

Meteor之父作序推荐，全栈开发必读

直击Web/移动应用开发痛点

通过实例详细讲解Meteor应用开发全流程

Meteor
in Action

Meteor

实战

[德] Stephan Hochhaus 著
Manuel Schoebel 著
杨学辉 译



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Meteor in Action

Meteor 实战



[德] Stephan Hochhaus 著
Manuel Schoebel 著
杨学辉 译

人民邮电出版社
北 京

图书在版编目 (CIP) 数据

Meteor实战 / (德) 霍赫豪斯 (Stephan Hochhaus),
(德) 施厄贝尔 (Manuel Schoebel) 著; 杨学辉译. —
北京: 人民邮电出版社, 2017.4
(图灵程序设计丛书)
ISBN 978-7-115-45017-3

I. ①M… II. ①霍… ②施… ③杨… III. ①JAVA语
言—程序设计 IV. ①TP312.8

中国版本图书馆CIP数据核字(2017)第042186号

内 容 提 要

本书秉承“实战”系列图书的一贯风格,以解决开发者实际问题为出发点,通过 Meteor 平台构建可扩展的高性能应用。全书共分为三部分、12章,详细介绍了如何用 Meteor 进行全栈开发,涵盖了 Meteor 栈的所有关键部分,涉及构成 Meteor 栈的各种组件和概念、响应式应用的基本模块和应用的构建与合理部署等。作者对 MongoDB、路由、包等进行了深入探讨,通过诸多实际案例,让读者全面掌握如何充分发挥 Meteor 在服务器端和可扩展性上的优势。

本书适合 Web 开发人员阅读。

-
- ◆ 著 [德] Stephan Hochhaus Manuel Schoebel
译 杨学辉
责任编辑 朱 巍
执行编辑 贺子娟 李 敏
责任印制 彭志环
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京昌平百善印刷厂印刷
- ◆ 开本: 800×1000 1/16
印张: 18.75
字数: 449千字 2017年4月第1版
印数: 1-3 500册 2017年4月北京第1次印刷
著作权合同登记号 图字: 01-2016-4631号
-

定价: 69.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广字第 8052 号

序

2011年,我和Geoff Schmidt、Nick Martin一起开始开发Meteor(流星),这是一个新的JavaScript应用平台。我们的计划是使JavaScript开发者能够在清晰的概念之上有条不紊地创建优秀的Web和移动应用。

JavaScript是一种非凡的技术。它最初是作为浏览器的脚本工具,现在已经成为一种无处不在的编程语言,比如在浏览器、移动设备以及云中。它得到了专家和初学者的青睐,而这是软件行业一个不寻常的组合。但JavaScript的生态系统是高度分散的,那些选择JavaScript的团队必须从无到有地构建和维护整个应用栈,需要在一些低级的技术任务上花费大量的时间,比如设计应用特定的WebSocket消息,而这些工作和他们实际的应用根本没有什么关系。

Meteor为那些希望创建现代化应用的JavaScript开发者提供了一个简单直接的解决方案,而《Meteor实战》则包含了为开始Meteor开发所需要知道的一切。它涵盖了Meteor栈的所有关键部分:从云上发布新的信息到每个在线用户的数据同步系统,在数据发生变化时重绘屏幕的响应式模板、事件和表单,同构的用户账户系统、路由、包和应用安全。

此外,《Meteor实战》涵盖了Meteor应用体系结构的基本知识。Meteor是一个全栈的响应式平台,这意味着它的各个部分——从数据库驱动程序到客户端模板引擎再到热码推送——一起工作,对数据的变化作出实时响应。Meteor是一种同构平台,也就是说,无论在哪里,你使用的JavaScript API会尽可能相同,比如在浏览器、移动设备以及云中。两位作者在书中通过一些清晰的示例解释了这些原则,并展示了如何在Meteor的开发过程中把它们整合在一起。

从我们发布最早的版本到现在,Stephan和Manuel一直活跃在Meteor社区。他们在无数的电子邮件和论坛主题中为Meteor作出了贡献,现在他们撰写了这本有趣的、让人易于接受的Meteor图书来分享他们的知识。

编码愉快!

