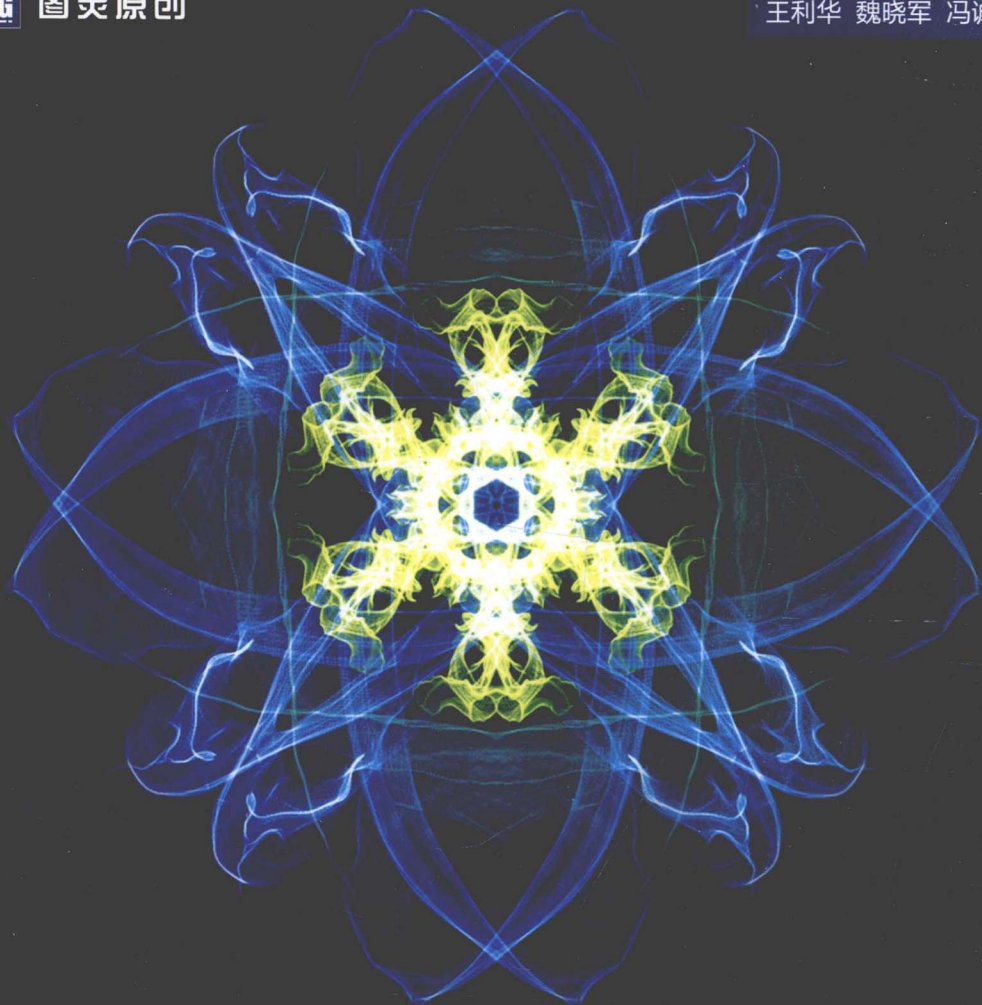


TURING

图灵原创

王利华 魏晓军 冯诚祺 编著



React Native

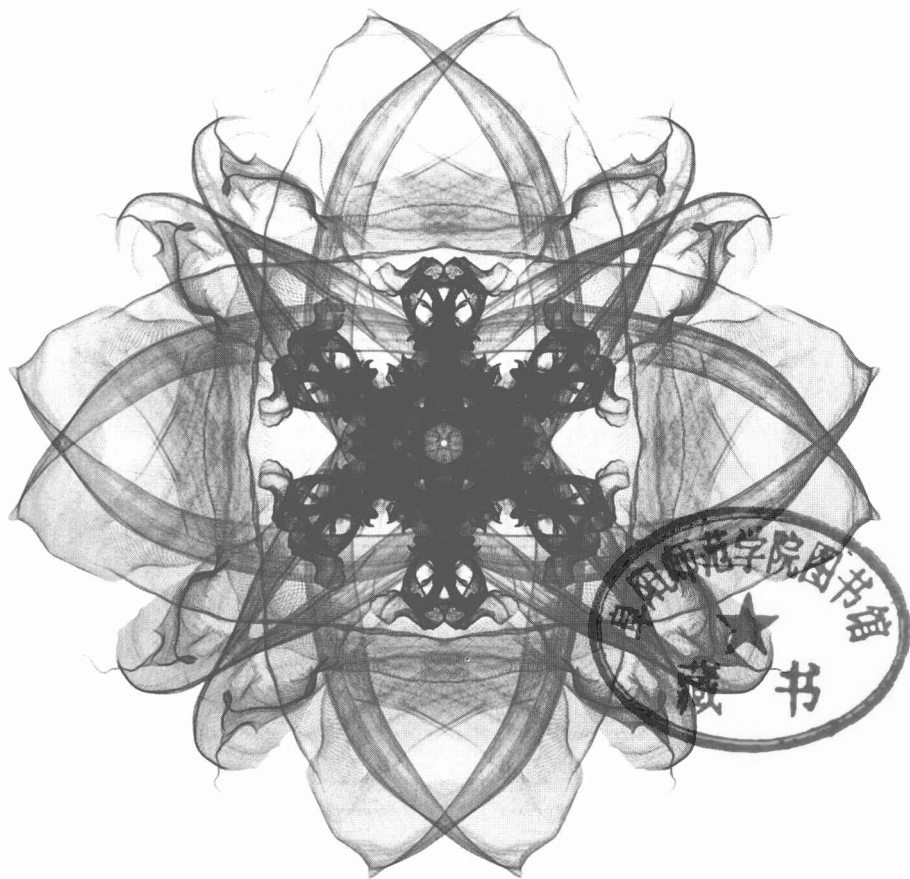
入门与实战



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



React Native

入门与实战

人民邮电出版社
北京

图书在版编目 (C I P) 数据

React Native入门与实战 / 王利华, 魏晓军, 冯诚
祺编著. -- 北京: 人民邮电出版社, 2016. 1
(图灵原创)
ISBN 978-7-115-41191-4

I. ①R… II. ①王… ②魏… ③冯… III. ①移动终
端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2015)第290196号

内 容 提 要

本书共4部分, 首先简要介绍了开发相关的基础知识, 然后介绍了React Native的API、组件以及Native扩展和组件的封装, 接着介绍了App的动态更新和上架过程, 最后通过3个案例介绍了如何使用React Native开发原生App。

本书适合想使用React Native开发原生应用的人阅读。

-
- ◆ 编 著 王利华 魏晓军 冯诚祺
责任编辑 王军花
责任印制 杨林杰
 - ◆ 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>
三河市海波印务有限公司印刷
 - ◆ 开本: 800×1000 1/16
印张: 24.25
字数: 454千字 2016年1月第1版
印数: 1-4 000册 2016年1月河北第1次印刷
-

定价: 79.00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: 京崇工商广字第0021号

序 一

When I heard React Native, It was 2014-8 from a Facebook friend who was in the team. They were creating this new technology for Facebook App (internal use). He said Facebook plans to open source this technology to App developers. In 2015-3, React Native was released to the public.

Although the technology have limitations across iOS and Android, its design approach was revolutionary. It proves to work close to native App experience (performance) as well as to leverage web developers skills. React Native has already impact on App developers now. It becomes a top choice for developing new App.

What Lihua、Xiaojun and Chengqi surprised me is they were writing this book together in last 6 months. While they are trying out React Native, they love it. They believe that this technology is useful for App development. By creating the book in Chinese, they take React Native to next level of the developer reach and make the wide impact on Chinese App Community.

Their spirit of spreading the new technology and their endeavor of creating the book together as team work should highly praised. They Not only master this technology for use by themselves, but also show it to Chinese developers. The mobile internet is pushing forward , with many developers in this spirit and effort!

