

Brad Dayley
[美] Brendan Dayley 著
Caleb Dayley

卢涛 译

 Pearson

Broadview[®]
www.broadview.com.cn

Node.js+MongoDB+ Angular Web开发

MEAN全栈权威指南

*Node.js, MongoDB and Angular Web Development,
Second Edition*



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Brad Dayley
[美] Brendan Dayley 著
Caleb Dayley

卢涛 译

Node.js+MongoDB+ Angular Web开发

MEAN全栈权威指南



*Node.js, MongoDB and Angular Web Development,
Second Edition*

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Node.js 是一种领先的服务器端编程环境, MongoDB 是最流行的 NoSQL 数据库, 而 Angular 正迅速成为基于 MVC 的前端开发的领先框架。它们结合在一起使得能够完全用 JavaScript 创建从服务器到客户端浏览器的高性能站点和应用程序。

本书为想要将这 3 种技术整合为全面的有效解决方案的 Web 程序员提供了完整指南。书中简洁而清晰地介绍了这 3 种技术, 然后迅速转向构建几种常见 Web 应用程序的实战。

读者将学会使用 Node.js 和 MongoDB 来建立更具可伸缩性的高性能网站, 并利用 Angular 创新的 MVC 方法构建更有效的网页和应用程序, 以及把这三者结合在一起使用, 从而提供卓越的下一代 Web 解决方案。

本书适合对 HTML 的基础知识已经有所了解, 并可以用现代编程语言完成一些开发工作的读者。读者如果对 JavaScript 有一定了解, 则将更易于理解本书的内容。

Authorized translation from the English language edition, entitled Node.js, MongoDB and Angular Web Development, Second Edition, ISBN: 978-0-13-465553-6 by Brad Dayley, Brendan Dayley, Caleb Dayley, published by Pearson Education, Inc, Copyright © 2018 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PUBLISHING HOUSE OF ELECTRONICS INDUSTRY, Copyright © 2018

本书简体中文版专有出版权由 Pearson Education 培生教育出版集团授予电子工业出版社。未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

本书简体中文版贴有 Pearson Education 培生教育出版集团激光防伪标签, 无标签者不得销售。

版权贸易合同登记号 图字: 01-2017-8357

图书在版编目 (CIP) 数据

Node.js+MongoDB+Angular Web 开发: MEAN 全栈权威指南 / (美) 布拉德·布雷 (Brad Dayley), (美) 布兰登·布雷 (Brendan Dayley), (美) 凯勒·布雷 (Caleb Dayley) 著; 卢涛译. —北京: 电子工业出版社, 2018.10

书名原文: Node.js, MongoDB and Angular Web Development, Second Edition

ISBN 978-7-121-35096-2

I. ①N… II. ①布… ②布… ③凯… ④卢… III. ①网页制作工具—JAVA 语言—程序设计—指南
IV. ①TP393.092.2-62②TP312.8-62

中国版本图书馆 CIP 数据核字 (2018) 第 218184 号

策划编辑: 张春雨

责任编辑: 李云静

印 刷: 三河市君旺印务有限公司

装 订: 三河市君旺印务有限公司

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: 34.75

字数: 715 千字

版 次: 2018 年 10 月第 1 版

印 次: 2018 年 10 月第 1 次印刷

定 价: 129.00 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

随着互联网的发展，曾几何时，“全栈工程师”（Full Stack Engineer）的概念开始兴起，这种职位要求应征者对开发堆栈的每个方面都有所掌握，包括服务器、网络及宿主环境、数据建模、业务逻辑、API/Action/MVC、用户界面等。过去，这意味着全栈工程师需要面对归属不同层次的多种软件环境和语言，并处理各层之间的交互，这些技术中的任何一种都要耗费大量的精力和时间来学习，因而会面临很大的挑战。现在，这种情况已经改变，JavaScript 语言既可用于客户端开发，又能用于编写服务器端应用程序，还能方便地与传统及新型数据库交互，其中有些代码还能复用，从而为开发功能丰富的互联网应用程序创造了良好的条件。

Node.js 是一种领先的服务器端编程环境，它和 Express 的结合能够实现高度可伸缩的动态 Web 服务器，并可用 JavaScript 编写 Web 服务。