Matt DeBergalis
Meteor开发小组创始人

前 言

2013年，一个朋友带我去参加鲁尔的第一次Meteor聚会，在那里我遇见了Manuel。那时，在企业工作多年的我正打算基于一个用PHP构建的Web平台开始创业。而Manuel向我们介绍了Meteor早期的情况，它解决了我面临的许多问题，并且使网络编程看起来轻而易举。这激发了我强烈的兴趣，我回到家就立即收集了有关这个新平台的更多资料。我把这些资料放在我的博客上，而且因为我刚刚读过一篇关于SEO的文章，其中建议使用夸大的言辞来吸引人们的注意，所以我大胆地宣称，这里提供了“学习Meteor.js的最佳资源”。我就是这么与Meteor结缘的。

2014年3月，Manning出版社联系我，问我是否有兴趣写一本关于前景光明的Meteor平台的书。他们看到了我的博客文章，并坚信我的相关知识够丰富，足以向其他开发者介绍这个平台。当然，我同意了。但是，尽管我收集了我认为最好的学习资源，我却不知道如何实际应用它们。我仍然停留在PHP的世界里。写一本书是学习一切Meteor相关知识的绝佳机会，所以我高兴地同意了，但我还是首先征询了Manuel的意见，问他是否愿意和我一起来写这本书。幸运的是，他同意了，所以我们便一起来解释这个新的平台。

写完这本书，我觉得Manuel的博学结合我自己的无知，帮助我们避免了针对读者做太多假设的陷阱，并在实用性和理论深度上取得了良好的平衡。请告诉我们这一策略是否真的有效。

我们觉得这本书的内容足以让你了解Meteor并基于此开发出精彩的应用。虽然我们不能解释Meteor工作中的每个细节，但我们希望这些基础知识可以帮助你更好地理解现有的文档、软件包和源代码。最终，Meteor和《Meteor实战》将能够帮助你把想法变成应用。请告诉我们你所开发的应用！你可以通过Twitter联系我们，也可以利用本书的GitHub仓库，还可以在这本书的作者在线论坛发表文章。我们很想听到你的反馈！

Stephan Hochhaus

致 谢

在这本书的封面上，你只会看到两个名字——Manuel和Stephan，但有很多好心人为这本书作出了贡献，如果不提到他们将是一种遗憾。首先，也是最重要的，要感谢Manning的工作人员，尤其是Robin De Jongh，他相信写一本关于Meteor的书是个好主意，还有Ozren Harlovic，他与我进行了第一次联系。也要感谢我们的编辑Sean Dennis和Dan Maharry，他们帮助我们吧晦涩的技术语言变成可理解的词汇和图表。感谢我们的文字编辑Liz Welch，他不得不帮我们修正大量用错的形近词。感谢我们的校对员Barbara Mirecki，以及其他许多和我们一起合作的Manning幕后工作人员。

Meteor社区对这本书的创作而言是非常重要的。我们要感谢所有早期（和现在）的开发者，他们使用Meteor、在网络上发表文章、开发包、拓展平台的界限。你们知道我说的是谁！

与Manning的编辑、生产和技术人员工作相当愉快，尽管他们不断地给我们施压，以使这本书做到最好。我们感激这种压力，这是值得的！

在写这本书的不同阶段，有很多人阅读了手稿，我们要感谢他们提供了宝贵的反馈信息：Carl Wolsey、Charlie Gaines、Cristian Antonioli、Daniel Anderson、Daniel Bertoi、David DiMaria、Dennis Hettema、John Griffiths、Jorge Bo、Keith Webster、Patrick Regan、Subhasis Ghosh、Tim Couger、Touko Vainio Kaila。

也要感谢我们的技术编辑Kostas Passidis，他保证了我们的技术解释是准确和可以理解的。感谢Al Krinker，他在本书付印前对最终的手稿做了彻底的技术审查。特别感谢Matt DeBergalis为本书作序。

万分感谢你们，我们的读者，特别是那些在这本书只有几章的时候就相信本书并加入本书早期预览计划的人。你们的反馈、兴趣和鼓励让我们得以持续前进！

Stephan Hochhaus

我想感谢Said Seihoub让我去参加那次Meteor聚会。没有他，我将不会写这本书。也非常感谢Manuel，当我遇到问题时，他总是能告诉我答案。写作是一项寂寞的事业，所以也要感谢Meteor的即时聊天频道，在我拖延时陪伴着我。当然了，要是没有你们，这本书可能在2014年就已经出版了！

