

TURING

图灵程序设计丛书

MANNING

Node.js in Action
Second Edition

Node.js

实战 (第2版)

[英] 亚历克斯·杨

布拉德利·马克

麦克·坎特伦

蒂姆·奥克斯利

马克·哈特

T.J. 霍洛瓦丘

内森·拉伊利赫

著

[美]

吴海星 译

Node.js 核心框架贡献者力作

展示 Node 核心技巧



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

丛书

Node.js in Action Second Edition

Node.js

实战

(第2版)

[英] 亚历克斯·杨

布拉德利·马克

麦克·坎特伦

蒂姆·奥克斯利

[美]

马克·哈特

T.J. 霍洛瓦丘

内森·拉伊利赫

著



吴海星 译

人民邮电出版社
北 京

图书在版编目 (C I P) 数据

Node.js 实战 / (英) 亚历克斯·杨 (Alex Young)
等著 ; 吴海星译. — 2版. — 北京 : 人民邮电出版社,
2018. 8

(图灵程序设计丛书)
ISBN 978-7-115-48730-8

I. ①N… II. ①亚… ②吴… III. ①JAVA语言—程序
设计 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第138685号

内 容 提 要

本书是 Node.js 的实战教程,涵盖了为开发产品级 Node 应用程序所需要的一切特性、技巧以及相关理念。从搭建 Node 开发环境,到一些简单的演示程序,到开发复杂应用程序所必不可少的异步编程。第2版介绍了全栈开发者所需的全部技术,包括前端构建系统、选择 Web 框架、在 Node 中与数据库的交互、编写测试和部署 Web 程序,等等。

本书适合 Web 开发人员阅读。

-
- ◆ 著 [英] 亚历克斯·杨 [美] 布拉德利·马克
[美] 麦克·坎特伦 [美] 蒂姆·奥克斯利
[美] 马克·哈特 [美] T.J. 霍洛瓦丘
[美] 内森·拉伊利赫
- 译 吴海星
- 责任编辑 朱 巍
- 责任印制 周昇亮
- ◆ 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>
北京鑫正大印刷有限公司印刷
- ◆ 开本: 800×1000 1/16
印张: 20.5
字数: 485千字 2018年8月第2版
印数: 13 801 - 17 300册 2018年8月北京第1次印刷
- 著作权合同登记号 图字: 01-2017-7475号
-

定价: 89.00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: 京东工商广登字 20170147 号

站在巨人的肩上
Standing on Shoulders of Giants



iTuring.cn

站在巨人的肩上
Standing on Shoulders of Giants



iTuring.cn

版 权 声 明

Original English language edition, entitled *Node.js in Action Second Edition* by Alex Young, Bradley Meck, Mike Cantelon, Tim Oxley, Marc Harter, T.J. Holowaychuk, Nathan Rajlich, published by Manning Publications. 178 South Hill Drive, Westampton, NJ 08060 USA. Copyright © 2017 by Manning Publications.

Simplified Chinese-language edition copyright © 2018 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Manning Publications 授权人民邮电出版社独家出版。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。

版权所有，侵权必究。

第 1 版赞誉

“内容循序渐进，由浅入深。”

——Node.js 项目负责人 Isaac Z. Schlueter

“Node 及 Node.js 生态系统的权威指南。”

——Kevin Baister, 1KB Software Solutions 公司

“书中的示例有非常强烈的现实意义（也很有趣）。”

——ÀlexMadurell, PolymediaSpA

“非常引人入胜……能让你非常快速地进入状态。”

——Gary Ewan Park, 霍尼韦尔公司

“由编写 Node.js 代码的人写出来的宝贵资源。”

——Brian Falk, NodeLingo, GoChime

前 言

《Node.js 实战》的第 1 版出版之后发生了很多事情，io.js 问世，治理模型也发生了翻天覆地的变化。Node 的包管理器孵化出了一家成功的新公司——npm，Babel 和 Electron 等技术也改变了 Node 开发。

虽然 Node 的核心库变化不大，但 JavaScript 变了，大多数开发人员都用上了 ES2015 的功能特性，所以我们改写了上一版中的所有代码，用上了箭头函数、常量和解构。因为 Node 的库和自带的工具看起来仍然和 4.x 之前的版本差不多，所以我们在这一版的更新中瞄准了社区。

为了体现 Node 开发人员在实际工作中面临的问题，本书在结构上进行了调整。Express 和 Connect 的分量轻了，涉及的技术范围广了。书中介绍了全栈开发者所需的全部技术，包括前端构建系统、选择 Web 框架、在 Node 中与数据库的交互、编写测试和部署 Web 程序。