MongoDB 是目前最流行的 NoSQL 数据库，可用于 Web 应用程序数据的存储，并能从 Node.js JavaScript 代码访问。

Angular 正迅速成为基于 MVC 的前端开发的领先框架，它的自定义指令扩展了 HTML 语言。

这三者整合在一起使得能够完全用 JavaScript 创建从服务器到客户端浏览器的高性能站点和应用程序。

本书提供了将这 3 种技术整合成全面的有效解决方案的完整指南。书中简要而清晰地讲述了这 3 种技术，然后迅速转到构建几种常见的 Web 应用程序上面。最后还用多个实际的例子实现了可以与 Node.js Web 服务器交互的客户端服务，为用户提供了丰富的交互功能的动态浏览器视图，以及为 Web 页面添加用户身份验证和嵌套评论等组件。

作者 Brad Dayley 是一名高级软件工程师，精通 jQuery、JavaScript、MongoDB，著有多本技术书籍，并在企业应用程序及 Web 界面方面具有丰富的开发经验，本书正是他对实际工作成果的总结。第 2 版增加了两位作者：Brendan Dayley 是经验丰富的技术书籍作家和 Web 软件工程师；Caleb Dayley 则是计算机科学专业的大学生，他对于设计和开发下

一代创新软件很有兴趣。

通过学习本书，读者将学会如何使用 Node.js 和 MongoDB 来建立更具可伸缩性的高性能网站，如何利用 Angular 创新的 MVC 方法构建更有效的网页和应用程序，以及如何把这三者结合在一起使用，从而提供卓越的下一代 Web 解决方案。

李绿霞、卢林、陈克非、李洪秋、张慧珍、李又及、卢晓瑶、陈克翠、汤有四、李阳、刘雯也参与了部分翻译工作，感谢他们在本书翻译工作中的辛勤付出。

感谢我们的儿子卢令一，他读小学三年级了，他努力好学，本书的出版也有他的一份贡献。

感谢电子工业出版社张春雨编辑对我们的信任，让我们继续从事第 2 版的翻译工作。

最后，希望这本书能对读者有所助益。但由于译者经验和水平有限，译文中难免有不妥之处，恳请读者批评指正！

卢涛 李颖

2018 年 9 月 18 日

作者简介

Brad Dayley 是一名高级软件工程师，在开发企业应用程序及 Web 界面方面，他拥有超过 20 年的工作经验。他熟练应用 JavaScript 和 jQuery 多年，并是 *Learning Angular, jQuery and JavaScript Phrasebook* 和 *Sams Teach Yourself AngularJS, JavaScript, and jQuery All in One* 的作者。他曾设计并实现了一大批应用程序和服务——从应用服务器到复杂的 Web 应用程序。

Brendan Dayley 是一名 Web 应用程序开发人员，他喜欢学习和实施最新、最好的技术。他是 *Learning Angular* 和 *Sams Teach Yourself AngularJS, JavaScript, and jQuery All in One* 的合著者。他使用 JavaScript、TypeScript 和 Angular 编写了大量的 Web 应用程序，他正在探索新的 Web 和移动技术（如增强现实）的功能并研究如何将其用于创新解决方案。

Caleb Dayley 是一名在校大学生，主修计算机科学。他尽可能地学习自己所能学到的东西，并且自学了很多关于程序设计的知识。他自学了几种语言，包括 JavaScript、C#，并且使用了本书的第 1 版。他对未来的发展，以及为有机会帮助设计和开发下一代创新软件感到兴奋，这些软件将继续改善我们生活、工作和娱乐的方式。

致谢

谨以此篇向所有为本书做出贡献的人致谢。首先，我要感谢我的贤妻，感谢她给予我灵感、爱和支持。没有她，我不会写就此书。我还要感谢我的儿子们，感谢在写作时间里他们给予我的支持。感谢 Mark Taber 让这本书朝正确的方向演进。

——Brad Dayley

我要感谢所有帮助我完成这本书的人。首先，要感谢我的妻子，她促使我变得更出色，并给予我全心全意的爱。另外，感谢我的父亲，他不仅在写作和编程方面，而且在生活中为我提供指导。感谢我的母亲，当我需要她的时候，她一直在我身边。最后，感谢 Mark Taber，他给了我参与写作本书部分章节的机会。

——Caleb Dayley

读者服务

轻松注册成为博文视点社区用户 (www.broadview.com.cn), 扫码直达本书页面。

- **下载资源:** 本书提供示例代码及资源文件, 可在[下载资源处](#)下载。
- **提交勘误:** 您对书中内容的修改意见可在[提交勘误处](#)提交, 若被采纳, 将获赠博文视点社区积分 (在您购买电子书时, 积分可用来抵扣相应金额)。
- **交流互动:** 在页面下方[读者评论处](#)留下您的疑问或观点, 与我们和其他读者一同学习交流。