也要感谢Anton Bruckner、Johann Sebastian Bach、Joss Whedon和Terry Pratchett创造了合适的

工作氛围。最后，我要感谢我的家人，他们对我表现出了很大的耐心，每个月我都告诉他们我写完了一章，结果下个星期又回到这一章重写，并再次“完成”它。

Manuel Schoebel

写一本书所要付出的努力比我想象的多得多，但这是一个伟大的旅程，它帮助我更深入地去挖掘Meteor的细节。谢谢Stephan邀请我共同写作这本书。一如既往地，和你一起工作非常愉快。

在这本书的写作过程中，我同时做着另一件事——创办了自己的公司，这也占据了我大量的时间。Christina，如果没有你的宽容、耐心和支持，这两件事我都不可能做好，所以感谢你，你太伟大了！

有一个家庭时刻在背后支持你是一种难得的幸福，不是每个人都有这样的福分。我深深明白这一点并万分感激——在事情变得困难时，你们总是能够带给我内心的宁静。

最后，我感谢每天都致力于让Web更加精彩的所有人。这不仅包括来自Meteor本身的伙伴，而且也包括开发新的软件包、参加聚会或是刚刚开始学习如何把自己的想法变成Web应用的每个人。网络给了我们学习、探索、工作和娱乐的自由，比我们过去所知道的自由还要多。这是一个甚至可以让你谋生的游乐场。我们邀请你来和我们一起玩！

关于本书

从一些有经验的开发人员那里，你常听到这样一句话：“开发应用并不像发射火箭那么高深。”虽然开发应用不像把人类送上太空那么复杂，但对新手来说它可能让人望而生畏。要把你的应用放在Web上通常需要大量的工具和服务器组件，更不要说移动设备了。Meteor的目标是成为一个游戏规则改变者。Meteor的创建者之一Nick Martin这样说道：

通过Meteor，我们希望Web应用的开发能够大众化，可以让任何人在任何地方创建应用。^①

我们已经看到，一些稍有HTML和CSS基础的人就可以通过Meteor在一天之内把他们的想法变成代码。因此，我们相信它会使开发变得更容易，甚至会让那些从来不认为自己是开发者的人开始开发应用。

除非有一个好老师，否则你可能需要大半天的时间来了解Meteor平台。这就是《Meteor实战》的用武之地。它是你的私人教师，将引导你学习创建应用的各个主要方面，不管你想开发Web应用还是移动应用。最终，你将能够把自己的想法转化为代码。如果你在了解Meteor平台之前做过这件事，就会惊讶于Meteor快速地解决了一些最常见的问题。

《Meteor实战》的主要读者有两类：一类是想把技能扩展到服务器端的前端开发者，另一类是具有服务器开发背景的、想转变为JavaScript全栈开发者的Java、Ruby或PHP开发者。这本书不是为初学者撰写的，我们希望你以前已经开发过（或至少尝试开发过）一些Web应用。

和所有仍在使用的工具一样，Meteor一直在不断变化和发展。我们很小心地在书中教授这个平台的基本原理，以帮助你奠定良好的基础。我们已确认，后面章节中描述的所有功能都可以在版本1.1上很好地工作。

路线图

本书分为三个部分。

第一部分概述了Meteor平台，介绍了构成Meteor栈的各种组件和概念。第1章概要介绍了Node.js、MongoDB、同构性及响应性，然后你将在第2章中创建你的第一个Meteor应用。

第二部分介绍了响应式应用的基本模块。每一章侧重于应用开发的一个方面。第3章介绍了模板，第4章介绍了如何使用数据和执行CRUD操作，第5章结合这两个方面强调了构建响应式界

^① <http://blog.heavybit.com/blog/2014/04/01/meteor>。

面时一些需要着重考虑的因素，第6章介绍了一种通过引入用户相关的功能来保护应用的方法。第7章介绍了如何取代Meteor默认使用的自动数据发布机制，涵盖了Meteor发布和订阅的概念，以及如何使用方法来实现另一个层次的安全。客户端和服务端上的路由操作都使用了流行的Iron.Router库，这将在第8章中讨论。第9章教你如何使用包来扩展Meteor的核心功能，你可以使用现有的Isopack、npm包或编写自己的包。第10章是这一部分的最后一章，包含了用于异步操作的服务器端方法、外部API访问和文件上传等内容。