2014年8月，我从一个在Facebook工作的朋友那里听说了React Native，他们正在使用这种新技术开发内部App。他表示，Facebook计划将该技术开源。2015年3月，React Native公开发布了。

尽管这项技术在兼容iOS和Android方面仍有一定的局限性，但它的设计理念别出心裁。在实现媲美原生App的用户体验（性能）的同时，React Native允许Web开发者更多地基于现有经验开发App。现在，它已经影响了很多App开发者，在开发下一个App时，React Native将成为开发人员的首选技术。

在过去的6个月中，利华、晓军和诚祺合力完成了本书的创作，这让我感到惊喜。他们在不断尝试React Native，他们热爱这项技术。他们坚信，这门技术对于App开发者而言十分有帮助。《React Native入门与实战》一书的出版，将使更多开发人员接触到这项技术，并对国内App开发者社区产生重要影响。

在创作本书的过程中，他们的分享精神和团队协作都值得表扬。他们不仅自己掌握了这门技术，同时也将其推介给了所有国内的开发人员。移动互联网的发展，无疑得益于众多像他们这样的开发人员的努力贡献和分享精神。

2015-11-20

Eric Ye, 携程旅行网CTO

序 二

随着苹果公司推出iPhone和App Store，移动开始持续升温，各种创新不断，风头逐渐盖过Web，吸引了大量开发人员进入移动领域。这无疑使得苹果公司成为最大的赢家，操控着很多公司的命运。

Facebook在Web技术上非常成功，深知移动未来的重要性，但又不想受制于苹果公司。于是投入大量的人力和物力，在移动HTML5上攻坚克难，虽取得了不少进展，但始终不如意。2012年9月，Facebook表示：“Betting on HTML5 was a mistake.”全力转型Native App开发。令人没有想到的是，两年之后Facebook居然推出了React Native for iOS技术，让人眼前一亮，兴奋不已。

大家都知道，Native App因其性能优越和功能强大而笑傲江湖，但终究逃不出Apple的掌心，多版本维护非常痛苦。HTML5虽然有Web的优势，但因WebView在移动设备的性能和电力等因素的制约，性能总被诟病，难成大器。而Hybrid App集Native App和Web优点于一体，还可以相互补短，似乎应该成为大家的选择。然而它在成熟度、标准化等方面的顾虑，也会是一个不小的问题。React Native技术的诞生则普遍被大家接受，各大公司纷纷介入，给人很大的想象空间。它的底层引擎是JavaScript Core，调用的是原生组件而非HTML5组件（HTML+CSS+JavaScript构建的组件）。运行时，可以做到与Native App媲美的体验，同时因为JavaScript代码可以使用后端强大的Web方式管理，既可以做到高效开发，也可以做到快速部署和问题热修复。React Native App运行在客户的手机上，而控制端可以在后端，可以充分发挥Web的能力，就像一个牵线木偶，任凭你表演。

该书不仅对React Native基础知识讲解得很透彻，同时还辅助了不少案例，比如第9章的LBS应用开发和第10章的OpenAPI应用开发。相信阅读本书后，你一定会有所收获。

雄关漫道真如铁，而今迈步从头越。很高兴看到携程框架团队同学在前端框架和React Native方面所做出的努力，也希望读者进入React Native领域后能够不断丰富社区，促进React Native技术的蓬勃发展。