页面入口: <http://www.broadview.com.cn/35096>



目 录

引 言.....	1
----------	---

第 1 部分 入 门

第 1 章 介绍 Node.js-to-Angular 套件.....	9
-------------------------------------	---

1.1 了解基本的 Web 开发框架.....	9
1.1.1 用户.....	10
1.1.2 浏览器.....	10
1.1.3 Web 服务器.....	12
1.1.4 后端服务.....	12
1.2 了解 Node.js-to-Angular 套件组件.....	13
1.2.1 Node.js.....	13
1.2.2 MongoDB.....	14
1.2.3 Express.....	15
1.2.4 Angular.....	15
1.3 小结.....	16
1.4 下一章.....	16

第 2 章 JavaScript 基础.....	17
--------------------------	----

2.1 定义变量.....	17
2.2 了解 JavaScript 数据类型.....	18
2.3 使用运算符.....	19
2.3.1 算术运算符.....	19
2.3.2 赋值运算符.....	20

2.3.3	运用比较和条件运算符	20
2.4	实现循环	22
2.4.1	while 循环	23
2.4.2	do/while 循环	23
2.4.3	for 循环	23
2.4.4	for/in 循环	24
2.4.5	中断循环	25
2.5	创建函数	26
2.5.1	定义函数	26
2.5.2	将变量传递给函数	26
2.5.3	从函数返回值	27
2.5.4	使用匿名函数	27
2.6	理解变量作用域	28
2.7	使用 JavaScript 对象	28
2.7.1	使用对象语法	29
2.7.2	创建自定义对象	30
2.7.3	使用原型对象模式	30
2.8	处理字符串	31
2.8.1	合并字符串	32
2.8.2	在字符串中搜索子串	33
2.8.3	在一个字符串中替换单词	33
2.8.4	将字符串分割成数组	33
2.9	使用数组	33
2.9.1	合并数组	35
2.9.2	遍历数组	35
2.9.3	将数组转换为字符串	35
2.9.4	检查数组是否包含某个条目	36
2.9.5	在数组中添加条目和删除条目	36
2.10	添加错误处理	36
2.10.1	try/catch 块	37
2.10.2	抛出你自己的错误	37
2.10.3	使用 finally	38

2.11 小结	38
2.12 下一章	38
第 2 部分 学习 Node.js	
第 3 章 开始使用 Node.js	41
3.1 了解 Node.js	41
3.1.1 谁在使用 Node.js	41
3.1.2 Node.js 的用途	42
3.1.3 Node.js 包含的内容	42
3.2 Node.js 安装	44
3.2.1 纵观 Node.js 安装位置	44
3.2.2 验证 Node.js 可执行文件	44
3.2.3 选择 Node.js IDE	45
3.3 使用 Node 包	45
3.3.1 什么是 Node 封装模块	45
3.3.2 了解 Node 包注册表	46
3.3.3 使用 Node 包管理器	46
3.3.4 搜索 Node 封装模块	47
3.3.5 安装 Node 封装模块	48
3.3.6 使用 package.json	49
3.4 创建 Node.js 应用程序	50
3.4.1 创建 Node.js 模块封装	51
3.4.2 将一个 Node.js 封装模块发布到 NPM 注册表	52
3.4.3 在 Node.js 应用程序中使用 Node.js 封装模块	54
3.5 将数据写入控制台	55
3.6 小结	56
3.7 下一章	56
第 4 章 在 Node.js 中使用事件、监听器、定时器和回调	57
4.1 了解 Node.js 事件模型	57
4.1.1 比较事件回调和线程模型	57

