因就在于，现在的前端项目的构建太复杂了。先不论前端框架，ES6 这么流行，不上个 babel 简直说不过去。还有 CSS，暂且不说 PostCSS，Sass 总得来一个吧。等到要上生产环境，合并、混淆也是必不可少的……搭建一个现代 Web 应用所需要的工具太多了，实在列举不完。这其中步骤之复杂、细节之烦琐，每一个搭建过前端构建流程的朋友肯定都深有体会。

对于初学者而言，前端工具更是难以入门。绝大部分开发者一般都没有机会参与搭建构建流程的环节。而且一般只有在项目初始阶段才需要搭建流程，之后就不会再去修改它。等到开始下一个项目时，构建工具又已经发生了翻天覆地的变化。这就导致了前端构建方面的知识难以积累和沉淀。

本书就非常适合前端构建的初学者。本书所介绍的 Gulp、Bower 和 Yeoman 都是业内流行且易于上手的工具。而且本书的目的并不仅仅是介绍工具的用法，还对前端构建整体的概念和思路进行了梳理。即使未来出现了更先进、更时髦的工具，总体的思路依然万变不离其宗，本书也不会因此而过时。

译者在翻译的时候力求用词准确、行文流畅，但也难免有所纰漏，恳请广大读者来信批评指正 (tanbowensg@gmail.com)。

自序 *Foreword*

我十几岁的时候就喜欢上网。正如所有小时候的爱好一样，网络对我也产生了深远的影响。在网上人们几乎可以发布任何信息，传播开放的思想。无论是谁在网络上都有发声的权利（无论是好是坏）。正是这一点吸引了我，我也因此从网络的用户成为了一名开发者。

我曾经深入钻研过《勇者斗恶龙》这款游戏。我曾在 11 年间，在网上发布了超过 1500 页的游戏攻略。一种对于文字和电影的热情驱使着我制作 podcast 来表达自己的想法，这一过程又持续了 4 年。在我硬盘的某个角落里，至今还存有以前未完成的视频原型，幸好它们永远也不会有机会重见天日了。我所有的爱好和疯狂的想法都以某种形式发布到了网上，而且也一直都受到好评。

直到 2010 年（可以说是任天堂的一年），我才意识到我最大的爱好并不是上面提到的那些，尽管我为它们也付出了很多。曾经有人邀请我给更多电脑游戏写攻略，当时我有点膨胀、骄傲、意气风发，但是很快我就意识到，我已经不像以前那样对电脑游戏那么感兴趣了，我再也不可能写出以前那么高质量的攻略了。但是我可以把精力用在创作所有作品所使用的工具上，让别人来创作出更好的作品。那就是所有网站的基础——HTML、CSS 和 JavaScript。

我辞去了图像处理的工作，我一直对它不感兴趣。从此以后我都管自己叫网站开发者。我很幸运地被当地的一家网络公司雇用了，能够从事自己喜欢的工作。那是我有生之年最正确的决定。不久之后，我还有幸领导过几个世界上最棒的前端开发团队。我发现在工作中最有挑战性的事情是：制定代码规范、项目架构和自动化流程，并且以一种高效而温和的方式推广给团队成员。这些技术并不是鸡毛蒜皮的事情，它们能

真正改善开发体验。我们尝试了很多工具，但是真正取得进展还是在 Yeoman 问世之后。从此以后，对于每一个新项目我都会用到 JavaScript 构建工具、依赖管理工具和脚手架工具，来保证项目能顺利地开始。

就像这本书中，展现了我和网络的缘份。我是由于喜欢发表东西才喜欢上网络，最后又回到了本源，发表了关于网络技术本身的东西。而且这次不是在网上发表，而是印刷出版了，这在几年前是我完全无法想象的。

如今网络还在影响着我。通过网络，我认识了遍布世界各地的朋友。网络还让我有机会在各个国家和城市之间旅行。甚至我还通过网络认识了我的妻子，未来网络一定还能给我更多惊喜。

前言 *Preface*

本书重点关注的是广义的开发 workflow，涵盖了开发者开发和维护项目时将会面临的三个阶段，分别是初始化、开发和部署。对于每个阶段，本书都详细解释了这一阶段的需求和概念，并且介绍了一个合适的工具来实现这些需求。

在现代 Web 开发中，新的工具层出不穷，但是寿命往往都很短。我写本书花了超过一年的时间，这在 JavaScript 界可能相当于过了十年。工具日新月异的变化对写书提出了巨大的挑战，但同时也提醒了我不要过于关注工具本身。因此，本书介绍的每个技术都可以轻松地替换为另一种合适的工具，而不会影响关键的概念。另一方面，本书挑选工具的标准不是工具的流行程度，而是工具的可维护性（sustainability）。本书介绍的工具都被广泛使用，且具有一定的用户基础和宽广的应用空间。正是因为这些特点，本书在几年后依然会有参考价值。