吴其敏，携程旅行网框架研发部负责人，高级研发总监

序 三

最近三四年间，国内外的前端与全栈开发者社区都在坚持不懈地追寻使用JavaScript与HTML、CSS技术体系开发App内场景的核心工程技术。这种技术，在国内很多公司与团队中，被通称为H5。这种工程类的尝试最早出现在新闻资讯页等强排版、弱交互的产品场景中，因其灵活的布局能力和免发版的敏捷迭代潜力而大受欢迎。而后在涌现出大批第三方应用市场的浪潮中，也成为了应用市场展示App应用详情的技术标配。在此过程中，PhoneGap（后来的Apache Cordova）等组件的出现满足了JavaScript与Android/iOS程序之间的通信需求，及时补全了这种工程方案在系统能力上的短板。大家在更大范围地推进这种方案的过程中，却遇到了一个致命的问题，那就是这种技术在处理无限滑动列表（如微博的信息流）时，受WebView的影响，表现出了极差的点击响应、内存性能和兼容性。社区中有不少有识之士提出了模板配置化+原生渲染，或引入多个WebView运行SPA以缓解内存问题等行之有效的方案，但均因一定程度上牺牲了灵活、敏捷的方案优势，而无法获得广泛采纳。这样的缺陷直接制约了JavaScript in App工程方案的再扩大。在百度供职时，我曾负责主持Clouda云端一体框架的研发工作，可说是在社区一线全程参与了整个演进过程。

React Native的出现，彻底、整洁且智慧地解决了这个痛点，且通过WebKit的引入完整保留了JavaScript语言的完整逻辑性，通过原生的渲染保持了“不沾手”的顺滑体验和出色的内存管理，没有妥协地实现了大家需要的JavaScript in App的工程体系能力。

本书的第一作者利华曾与我在阿里高德共事，当时就表现出了对社区技术发展的很强敏感性。本书的出版，距React Native宣布支持Android平台仅月余，可以说是React Native实战的第一手中文资料。

我推荐大家阅读这本书。在掩卷之后，你当能执React Native于实际工程项目中快速作战，甚至实践于“云控App”的高精尖领域。届时，我也希望读者你进一步思考与了解React Native的设计与结构，在社区中贡献一份属于你的力量，为这个领域中更多的同学引路。

童遥，阿里高德开放平台总经理

序 四

从React Native诞生那一天起，对于作为后端出身但一直对前端念念不忘的我来说，就像发现了一个埋藏很深亟待挖掘的金矿一样兴奋。

可以说，从16年前第一次接触DHTML后，只有NodeJS的出现，才能第二次让我产生对前端技术的兴奋感。而这一次，React Native的横空出世，再次触及了我快要冷淡下去的前端兴奋感。

本书作者魏晓军是我在携程的同事，也是当时及现在携程技术团队中的顶尖前端高手。晓军所领导的携程H5/Hybrid框架团队，是整个携程移动优先战略得以顺利推进的核心技术动力。

感谢晓军和他的团队为前端技术社区及广大前端技术爱好者推出这本期期待已久的作品，而我，作为一名仍对前端技术念念不忘的 IT 从业人员，也会在该书上市后第一时间抢购回家，饱饱眼福。

张雪峰，饿了么CTO，于饿了么总部

致 谢

因为我们团队比较早关注了React Native，同时也做了些分享，因此有幸结识了图灵公司的王军花老师。2015年6月24日，收到出版社王老师的邮件，希望我们能写一本关于React Native的图书，但是因为React Native才刚刚发布不久，所以在思考了一段时间后，才下定决心写作。

新事物的发展总是曲折的。React Native就像一个新生儿一样接受着质疑和考究。我们期待它更加完善、更加稳定。在本书的写作过程中，我们一直关注着React Native的更新和发展，本书的内容也随着React Native的发展而修改了多次。

对于工程师而言，写作是一件比较痛苦的事，我们更加喜欢使用代码来表达我们的想法。很多时候，一句话需要推敲好多次才能下笔。其中，初稿有很多不完善的地方，都是王老师日夜兼程地提出建议和修改，这里要对王老师表示深深的感谢！写作本书，对我们而言，不仅加深了对React Native的理解，同时也提升了个人的写作水平，这要感谢人民邮电出版社图灵公司给予的机会。

本书得以完成要感谢很多人，感谢储诚栋等领导对工作的支持和对本书写作的指导，感谢Eric Ye、吴其敏、童遥、张雪峰为本书慷慨作序，感谢储诚栋、陈浩然、勾股、鬼道为本书写推荐语，感谢潘菲菲、卢玮、蒋竟等同事的帮助。

人生难得3万日，青春区区5千天。希望本书能成为React Native开发者的小助手，也希望我们和学习React Native的同学一起努力，为React Native的蓬勃发展贡献自己的力量。