4.1.2	在 Node.js 中阻塞 I/O.....	58
4.1.3	会话示例.....	60
4.2	将工作添加到事件队列.....	60
4.2.1	实现定时器.....	61
4.2.2	使用 nextTick 来调度工作.....	64
4.2.3	实现事件发射器和监听器.....	65
4.3	实现回调.....	68
4.3.1	向回调函数传递额外的参数.....	69
4.3.2	在回调中实现闭包.....	70
4.3.3	链式回调.....	71
4.4	小结.....	72
4.5	下一章.....	72
第 5 章	在 Node.js 中处理数据 I/O.....	73
5.1	处理 JSON.....	73
5.1.1	把 JSON 转换成 JavaScript 对象.....	73
5.1.2	把 JavaScript 对象转换为 JSON.....	74
5.2	使用 Buffer 模块缓冲数据.....	74
5.2.1	了解缓冲数据.....	75
5.2.2	创建缓冲区.....	75
5.2.3	写入缓冲区.....	76
5.2.4	从缓冲区读取.....	77
5.2.5	确定缓冲区的长度.....	78
5.2.6	复制缓冲区.....	78
5.2.7	对缓冲区切片.....	80
5.2.8	拼接缓冲区.....	80
5.3	使用 Stream 模块来传送数据.....	81
5.3.1	Readable 流.....	82
5.3.2	Writable 流.....	84
5.3.3	Duplex 流.....	86
5.3.4	Transform 流.....	88
5.3.5	把 Readable 流用管道输送到 Writable 流.....	89

5.4	用 zlib 压缩与解压缩数据	91
5.4.1	压缩和解压缩缓冲区	91
5.4.2	压缩/解压缩流	93
5.5	小结	93
5.6	下一章	93
第 6 章	从 Node.js 访问文件系统	95
6.1	同步和异步文件系统调用	95
6.2	打开和关闭文件	96
6.3	写入文件	97
6.3.1	简单文件写入	97
6.3.2	同步文件写入	98
6.3.3	异步写入文件	99
6.3.4	流式文件写入	101
6.4	读取文件	102
6.4.1	简单文件读取	102
6.4.2	同步文件读取	103
6.4.3	异步文件读取	104
6.4.4	流式文件读取	105
6.5	其他文件系统任务	106
6.5.1	验证路径的存在性	106
6.5.2	获取文件信息	107
6.5.3	列出文件	108
6.5.4	删除文件	110
6.5.5	截断文件	110
6.5.6	建立和删除目录	111
6.5.7	重命名文件和目录	112
6.5.8	监视文件更改	112
6.6	小结	113
6.7	下一章	113

第 7 章 在 Node.js 中实现 HTTP 服务	115
7.1 处理 URL	115
7.1.1 了解 URL 对象	116
7.1.2 解析 URL 组件	117
7.2 处理查询字符串和表单参数	117
7.3 了解请求、响应和服务器对象	118
7.3.1 <code>http.ClientRequest</code> 对象	118
7.3.2 <code>http.ServerResponse</code> 对象	121
7.3.3 <code>http.IncomingMessage</code> 对象	122
7.3.4 <code>http.Server</code> 对象	122
7.4 在 Node.js 中实现 HTTP 客户端和服务端	124
7.4.1 提供静态文件服务	124
7.4.2 实现动态的 GET 服务器	126
7.4.3 实现 POST 服务器	128
7.4.4 与外部源交互	131
7.5 实现 HTTPS 服务器和客户端	133
7.5.1 创建 HTTPS 客户端	134
7.5.2 创建 HTTPS 服务器	135
7.6 小结	136
7.7 下一章	136
第 8 章 在 Node.js 中实现套接字服务	137
8.1 了解网络套接字	137
8.2 了解 TCP 服务器和 <code>Socket</code> 对象	138
8.2.1 <code>net.Socket</code> 对象	138
8.2.2 <code>net.Server</code> 对象	141
8.3 实现 TCP 套接字服务器和客户端	144
8.3.1 实现 TCP 套接字客户端	144
8.3.2 实现 TCP 套接字服务器	146
8.4 实现 TLS 服务器和客户端	149
8.4.1 创建 TLS 套接字客户端	150
8.4.2 创建 TLS 套接字服务器	151

8.5 小结	153
8.6 下一章	153
第 9 章 在 Node.js 中使用多处理器扩展应用程序	155
9.1 了解 process 模块.....	155
9.1.1 了解进程 I/O 管道.....	155
9.1.2 了解进程的信号	156
9.1.3 使用 process 模块控制进程执行.....	156
9.1.4 从 process 模块获取信息.....	157
9.2 实现子进程	159
9.2.1 了解 ChildProcess 对象.....	159
9.2.2 使用 exec() 在另一个进程上执行一个系统命令	161
9.2.3 使用 execFile() 在另一个进程上执行一个可执行文件.....	162
9.2.4 使用 spawn() 在另一个 Node.js 实例中产生一个进程.....	163
9.2.5 实现子派生	165
9.3 实现进程集群	167
9.3.1 使用 cluster 模块	168
9.3.2 了解 Worker 对象.....	169
9.3.3 实现一个 HTTP 集群	170
9.4 小结	172
9.5 下一章	172
第 10 章 使用其他 Node.js 模块	173
10.1 使用 os 模块.....	173
10.2 使用 util 模块.....	175
10.2.1 格式化字符串	175
10.2.2 检查对象类型	176
10.2.3 将 JavaScript 对象转换为字符串	176
10.2.4 从其他对象继承功能	177
10.3 使用 dns 模块	178
10.4 使用 crypto 模块	180
10.5 其他 Node 模块和对象.....	181