第三部分则更进一步，讨论了应用的构建和合理部署。第11章解释了Meteor的构建系统、代码调试，以及如何将你的代码转变为Web和移动应用。最后的第12章则涉及把Meteor应用到生产环境的各个方面。

本书后面有三个附录。附录A涵盖了所有支持平台上Meteor的安装过程。附录B揭示了MongoDB的架构以及用来实现高可用性的相关组件，还介绍了如何设置最新操作日志（oplog tailing），这是Meteor实现应用可伸缩性背后的重要技术。附录C教你如何设置反向代理以实现多个Meteor服务器之间的负载均衡，提供静态内容服务并启用SSL。

先决条件

要想从这本书中获得最大收益，你需要在系统上安装Meteor。安装Meteor的方法可以在附录A中找到，也可以在Meteor的官方网站（<http://meteor.com>）上找到。

在本书中，我们假定你至少对HTML、CSS和JavaScript有基本的了解。你应该知道如何使用对象，并使用过回调函数。对数据库的工作方式有基本的了解是有帮助的，但不是必需的。阅读本书时，你不需要有任何使用服务器端JavaScript甚至是Node.js的经验。

代码

本书中的所有代码都可从Manning网站下载：<http://www.manning.com/books/meteor-in-action>。你也可以在GitHub上找到它：<http://www.github.com/meteorinaction>。

为了便于学习，每一章都建了一个单独的Git仓库。因为不是所有的代码都印在书中，所以我们为每个仓库添加了标签，方便你在迷茫的时候找到方向。例如，当你开始第2章时，可以参考标签为begin的代码来查看起始代码。如果你想跳过前面的部分，直接查看服务器启动时添加的夹具代码，可以查看标签为listing-2.9的代码。

作者在线

购买英文版的读者可免费访问Manning出版社的专有论坛^①，并可以在那里发表关于本书的评论，咨询技术问题，获得作者和其他用户的帮助。要访问和订阅论坛，可在浏览器中键入这个地

^① 中文版意见和勘误请提交到图灵社区：www.it-ebooks.com.cn/book/1837。

址：<http://www.manning.com/books/meteor-in-action>。这个页面提供了一些基本信息，包括注册之后如何进入论坛、它可以提供什么样的帮助以及论坛上的行为准则等。

Manning承诺为读者之间以及读者和作者之间的有效交流提供一个平台。但这并不意味着作者会有任何特定程度的参与，他们对本书论坛的贡献是自愿的（无报酬的）。我们建议你试着问他们一些具有挑战性的问题，以免他们的兴趣转移到别的地方！

只要本书还在销售，作者在线论坛和以前所讨论内容的归档文件就都可以在出版社的网站上访问到。

关于作者

Stephan Hochhaus偶然在Perl语言中找到了自己的定位，然后开始了他的开发职业生涯。他为大型企业开发可扩展的Web解决方案，学习钻研PHP、C#甚至Java。工作多年以后，他于2013年开始创业，为中小企业开发Web应用。自从遇见了Meteor，他觉得自己今后就应该使用JavaScript。Stephan也为引入Scrum^①或进行持续交付开发的团队提供咨询服务。他拥有波鸿鲁尔大学的语言学和社会心理学硕士学位，能够熟练地使用正则表达式。

Manuel Schoebel拥有杜伊斯堡—埃森大学（埃森校区）商业信息学文凭，专注于Web创业。Manuel花费了大量时间辅导创始人、开发MVP，甚至创立了多家公司。他从2012年开始接触Meteor，当时这个平台还处于起步阶段。他写了一些有价值的博客文章，很快就成为了Meteor社区中著名的专家。自2013年以来，Manuel在他的项目中就只使用Meteor。

Manuel和Stephan一起组织了德国科隆和鲁尔地区的Meteor聚会，把Meteor开发者聚在一起交流思想、探讨新的发展。

关于书名

通过介绍、概述以及操作实例，“实战”系列图书的目的是帮助学习和记忆。根据认知科学的研究，人们记住的东西是他们在自我激励的探索中发现的东西。

虽然在Manning没有一个人是认知科学家，但我们相信，要想使学到的东西终生不忘，必须经过探索、把玩、复述所学这几个阶段。只有经过积极的探索，人们才能理解并记住新事物，也就是掌握新的东西。人类在实战中学习，而“实战”系列图书的一个重要特点就是实例驱动。它鼓励读者尝试新的事物、把玩新的代码、探索新的想法。