除了 Web 开发，本书还有编写命令行程序和 Electron 桌面程序的章节，让你充分利用自己的 Node 和 JavaScript 技能。

本书不仅要向你介绍 Node 和它的生态系统，还想尽可能让你了解那些影响 Node 发展的背景知识，比如一般在 Node 和 JavaScript 书籍中并不介绍的 Unix 哲学和如何正确、安全地使用数据库。希望这些知识能拓宽你的眼界，加深你对 Node 和 JavaScript 的理解，帮你在面临新的问题时找到解决办法。

致 谢

首先要感谢本书上一版的作者们，他们做出了巨大贡献：Mike Cantelon、Marc Harter、T.J. Holowaychuk 和 Nathan Rajlich。还要感谢 Manning 的团队，如果没有他们的鼓励，这一版也不会问世。感谢我的策划编辑 Cynthia Kane，在更新原内容的漫长过程中让我保持专注。如果没有 Doug Warren 详尽的技术校对，本书及其中代码的正确率恐怕连现在的一半都不到。最后要感谢在写作及开发过程中提供反馈的评审人员：Austin King、Carl Hope、Chris Salch、Christopher Reed、Dale Francis、Hafiz Waheedud din、HarinathMallepally、Jeff Smith、Marc-Philippe Huget、Matthew Bertoni、Philippe Charrière、Randy Kamradt、Sander Rossel、Scott Dierbeck 和 William Wheeler。

关于本书

本书第 1 版重点介绍了如何用 Web 框架 Connect 和 Express 开发 Web 程序。第 2 版则根据 Node 开发的变化做了调整。我们会介绍前端构建系统、流行的 Node Web 框架，以及如何用 Express 从头开始搭建 Web 程序，还会讲到自动化测试和 Node Web 程序的部署。

因为用 Node 做的命令行开发者工具和用 Electron 做的桌面端程序越来越多，所以本书专门用了两章的篇幅分别介绍这两块内容。

本书假定你熟悉基本的编程概念。但考虑到有些开发人员还没有接触过新的 JavaScript，所以第 1 章将会介绍 JavaScript 和 ES2015。

路线图

本书分为三部分。

第一部分介绍 Node.js，讲解用它进行开发所需的基础技术。第 1 章介绍了 JavaScript 和 Node 的特性，通过示例代码一步步进行讲解。第 2 章介绍了基本的 Node.js 编程概念。第 3 章完整地演示了如何从头开始搭建一个 Web 程序。

第二部分重点介绍 Web 开发，内容最多，篇幅也最长。第 4 章是前端构建系统的揭秘。如果你在项目中用到过 Webpack 或 Gulp，但并没有真正掌握它们，那么可以学习一下这一章的内容。第 5 章介绍了 Node 中最流行的服务器端框架。第 6 章详细介绍了 Connect 和 Express。第 7 章是模板语言，它可以提升服务端代码的编写效率。大多数 Web 程序都需要数据库，所以第 8 章介绍了很多种可以用在 Node 中的数据库，关系型和 NoSQL 都有涉及。第 9 章和第 10 章讲了测试和部署，包括云端部署。

第三部分是 Web 程序开发之外的内容。第 11 章讲了如何用 Node 搭建命令程序，创建出开发人员熟悉的文字界面。如果你喜欢用 Node 搭建像 Atom 一样的桌面程序，可以看看介绍 Electron 的第 12 章。

本书还有三个附录。附录 A 讲了如何在 macOS 和 Windows 上安装 Node，附录 B 详细介绍了如何实现网络内容抓取，附录 C 介绍了 Connect 的官方中间件组件。

编码规范及下载

书中的代码遵循通用 JavaScript 规范。缩进用空格，不用制表符。尽量不要让一行代码的长

度超过 80 个字符。很多代码清单中都加了注释，指出了其中的关键概念。

每行一条语句，简单语句后面加分号。代码块放在大括号中，左括号放在代码块开始行的末尾处，右括号的缩进跟代码块开始行的缩进保持一致，在垂直方向上对齐。

书中示例的源码请至图灵社区本书主页 <http://www.ituring.com.cn/book/1993> 随书下载处下载。

本书论坛

购买了英文版的读者可以免费访问 Manning 出版社运营的专享论坛，你可以在那里发表对图书的评论，提出技术问题，寻求作者和其他读者的帮助。

Manning 的初衷是为读者间、读者与作者间提供一个交流场所。作者完全可以根据个人意愿进行参与，在论坛上所做的贡献是没有报酬的。所以我们建议你尽可能提出一些有挑战性的问题，以激发作者的兴趣！只要书还在发行，出版社的网站上就会有关于书的论坛和之前讨论过的内容的归档。