10.6 小结	182
10.7 下一章	182

第 3 部分 学习 MongoDB

第 11 章 了解 NoSQL 和 MongoDB.....	185
11.1 为什么要采用 NoSQL	185
11.2 了解 MongoDB	186
11.2.1 理解集合	186
11.2.2 了解文档	186
11.3 MongoDB 的数据类型	187
11.4 规划你的数据模型	188
11.4.1 使用文档引用来规范化数据	189
11.4.2 使用嵌入式文档反规范化数据	190
11.4.3 使用封顶集合	191
11.4.4 了解原子写操作	192
11.4.5 考虑文件增长	192
11.4.6 识别索引、分片和复制的机会	193
11.4.7 大集合与大量集合的对比	193
11.4.8 决定数据生命周期	193
11.4.9 考虑数据的可用性和性能	194
11.5 小结	194
11.6 下一章	194
第 12 章 MongoDB 入门.....	195
12.1 构建 MongoDB 的环境	195
12.1.1 MongoDB 的安装	195
12.1.2 启动 MongoDB	196
12.1.3 停止 MongoDB	197
12.1.4 从 shell 客户端访问 MongoDB.....	197
12.2 管理用户账户	200
12.2.1 列出用户	200

12.2.2	创建用户账户	201
12.2.3	删除用户	202
12.3	配置访问控制	203
12.3.1	创建用户管理员账户	203
12.3.2	打开身份验证	204
12.3.3	创建数据库管理员账户	204
12.4	管理数据库	205
12.4.1	显示数据库清单	205
12.4.2	切换当前数据库	205
12.4.3	创建数据库	206
12.4.4	删除数据库	206
12.4.5	复制数据库	207
12.5	管理集合	207
12.5.1	显示数据库中的集合列表	207
12.5.2	创建集合	208
12.5.3	删除集合	208
12.5.4	在集合中查找文档	209
12.5.5	将文档添加到集合中	210
12.5.6	从集合中删除文档	210
12.5.7	更新集合中的文档	211
12.6	小结	212
12.7	下一章	212
第 13 章 MongoDB 和 Node.js 入门		213
13.1	把 MongoDB 的驱动程序添加到 Node.js	213
13.2	从 Node.js 连接到 MongoDB	213
13.2.1	了解写入关注	214
13.2.2	通过 MongoClient 对象从 Node.js 连接到 MongoDB	214
13.3	了解用在 MongoDB Node.js 驱动程序中的对象	218
13.3.1	了解 Db 对象	218
13.3.2	了解 Admin 对象	220
13.3.3	了解 Collection 对象	220

13.3.4	了解 Cursor 对象.....	222
13.4	访问和操作数据库.....	223
13.4.1	列出数据库.....	223
13.4.2	创建数据库.....	223
13.4.3	删除数据库.....	224
13.4.4	创建、列出和删除数据库实例.....	224
13.4.5	获取 MongoDB 服务器的状态.....	226
13.5	访问和操作集合.....	227
13.5.1	列出集合.....	227
13.5.2	创建集合.....	227
13.5.3	删除集合.....	227
13.5.4	创建、列出和删除集合的示例.....	228
13.5.5	获取集合信息.....	229
13.6	小结.....	230
13.7	下一章.....	230
第 14 章 从 Node.js 操作 MongoDB 文档.....		231
14.1	了解数据库更改选项.....	231
14.2	了解数据库更新运算符.....	232
14.3	将文档添加到集合.....	233
14.4	从集合获取文档.....	235
14.5	更新集合中的文档.....	237
14.6	原子地修改文档的集合.....	239
14.7	保存集合中的文档.....	241
14.8	使用 upsert 往集合中插入文档.....	242
14.9	从集合中删除文档.....	244
14.10	从集合中删除单个文档.....	246
14.11	小结.....	248
14.12	下一章.....	248
第 15 章 从 Node.js 访问 MongoDB.....		249
15.1	介绍数据集.....	249