给这本书起“实战”这样的书名还有另一个更普遍的原因：我们的读者都很忙。他们用书来完成一项工作或解决一个问题。他们需要的是这样的书：可以让他们轻松地跳进跳出，只在需要的时候学习想学的。他们需要有助于行动的书。这个系列的图书就是为这样的读者准备的。

^① Scrum是迭代式增量软件开发过程，通常用于敏捷软件开发。——译者注

关于封面图片

本书的封面图片是一个“挑夫”——一个小工、水手或码头工人，看着似乎很好斗。插图取自Sylvain Maréchal在法国出版的四卷区域服饰习俗概要的19世纪版。其中每幅插图都是手工精心绘制和着色的。Maréchal丰富的收藏生动地提醒我们，仅仅在200年前，世界上的城镇和区域文化是多么不同。各个区域彼此孤立，人们说着不同的方言和语言。在街头或在农村，仅通过衣着就很容易确定他们生活的地方、从事的行业和社会地位。

但是从那时起，服饰已经改变，地区的多样性也已经慢慢消失了。现在人们已很难分辨来自不同大陆的居民，更不用说不同城镇或区域的居民了。也许文化的多样性已经被我们多样化的个人生活——当然是更加多样化和快节奏的高科技生活——取代了。

如今，很难说出一本计算机书和另一本的区别。Manning利用图书的封面来颂扬计算机事业的创造性和原创性，这些封面都反映了两个世纪以前各地区丰富多彩的生活，它们通过Maréchal的图片重新获得生机。

目 录

第一部分 看，一颗流星！

第 1 章 构建应用程序的更好方式	2
1.1 Meteor 简介	3
1.1.1 Meteor 背后的故事	3
1.1.2 Meteor 栈	4
1.1.3 同构框架：全栈式 JavaScript	5
1.1.4 在浏览器中处理：在分布式平 台上运行	7
1.1.5 响应式编程	8
1.2 Meteor 的工作原理	10
1.2.1 核心项目	11
1.2.2 Isobuild 和 CLI 工具	13
1.2.3 客户端代码和服务器端代码	14
1.3 优势和劣势	15
1.3.1 使用 Meteor 的好处	16
1.3.2 使用 Meteor 时的挑战	17
1.4 创建新的应用程序	19
1.4.1 创建新项目	19
1.4.2 启动应用	20
1.5 剖析默认项目	21
1.5.1 helloWorld.css	21
1.5.2 helloWorld.html	21
1.5.3 helloWorld.js	22
1.6 总结	24
第 2 章 我的冰箱：一个响应式游戏	25
2.1 应用概述	25
2.2 初始设置	26
2.3 创建布局	27
2.3.1 设置样式	27

2.3.2 添加模板	29
2.4 向数据库中实时添加内容	31
2.4.1 在数据库中存储物品	31
2.4.2 将数据连接到模板	33
2.4.3 添加一组预定义的物品	35
2.5 把物品放进冰箱里	36
2.5.1 为项目添加 jQuery UI	36
2.5.2 为物品定义拖放目标	37
2.5.3 允许物品被拖动	39
2.6 部署应用到 meteor.com 并使用它	40
2.7 总结	40

第二部分 3, 2, 1——撞击！

第 3 章 模板	42
3.1 模板介绍	42
3.2 使用模板	44
3.2.1 Blaze 引擎	44
3.2.2 组织模板文件	45
3.3 创建动态 HTML 模板	45
3.3.1 双重和三重大括号标签 （表达式）	46
3.3.2 包含标签（局部模板）	48
3.3.3 块标签	50
3.3.4 辅助函数	53
3.4 处理事件	58
3.4.1 模板的事件映射	58
3.4.2 事件传播	59
3.4.3 阻止浏览器的默认行为	60
3.5 模板生命周期	61
3.6 总结	63