目标读者

本书的目标读者是想要在日常 workflow 中添加自动化流程和工具链的开发者。本书建立的工作流模板既适合初学者，也适合有经验的开发者。尽管市面上已经有了许多教程和博文来介绍这些工具，但大都只是单独介绍其中的某一个。而本书则把所有这些工具融合到一个 workflow 中，绝不仅仅只是介绍工具用法这些表面知识而已。

本书大纲

本书分为两部分，每一部分包括五章。

第一部分介绍一套全新前端开发 workflow 和相关的工具，不仅描述了如何用三个专门的工具来搭建一个 workflow，还介绍了前端开发 workflow 的一些总体构想。建议按照顺

序来读这一部分。

- ❑ 第1章展现了前端开发者日常面临的挑战，并且介绍了如何用三种类型的工具建立工作流以克服这些挑战。
- ❑ 第2章介绍了第一种工具，也就是构建工具。它提供了抽象接口来自动化处理底层代码的任务(task)。这一章的内容主要基于 Gulp。
- ❑ 第3章介绍依赖(dependency)和执行链(execution chain)。这一章会结合上一章的任务，把它们添加到一系列的执行中，以此来建立一个本地的开发环境。
- ❑ 第4章引入依赖管理的概念。这一章会介绍对于前端开发而言，扁平依赖树和嵌套依赖树之间的区别和优劣。这一章选择 Bower 作为依赖管理的工具。
- ❑ 第5章是前端工作流的最后一步。这一章会介绍如何通过脚手架工具 Yeoman，来复用第2章到第4章中搭建好的流程。

读完第一部分，读者就能够运用上述的三种工具来改进工作流，以适应自己的需求了。所有介绍的工具都使用 Node.js 作为运行时。刚刚接触 Node.js 的开发者可以参考附录来大致了解一下。

第一部分中介绍的 Gulp、Bower 和 Yeoman 都可以轻松地替换为其他工具。不过，第二部分就会深入探讨这三个工具的内在工作原理。其中每一章都可以提高工作流的效率，还会说明这些工具的独特和有用之处。

- ❑ 第6章会介绍如何利用 Gulp 的特性，包括增量编译(incremental build)和管道开关(pipeline switch)，来改变 Gulp 在不同开发环境中的输出结果。
- ❑ 第7章深入解释了 Gulp 的底层技术——文件对象流(file object stream)。使用这一技术可以在不同的输入/输出场景中复用相似的管道。
- ❑ 第8章介绍了 Gulp 的插件生态系统，并且告诉读者在选择新插件作为构建管道(build pipeline)时，如何避免冗余的或者质量不好的插件。此外还介绍了如何通过 Gulp 的 task 和 streaming API 来集成插件以外的工具。
- ❑ 第9章又回过来讲依赖管理。这一章介绍了模块(module)的概念，并且展示几个不同的模块规范之间的差异，模块能让我们更轻松地依赖集成到应用中。
- ❑ 第10章以脚手架工具(scaffolding tool) Yeoman 来结束全书。在 Yeoman sub-generator 的帮助下，项目的脚手架不仅在项目初期十分有用，并且会贯穿整个项目的始终。

第二部分的目的是让读者精通这些工具。尽管这些工具的示例都是有联系的，但

读者也可以跳着阅读分别来了解它们的概念和细节。

代码格式和下载

本书包含了很多示例的源代码，有时会加粗代码，来凸显和前几章代码的不同之处，比如向代码中加入新功能。

大多数情况下，原始的源代码都被格式化过。本书在代码中加入了换行和缩进来适应本书的版面。在极端情况下，偶尔还会使用续行符（line-continuation marker）。另外，当源代码在正文中出现时，会移除源代码中的注释。包含重要概念的代码会附上注释。

本书中示例的源代码可以在出版社的网站 www.manning.com/books/front-end-tooling-with-gulp-bower-and-yeoman 上下载。

本书中的示例也可以从 GitHub 网站 <https://github.com/frontend-tooling> 上下载。项目“sample-project-gulp”中的分支中包含了每一章的解决方案。

网上资源

- ❑ StackOverflow 上的 Gulp.js 标签 (<http://stackoverflow.com/questions/tagged/gulp>) 里有很多问题和回答。上面的网友都非常友好而且见多识广，能够很好地回答一些特殊的问题。
- ❑ Yeoman 网站 (<http://yeoman.io>) 不仅包含 Yeoman 自己的资源，也提供了关于其他工具的丰富的教程资源。
- ❑ 作者的博客上也有很多 Gulp 和 Yeoman 的教程：<https://fettblog.eu>。这些文章更加简明扼要，解决的都是更为具体的问题。