读者也可登录图灵社区本书主页 <http://www.ituring.com.cn/book/1993> 提交反馈意见和勘误。

电子书

扫描如下二维码，即可购买本书电子版。



关于封面图片

本书封面上的画像标题为“城镇里的男人”，摘自 19 世纪法国出版的沙利文·马雷夏尔（Sylvain Maréchal）四卷本的地域服饰风俗摘要。其中每幅插图都是手工精心绘制并上色的。马雷夏尔这套书展示的丰富服饰，令我们强烈感受到 2000 年前乡村与城镇的巨大文化差异。不同地域的人山水阻隔，语言不通。无论奔走于街巷，还是驻足于乡间，通过他们的服饰，一眼就能看出他们的生活场所、职业，以及生活境况。

时过境迁，书中描绘的那些区域性服饰差异如今已经不复存在。即使是不同国家，都很难再看出人们着装的区别，再不必说城镇和乡村了。或许，我们今天多姿多彩的人生，正是从前那些文化差异的体现。只不过，如今的生活更加多元，而且技术环境下的生活节奏也更快了。

今时今日，计算机图书层出不穷，Manning 就以马雷夏尔这套书中多样性的图片，来表达对 IT 行业日新月异的发明与创造的赞美。

目 录

第一部分 Node 基础知识介绍

第 1 章 欢迎进入 Node.js 的世界 2

- 1.1 一个典型的 Node Web 应用程序 2
 - 1.1.1 非阻塞 I/O 3
 - 1.1.2 事件轮询 4
- 1.2 ES2015、Node 和 V8 5
 - 1.2.1 Node 与 V8 7
 - 1.2.2 使用特性组 8
 - 1.2.3 了解 Node 的发布计划 8
- 1.3 安装 Node 8
- 1.4 Node 自带的工具 9
 - 1.4.1 npm 9
 - 1.4.2 核心模块 10
 - 1.4.3 调试器 11
- 1.5 三种主流的 Node 程序 12
 - 1.5.1 Web 应用程序 12
 - 1.5.2 命令行工具和后台程序 13
 - 1.5.3 桌面程序 14
 - 1.5.4 适合 Node 的应用程序 14
- 1.6 总结 15

第 2 章 Node 编程基础 16

- 2.1 Node 功能的组织及重用 16
- 2.2 开始一个新的 Node 项目 18
- 2.3 用 `module.exports` 微调模块的创建 20
- 2.4 用 `node_modules` 重用模块 22
- 2.5 注意事项 23
- 2.6 使用异步编程技术 24
- 2.7 用回调处理一次性事件 25
- 2.8 用事件发射器处理重复性事件 28

- 2.8.1 事件发射器示例 28
- 2.8.2 响应只应该发生一次的事件 29
- 2.8.3 创建事件发射器：一个
PUB/SUB 的例子 29
- 2.8.4 扩展事件监听器：文件监视器 32
- 2.9 异步开发的难题 34
- 2.10 异步逻辑的顺序化 35
- 2.11 何时使用串行流程控制 36
- 2.12 实现串行化流程控制 37
- 2.13 实现并行化流程控制 39
- 2.14 利用社区里的工具 41
- 2.15 总结 43

第 3 章 Node Web 程序是什么 44

- 3.1 了解 Node Web 程序的结构 45
 - 3.1.1 开始一个新的 Web 程序 45
 - 3.1.2 跟其他平台比一比 47
 - 3.1.3 然后呢 47
- 3.2 搭建一个 RESTful Web 服务 47
- 3.3 添加数据库 50
 - 3.3.1 制作自己的模型 API 51
 - 3.3.2 让文章可读并把它存起来 53
- 3.4 添加用户界面 54
 - 3.4.1 支持多种格式 54
 - 3.4.2 渲染模板 55
 - 3.4.3 用 npm 管理客户端依赖项 56
- 3.5 总结 57