第4章 数据	64	第6章 用户、认证和权限	110
4.1 Meteor的默认数据源	66	6.1 将用户加入应用	111
4.1.1 什么使数据源具有响应性	67	6.1.1 添加密码认证	111
4.1.2 如何将响应式数据连接到 函数	68	6.1.2 注册和密码重置	113
4.2 构建房屋保姆应用	69	6.1.3 设置邮件	117
4.2.1 设置模板	71	6.2 使用 OAuth 认证用户	120
4.2.2 连接到数据库并声明集合	71	6.2.1 OAuth 介绍	121
4.3 Session 对象	72	6.2.2 整合 Facebook 认证	122
4.3.1 Session 对象简介	73	6.2.3 集成其他的 OAuth 提供者	126
4.3.2 使用 Session 存储选定的 下拉值	73	6.3 管理用户权限、角色和组	127
4.3.3 使用 Tracker.autorun 创建 响应性上下文	75	6.4 总结	130
4.4 MongoDB 集合	76	第7章 数据交换	131
4.4.1 在 MongoDB 中查询文档	77	7.1 发布和订阅	132
4.4.2 Meteor 的集合	78	7.1.1 publish() 和 subscribe()	133
4.4.3 初始化集合	78	7.1.2 全局订阅	134
4.4.4 查询集合	79	7.1.3 模板级订阅	135
4.4.5 在模板中显示集合数据	80	7.1.4 参数化订阅	136
4.4.6 在集合中更新数据	83	7.1.5 向客户端独有的集合发布汇总 数据	138
4.4.7 向集合中插入新数据	85	7.1.6 将聚合发布变成响应式 数据源	141
4.4.8 从集合中删除数据	87	7.1.7 通过用户 ID 限制数据 可见性	143
4.5 总结	88	7.2 Meteor 的方法	145
第5章 全响应式编辑	89	7.2.1 删除 insecure 包	146
5.1 响应式编辑的工作流程	89	7.2.2 使用方法将数据写入集合	147
5.2 响应式前端与 DOM 操作	90	7.3 总结	150
5.3 在本地集合中进行临时更改	94	第8章 路由	151
5.4 在表单中显示集合数据	97	8.1 Web 应用中的路由	151
5.5 使用本地集合进行响应式更新	100	8.2 客户端路由	154
5.5.1 houseForm 模板的事件 映射	101	8.2.1 添加 Iron.Router	155
5.5.2 plantFieldset 模板的事件 映射	102	8.2.2 创建第一个路由	155
5.6 实现一个简单的通知系统	105	8.2.3 基于路由定义布局	157
5.6.1 添加通知模板	106	8.2.4 根据路由设置数据上下文	161
5.6.2 添加状态属性	106	8.2.5 使用 Iron.Router 订阅 数据	163
5.6.3 使用会话变量触发通知	107	8.3 高级的路由方法	166
5.7 总结	109	8.3.1 使用命名路由和链接辅助 函数	166

8.3.2 让活动路由有更好的导航 链接.....	167	10.3.1 使用 HTTP 包进行 RESTful 调用.....	208
8.3.3 等待外部库加载.....	168	10.3.2 使用同步方法查询 API.....	208
8.3.4 将路由组织为控制器.....	169	10.3.3 使用异步方法调用 API.....	209
8.3.5 使用钩子扩展路由过程.....	171	10.4 将文件上传到集合.....	211
8.3.6 创建 Iron.Router 插件.....	172	10.5 总结.....	215
8.4 用于 REST API 的服务器端路由.....	173		
8.5 总结.....	175		
第 9 章 包.....	176	第三部分 走出陨石坑	
9.1 所有应用的基础.....	176	第 11 章 构建和调试.....	218
9.2 使用 Isopack.....	177	11.1 Meteor 的构建过程.....	218
9.2.1 版本求解器和有语义的 版本号.....	178	11.1.1 构建阶段.....	220
9.2.2 查找包.....	179	11.1.2 使用 --production 选项 运行.....	223
9.2.3 添加和删除 Isopack.....	182	11.1.3 加载顺序.....	223
9.2.4 更新包.....	183	11.1.4 通过包添加构建阶段.....	226
9.3 使用 npm 包.....	183	11.1.5 添加自定义构建阶段.....	227
9.4 创建 Isopack.....	184	11.2 访问正在运行的应用.....	229
9.4.1 创建包.....	185	11.2.1 使用交互式服务器外壳.....	229
9.4.2 声明包的元数据.....	186	11.2.2 使用 node-inspector 进行调试.....	230
9.4.3 添加包的功能.....	188	11.3 创建浏览器应用.....	232
9.4.4 使用 tinystest 测试 Isopack.....	191	11.3.1 使用 Meteor.settings 进行应用配置.....	232
9.4.5 发布.....	194	11.3.2 构建 Meteor 项目.....	233
9.5 总结.....	195	11.4 创建移动应用.....	235
第 10 章 高级服务器方法.....	196	11.4.1 使用 Cordova 的混合应用.....	235
10.1 再次介绍 Node.js.....	196	11.4.2 加入移动平台.....	236
10.1.1 同步代码.....	197	11.4.3 配置移动应用.....	238
10.1.2 异步代码.....	198	11.4.4 添加移动功能.....	240
10.2 使用纤维的异步功能.....	200	11.5 总结.....	242
10.2.1 将多任务引入事件循环.....	200	第 12 章 开始生产.....	243
10.2.2 使用 wrapAsync 为纤维 绑定回调函数.....	203	12.1 准备生产.....	243
10.2.3 为单个客户端解除方法调用 阻塞.....	204	12.1.1 使用版本控制.....	243
10.2.4 使用 bindEnvironment 创建纤维.....	206	12.1.2 功能测试: Velocity 框架.....	245
10.3 整合外部 API.....	207	12.1.3 估算和测试负载.....	247
		12.1.4 服务器管理.....	249
		12.1.5 清单.....	250
		12.2 安装和部署.....	250