Acknowledgements 致谢

你可能觉得写书是一项孤独的工作，但事实却并非如此。

在所有帮助过我的人中，首先要提到的就是 Lesley Trites。作为我在 Manning 的策划编辑，她给了我非常有价值的反馈，是我写作时最棒的伙伴。她手把手地教我写作的技巧，如果没有她，本书根本不会像现在这个样子。

说到 Manning 的编辑们，我跟技术编辑 Johan Pretorius 和 Nick Watts 也合作得非常愉快！感谢你们检查我提出的想法。

还要感谢那些在我的写作过程中提出了很多有用反馈的读者们，他们是：Andy Knight、David DiMaria、Giancarlo Massari、Harinath Mallepally、Jason Gretz、Jeroen Benckhuijsen、Johan Pretorius、Mario Ruiz、Nikander 和 Margriet Bruggeman、Palak Mathur、Tanya Wilke、Unnikrishnan Kumar 和 Zorodzayi Mukuya。

Alexander Zaytsev、Lars Johansen 和 Jens Klinger，他们怀着对开源的高涨热情，在我写书的时候，一直都在改进我在 GitHub 上的 Gulp 示例。

我所有关于前端工作流的知识都是 Addy Osmani 教我的。我从 Sebastian Gierlinger 和 Blaine Bublitz 那里知道了什么是开源。

Thomas Pink 和 Thomas Heller 不仅是最棒的同事，而且还一直提出有价值的问题，让我改进构建工具技术。他们对技术改进的推动和现实工作的具体需求是本书内容来源的坚实基础。

还要感谢 “Working Draft” podcast 上的热心朋友，他们是 Christian Schaefer、Peter Kröner、Rodney Rehm、Anselm Hannemann 和 Hans-Christian Reinl，他们欢乐和富有挑战性的讨论让我不放过文中的每一个细节。同时还要感谢 “Klassenfahrt” 团队：Fabian

Beiner、Sven Wolfermann、Sebastian Golasch、Der Pepo、Robert Weber、Marc Hinse、Bianca Kastl、Joschi Kuphal、Marc Thiele、Khalil Lechelt、Frederic Hemberger、Tobias Baldauf、Tom Arnold 和 Maik Wagner。没有你们我绝不会取得今天的成就。

人们总是对作家有一种浪漫的幻想，觉得他们每周都在法国普罗旺斯的小木屋中写作。为了满足读者的幻想，我要感谢我的岳父岳母 Hans 和 Marianne，因为每隔一周的周末我都是在他们奥地利 Innviertel 的家中度过的。我还要感谢我的父母 Hans 和 Rosi 平时对我的照顾！感谢你们的支持！

最后，我最想感谢的人是最爱的妻子 Doris，她的耐心、爱意和关心对我来说无可替代。Doris，我欠你太多了。

about the cover illustration 关于原书封面插图

本书封面插图的标题为“穿着中世纪衣服的男人”。这幅插图的作者是 Paolo Mercuri (1804—1884), 选自“Costumes Historiques”。这是一本多卷历史服饰简史, 时间横跨 12 世纪到 15 世纪, 由 Camille Bonnard 整理和编辑, 并于 19 世纪 50 年代或 60 年代在巴黎出版。19 世纪的人们对异国情调的兴趣日益增加, 人们沉迷于这类收藏, 因为这可以帮助他们探索所生活的世界和很久以前的世界。

Mercuri 的这部历史藏书中丰富多彩的插图生动地展示了几百年前世界各地的城市和文化差异。在街头或农村, 仅仅通过服装, 就可以识别出人们的生活状态、贸易和职业。几个世纪以来, 着装习俗发生了变化。曾经的各个地区的服装的多样性, 现在都已经渐渐消失了。如今我们已经很难区别来自不同大陆的居民, 更不要说区别他们来自的城市和国家, 以及社会地位和职业了。也许如今文化多样性已经被更为多样的个人生活所取代——一种更加多样、节奏更快的科技生活。

在这样一个计算机书籍逐渐趋同的时代, Manning 利用几个世纪之前不同地域丰富多样的生活来制作图书封面, 借此赞美计算机行业的创造力和引领时代的精神。

目 录 Contents

译者序	1.3 用 Yeoman 搭建脚手架	13
自 序	1.3.1 Yeoman 作为脚手架工具	14
前 言	的优点	14
致 谢	1.3.2 什么是 generator	14
关于原书封面插图	1.4 依赖管理工具 Bower	15
	1.4.1 Bower 的优点	16
	1.4.2 Bower 的依赖树	17
	1.5 Gulp 流式构建系统	18
	1.5.1 Gulp 的优点	18
	1.5.2 构建管道	19
	1.6 总结	21
第一部分 现代 Web 应用 workflow	第 2 章 Gulp 入门	22
第 1 章 现代前端工作流的工具链	2.1 配置 Gulp	23
1.1 软件开发者的 workflow 和任务清单	2.1.1 Gulp 的构建块	23
1.1.1 初始化阶段的任务	2.1.2 Gulp 命令行接口	25
1.1.2 开发阶段的任务	2.1.3 安装本地 Gulp	26
1.1.3 部署阶段的任务	2.2 创建 Gulpfile	28
1.1.4 人为因素	2.2.1 Gulp 的“Hello World”	28
1.2 JavaScript 工具和新的 workflow	task	28
1.2.1 三类代码及其工具		
1.2.2 Node.js 中的 JavaScript 工具		