王利华 魏晓军 冯诚祺
2015年11月于上海

前言

React Native开启了开发原生App的新方式，不仅提高了开发效率，同时提高了App的用户体验。相比Web App而言，React Native可以使用原生的组件和API，这样就可以释放Native的能力和体验；相比Native开发而言，前端开发者可以使用JavaScript开发原生应用，这样开发效率将会得到很大的提高。

本书目的

目前，国内针对React Native讲解的图书和资料都很少，阅读本书可以帮助你更好地开发React Native应用。或许你已经了解React Native的基本内容，或许你已经开始了React Native的开发之旅，无论如何，本书都希望可以带领大家拥抱React Native，使用React Native。当然，本书也希望弥补中文资料在这方面的欠缺。

内容和组织结构

本书的内容是我们在实践过程中总结得到的，一共分为4部分。

第一部分为基础语法篇，共两章内容，主要介绍了React Native的开发基础知识。

第1章介绍了React Native的环境搭建、React与React Native之间的关系，以及如何学习React Native。

第2章主要介绍了React Native的开发基础知识，包含flexbox布局、JSX语法，并且详细介绍了React Native创建项目的过程。

第二部分为API和组件篇，共4章内容，主要介绍了React Native的API、组件以及Native扩展和组件的封装。

第3章介绍了React Native常用组件，包含View组件、Text组件、NavigatorIOS组件、TextInput组件、Touchable类组件（TouchableHighlight、TouchableOpacity和TouchableWithoutFeedback）、Image组件、TabBarIOS组件和WebView组件。

第4章介绍了React Native常用的API，包含AppRegistry、AsyncStorage、AlertIOS、ActionSheetIOS、PixelRatio、AppStateIOS、StatusBarIOS、NetInfo、CameraRoll、VibrationIOS、Geolocation、网

络数据请求的应用以及定时器和动画。

第5章介绍了React Native的实现原理以及在原生组件和API上的扩展，并且以一个“图表”组件为案例进行实战讲解。

第6章介绍了使用JavaScript封装React Native组件，主要实现了二级菜单组件、日历组件以及初步介绍了开源组件的用法。

第三部分为App更新和上架篇，共一章内容。

第7章介绍了App的动态更新和上架过程。

第四部分为实战篇，共3章内容，介绍了如何使用React Native开发原生App。

第8章介绍了使用React Native和Node.js开发企业内部通讯录应用——“百灵鸟”App。

第9章介绍了使用React Native Geolocation API和高德地图API开发LBS应用——“附近”App。

第10章介绍了使用豆瓣开放API开发一款搜索App，主要包含图书、电影和音乐搜索。

本书特色介绍

本书的特色主要在于理论结合实战，读者不仅可以了解React Native的API和组件，同时可以通过案例和实战深入学习。

源代码

本书包含的代码及其案例可以到<https://github.com/vczero/React-Native-Code>下载或者到图灵社区本书主页免费注册下载。本书创作时间较短，难免会有疏漏，恳请各位读者斧正。

王利华

2015年11月于上海凌空SOHO

目 录

第一部分 基础语法篇

第 1 章 React Native 简介	2
1.1 环境搭建	2
1.1.1 安装 Node.js	2
1.1.2 安装 React Native	6
1.1.3 使用 NVM 管理 Node.js 版本	7
1.1.4 创建项目	8
1.2 从 React 到 React Native	9
1.2.1 React 简介	9
1.2.2 React Native 简介	13
1.3 为什么要使用 React Native	17
1.4 如何学习 React Native	17
1.5 说明	18
第 2 章 React Native 开发基础	19
2.1 flexbox 布局	19
2.1.1 介绍	19
2.1.2 布局模型	20
2.1.3 伸缩容器属性	20
2.1.4 伸缩项目属性	37
2.1.5 在 React Native 中使用 flexbox	46
2.1.6 实例	47
2.2 React 中的 JSX	53
2.2.1 JSX 入门	53
2.2.2 JSX 实战之 ReactJS	59
2.2.3 JSX 实战之 React Native	77