12.2.1 最简单的部署： meteor.com.....	251	12.4.1 使用冗余的主动-被动高 可用性.....	262
12.2.2 无所不包的主机：云 供应商.....	253	12.4.2 单组件部署.....	263
12.2.3 最灵活的方式：手动 设置.....	256	12.4.3 冗余和负载均衡.....	263
12.3 将各部分连接起来.....	259	12.4.4 绝对可用性.....	265
12.3.1 环境变量.....	260	12.5 总结.....	266
12.3.2 Meteor 和 MongoDB 的 连接.....	260	附录 A 安装 Meteor.....	267
12.4 扩展策略.....	261	附录 B MongoDB 剖析.....	271
		附录 C 设置 nginx.....	278

Part 1

第一部分

看，一颗流星！

第一部分概述Meteor（流星）平台的各个组成部分以及它们是如何协作的。我们将介绍Node.js、MongoDB以及响应式编程的概念。你将在第1章中全面理解Meteor平台的整个技术栈，在第2章中构建你的第一个Meteor应用。

本章内容

- ❑ Meteor背后的故事
- ❑ Meteor栈
- ❑ 全栈式JavaScript、响应和分布式平台
- ❑ Meteor平台的核心部件
- ❑ 使用Meteor的优点和缺点
- ❑ Meteor应用解析

正如我们所知道的那样，流星（meteor）以改变生活著称。它们能使恐龙灭绝，或者迫使布鲁斯·威利斯为拯救人类而牺牲生命^①。本书讲述的是一颗影响Web开发的流星，但它不会威胁和毁灭任何东西。相反，它将提供一种更好的方式来构建应用程序。Meteor借用几个现有的工具和库，将它们与新的思想以及新的库、标准和服务结合起来，并将它们捆绑在一起，为开发易用的Web和移动应用提供一个完整的生态系统。

Meteor是基于MEAN栈^②的开源应用开发平台。该平台的客户端和服务端使用一致的JavaScript API。它专注于实时的响应式应用、快速原型开发和代码重用。

作为开发者，你知道一旦打开浏览器的源码视图，所有Web应用程序都只是HTML、CSS和JavaScript的组合。像Google、Twitter和Facebook这样的巨头在网站开发上取得了令人印象深刻的成果，它们使得网站看起来更像是桌面应用。Google Maps的流畅和Facebook Messenger的直接，使得用户对于互联网上所有的网站具有更高的期望。Meteor使你能够满足这些高期望，因为它提供了所有基础设施功能，如数据订阅和用户处理，让你专注于实现业务功能。

本章将告诉你Meteor如何让开发变得更容易。我们先大致了解它的来历，然后将重点放在它由什么组成以及如何使用它来快速构建应用。

^① 来自布鲁斯·威利斯主演的电影《绝世天劫》，影片中布鲁斯·威利斯为拯救人类需要阻止巨型陨石撞击地球。

——译者注

^② MEAN栈是指所有构建于MongoDB、Node.js、Angular和Express.js之上的应用。MEAN栈有几种不同的形式，比如MEEN栈，它指的是MongoDB、Ember.js、Express和Node.js。有时这个词用于泛指那些运行于Node.js之上并使用NoSQL数据库的任何框架。