2.2.2	流的使用	31
2.2.3	Gulp 中的可读流和 可写流	32
2.3	使用 Gulp 插件处理 task	34
2.3.1	转换数据	34
2.3.2	改变文件结构	35
2.3.3	链式调用插件	37
2.4	总结	39

第 3 章 用 Gulp 配置本地开发环境

3.1	本地开发环境	42
3.1.1	给 task 分类	43
3.1.2	重置步骤: clean task	43
3.1.3	构建和迭代	45
3.2	初次构建步骤	45
3.2.1	依赖链和执行链	46
3.2.2	顺序和并行执行 task	47
3.2.3	依赖链的 task 执行函数	48
3.2.4	更新 Gulpfile	49
3.3	迭代步骤	52
3.3.1	监控过程	52
3.3.2	使用 Browsersync 建立按 需实时重载的服务器	53
3.3.3	更新 Gulpfile	55
3.4	总结	57

第 4 章 Bower 依赖管理

4.1	依赖管理基础	61
4.1.1	组件和依赖	61

4.1.2	依赖树	62
4.2	Bower 的扁平依赖树管理	64
4.2.1	安装和更新组件	65
4.2.2	安装有依赖的组件	66
4.2.3	解决依赖冲突	68
4.3	集成 Gulp	69
4.3.1	用 Gulp 把依赖集成到 应用中	70
4.3.2	将依赖集成到 Gulp 构建 流程中	72
4.4	总结	74

第 5 章 脚手架工具 Yeoman

5.1	Yeoman generator	76
5.1.1	generator 的构成	78
5.1.2	调用 generator	79
5.2	项目模板	81
5.2.1	固定文件	81
5.2.2	灵活文件	83
5.2.3	可选文件	83
5.2.4	可恢复文件	84
5.3	Yeoman 的组装流程	84
5.3.1	安装提示	85
5.3.2	写入文件	87
5.3.3	安装依赖	89
5.4	创建一个 generator	89
5.4.1	准备项目模板	90
5.4.2	创建组装指令	92
5.4.3	构建和测试	93
5.5	总结	94

第二部分 工具链的集成和扩展

第6章 不同环境下的 Gulp 97

6.1 增量构建和构建缓存 99

6.1.1 Gulp 内置功能 99

6.1.2 安装构建缓存 101

6.1.3 处理被删除的文件 104

6.2 通过 sourcemap 来调试 源文件 105

6.2.1 什么是 sourcemap 107

6.2.2 Gulp 内置的 sourcemap 108

6.2.3 sourcemap 插件 109

6.3 切换环境 111

6.3.1 noop() 111

6.3.2 dev() 和 prod() 选择函数 113

6.3.3 传递命令行参数 115

6.4 总结 116

第7章 使用流 117

7.1 处理不同类型的输入 119

7.1.1 穿插流 119

7.1.2 合并流 121

7.2 处理不同的输出 123

7.2.1 在 task 中根据参数 创建流 123

7.2.2 流数组 126

7.3 其他关于流的使用技巧 128

7.3.1 使用 snippet 来避免重复 代码 129

7.3.2 使用流队列来管理顺序 130

7.3.3 通过 Gulp filter 来修改流 的内容 131

7.4 总结 133

第8章 扩展 Gulp 134

8.1 Gulp 插件黑名单 135

8.1.1 理由：不是一个 Gulp 插件 136

8.1.2 理由：可以用别的东西 替代 137

8.1.3 理由：做的事情太多 138

8.2 集成其他基于流的工具 139

8.2.1 流、缓冲和 Vinyl 文件 对象 139

8.2.2 集成 Browserify 142

8.2.3 转换内容 145

8.3 集成基于 Promise 的工具 146

8.3.1 Promise 的用法 147

8.3.2 在 Gulp task 系统中使用 Promise 149

8.4 总结 151

第9章 创建模块和 Bower 组件 153

9.1 异步模块规范 155

9.1.1 AMD——为浏览器设计的 模块定义规范 155

9.1.2 重构到 AMD 158

9.2 CommonJS 和通用模块 规范 160

9.2.1 CommonJS 模块 160