第二部分 Node 的 Web 开发

第 4 章 前端构建系统 60

- 4.1 了解基于 Node 的前端开发 60

4.2 用 npm 运行脚本	61	5.5.5 优点	83
4.2.1 创建定制的 npm 脚本	62	5.5.6 弱点	83
4.2.2 配置前端构建工具	63	5.6 Sails.js	83
4.3 用 Gulp 实现自动化	63	5.6.1 设置	83
4.3.1 把 Gulp 添加到项目中	64	5.6.2 定义路由	84
4.3.2 Gulp 任务的创建及运行	64	5.6.3 REST API	84
4.3.3 监测变化	66	5.6.4 优点	85
4.3.4 在大项目中把任务分散到不同 文件中	66	5.6.5 弱点	85
4.4 用 Webpack 构建 Web 程序	67	5.7 DerbyJS	85
4.4.1 使用打包器和插件	67	5.7.1 设置	85
4.4.2 配置和运行 Webpack	68	5.7.2 定义路由	86
4.4.3 用 Webpack 开发服务器	68	5.7.3 REST API	87
4.4.4 加载 CommonJS 模块和静态 资源	70	5.7.4 优点	87
4.5 总结	71	5.7.5 弱点	87
第 5 章 服务器端框架	72	5.8 Flatiron.js	87
5.1 用户画像	72	5.8.1 设置	88
5.1.1 菲尔：代理开发者	73	5.8.2 定义路由	88
5.1.2 纳迪娜：开源开发者	73	5.8.3 REST API	89
5.1.3 爱丽丝：产品开发者	73	5.8.4 优点	90
5.2 框架是什么	74	5.8.5 弱点	90
5.3 Koa	74	5.9 LoopBack	90
5.3.1 设置	76	5.9.1 设置	91
5.3.2 定义路由	76	5.9.2 定义路由	92
5.3.3 REST API	77	5.9.3 REST API	92
5.3.4 优点	77	5.9.4 优点	93
5.3.5 弱点	77	5.9.5 弱点	93
5.4 Kraken	77	5.10 比较	93
5.4.1 设置	77	5.11 编写模块化代码	95
5.4.2 定义路由	78	5.12 用户选择	95
5.4.3 REST API	79	5.13 总结	95
5.4.4 优点	79	第 6 章 深入了解 Connect 和 Express	96
5.4.5 弱点	79	6.1 Connect	96
5.5 hapi	79	6.1.1 创建 Connect 程序	97
5.5.1 设置	80	6.1.2 了解 Connect 中间件的工作 机制	97
5.5.2 定义路由	80	6.1.3 组合中间件	98
5.5.3 插件	81	6.1.4 中间件的顺序	98
5.5.4 REST API	82	6.1.5 创建可配置的中间件	99
		6.1.6 使用错误处理中间件	101

6.2 Express	103	8.3.2 用 Knex 实现连接和查询	168
6.2.1 生成程序框架	103	8.3.3 切换数据库	170
6.2.2 Express 和程序的配置	107	8.3.4 注意抽象漏洞	171
6.2.3 渲染视图	109	8.4 MySQL 和 PostgreSQL	171
6.2.4 Express 路由入门	113	8.5 ACID 保证	172
6.2.5 用户认证	120	8.5.1 原子性: 无论成败, 事务必须 整体执行	172
6.2.6 注册新用户	124	8.5.2 一致性: 始终确保约束条件	172
6.2.7 已注册用户登录	130	8.5.3 隔离性: 并发事务不会相互 干扰	172
6.2.8 用户加载中间件	133	8.5.4 耐用性: 事务是永久性的	173
6.2.9 创建 REST API	134	8.6 NoSQL	173
6.2.10 启用内容协商	140	8.7 分布式数据库	173
6.3 总结	142	8.8 MongoDB	174
第 7 章 Web 程序的模板	143	8.8.1 安装和配置	174
7.1 用模板保持代码的整洁性	143	8.8.2 连接 MongoDB	174
7.2 Embedded JavaScript 的模板	147	8.8.3 插入文档	175
7.2.1 创建模板	148	8.8.4 查询	176
7.2.2 将 EJS 集成到你的程序中	149	8.8.5 使用 MongoDB 标识	177
7.2.3 在客户端程序中使用 EJS	150	8.8.6 使用复制集	178
7.3 使用 Mustache 模板语言与 Hogan	151	8.8.7 了解写关注	180
7.3.1 创建模板	151	8.9 键/值存储	181
7.3.2 Mustache 标签	151	8.10 Redis	181
7.3.3 微调 Hogan	154	8.10.1 安装和配置	182
7.4 用 Pug 做模板	154	8.10.2 初始化	182
7.4.1 Pug 基础知识	155	8.10.3 处理键/值对	183
7.4.2 Pug 模板中的逻辑	157	8.10.4 处理键	184
7.4.3 组织 Pug 模板	159	8.10.5 编码与数据类型	184
7.5 总结	163	8.10.6 使用散列表	186
第 8 章 存储数据	164	8.10.7 使用列表	186
8.1 关系型数据库	164	8.10.8 使用集合	187
8.2 PostgreSQL	164	8.10.9 用频道实现发布/订阅功能	188
8.2.1 安装及配置	164	8.10.10 提升性能	189
8.2.2 创建数据库	165	8.11 嵌入式数据库	189
8.2.3 从 Node 中连接 Postgres	165	8.12 LevelDB	190
8.2.4 定义表	166	8.12.1 LevelUP 与 LevelDOWN	190
8.2.5 插入数据	166	8.12.2 安装	191
8.2.6 更新数据	166	8.12.3 API 概览	191
8.2.7 查询数据	167	8.12.4 初始化	191
8.3 Knex	167	8.12.5 键/值编码	192
8.3.1 查询构建器	168	8.12.6 键/值对的读写	192

8.12.7	可插拔的后台	193
8.12.8	模块化数据库	194
8.13	昂贵的序列化和反序列化	195
8.14	浏览器内存储	196
8.14.1	Web 存储: localStorage 和 sessionStorage	196
8.14.2	值的读写	197
8.14.3	localStorage	199
8.14.4	读和写	199
8.15	存储托管	200
8.16	选哪个数据库	201
8.17	总结	201
第 9 章	测试 Node 程序	202
9.1	单元测试	203
9.1.1	assert 模块	203
9.1.2	Mocha	206
9.1.3	Vows	211
9.1.4	Chai	213
9.1.5	Should.js	214
9.1.6	Sinon.JS 的探测器和存根	216
9.2	功能测试	218
9.3	处理失败的测试	221
9.3.1	获取更详细的日志	221
9.3.2	更好的栈跟踪	223
9.4	总结	224
第 10 章	Node 程序的部署及运维	225
10.1	安置 Node 程序	225
10.1.1	平台即服务	226
10.1.2	服务器	227
10.1.3	容器	228
10.2	部署的基础知识	229
10.2.1	从 Git 库部署	229
10.2.2	保证 Node 不掉线	230
10.3	在线时长和性能的最大化	231
10.3.1	用 Upstart 保证在线时长	232
10.3.2	集群 API: 充分利用多核 处理器	233
10.3.3	静态文件及代理	235
10.4	总结	236

第三部分 超越 Web 开发

第 11 章	编写命令行程序	238
11.1	了解惯例和理念	238
11.2	parse-json	239
11.3	使用命令行参数	239
11.3.1	解析命令行参数	239
11.3.2	验证参数	240
11.3.3	将 stdin 作为文件传递	241
11.4	用 npm 分享命令行工具	242
11.5	用管道连接脚本	243
11.5.1	将数据通过管道传给 parse-json	243
11.5.2	处理错误和退出码	243
11.5.3	在 Node 中使用管道	245
11.5.4	管道与命令的执行顺序	246
11.6	解释真正的脚本	247
11.7	总结	247
第 12 章	用 Electron 征服桌面	248
12.1	认识 Electron	248
12.1.1	Electron 的技术栈	249
12.1.2	界面设计	250
12.2	创建一个 Electron 程序	250
12.3	搭建完整的桌面端程序	252
12.3.1	引导 React 与 Babel	253
12.3.2	安装依赖项	253
12.3.3	设置 Webpack	254
12.4	React 程序	255
12.4.1	定义 Request 组件	256
12.4.2	定义 Response 组件	258
12.4.3	React 组件之间的通信	261
12.5	构建与分发	261
12.5.1	用 Electron 打包器构建程序	261
12.5.2	打包	262
12.6	总结	263
附录 A	安装 Node	264
附录 B	自动化的网络抓取	267
附录 C	Connect 的官方中间件	277
术语表		307

Part 1

第一部分

Node 基础知识介绍

现如今，Node 已经出落成了一个成熟的 Web 开发平台。本书第 1 章到第 3 章介绍 Node 的主要特性，包括如何使用 npm 和 Node 的核心模块。你还将看到如何在 Node 上使用现代版 JavaScript，以及如何从头开始构建一个 Web 应用程序。看完这些章节之后，对于 Node 能做什么，以及该如何创建自己的项目，你将会有非常深刻的认识。