2.3 React Native 开发向导	80
2.3.1 配置文件	80
2.3.2 运行	83
2.3.3 调试	85
2.3.4 内部发布	91
2.4 参考资料	91

第二部分 API 和组件篇

第 3 章 常用组件及其实践	94
3.1 View 组件	94
3.1.1 View 介绍	94
3.1.2 案例：九宫格实现	94
3.2 Text 组件	102
3.2.1 Text 组件介绍	102
3.2.2 案例：网易新闻列表展示	102
3.3 NavigatorIOS 组件	109
3.3.1 NavigatorIOS 组件介绍	109
3.3.2 案例：列表页跳转详情页	110
3.4 TextInput 组件	113
3.4.1 TextInput 组件介绍	113
3.4.2 案例：搜索自动提示	114
3.5 Touchable 类组件	121
3.5.1 TouchableHighlight 组件	121
3.5.2 TouchableOpacity 组件	123
3.5.3 TouchableWithoutFeedback 组件	123
3.6 Image 组件	124

3.6.1	Image 组件介绍	124	4.8.2	NetInfo 示例	159
3.6.2	加载网络图片	124	4.9	CameraRoll	159
3.6.3	加载本地图片	127	4.9.1	CameraRoll 介绍	160
3.7	TabBarIOS 组件	128	4.9.2	CameraRoll 应用	160
3.7.1	TabBarIOS 组件介绍	128	4.9.3	react-native-camera	167
3.7.2	案例: 类 QQ Tab 切换	129	4.10	VibrationIOS	170
3.8	WebView 组件	133	4.11	Geolocation	171
3.8.1	WebView 组件介绍	133	4.11.1	Geolocation 介绍	171
3.8.2	案例: 使用 WebView 组件 加载微博页面	134	4.11.2	Geolocation 应用	172
3.8.3	案例: 新浪微博 OAuth 认证	136	4.12	数据请求	173
第 4 章 常用 API 及其实践			4.12.1	XMLHttpRequest	174
4.1	AppRegistry	138	4.12.2	Fetch	174
4.1.1	AppRegistry 介绍	138	4.13	定时器	175
4.1.2	AppRegistry 示例	138	4.13.1	setTimeout	175
4.2	AsyncStorage	140	4.13.2	setInterval	176
4.2.1	AsyncStorage 介绍	140	4.13.3	setImmediate	176
4.2.2	案例: 购物车	140	4.13.4	使用 requestAnimationFrame 开发进度条	176
4.3	AlertIOS	149	4.13.5	完整代码	177
4.3.1	AlertIOS	149	第 5 章 Native 扩展		
4.3.2	AlertIOS 组件的应用	150	5.1	通信机制	178
4.4	ActionSheetIOS	152	5.1.1	模块配置映射	178
4.4.1	ActionSheetIOS 介绍	152	5.1.2	通信流程	180
4.4.2	ActionSheetIOS 应用	152	5.2	自定义 Native API 组件	183
4.5	PixelRatio	154	5.2.1	模块和方法定义	184
4.5.1	PixelRatio 介绍	154	5.2.2	回调函数	185
4.5.2	PixelRatio 应用	155	5.2.3	线程	187
4.6	AppStateIOS	156	5.2.4	常量导出	188
4.6.1	AppStateIOS 介绍	156	5.2.5	事件	189
4.6.2	AppStateIOS 实例	156	5.2.6	实战	190
4.7	StatusBarIOS	157	5.3	构建 Native UI 组件	196
4.7.1	StatusBarIOS 介绍	157	5.3.1	概述	196
4.7.2	StatusBarIOS 应用	157	5.3.2	UI 组件的定义	196
4.8	NetInfo	158	5.3.3	UI 组件属性	197
4.8.1	NetInfo 介绍	159	5.3.4	组件方法	199

5.3.5 事件	199	8.3.5 创建项目	260
5.3.6 实例	201	8.3.6 app.js 入口文件	262
第 6 章 组件封装	208	8.3.7 加载服务模块到内存	263
6.1 二级菜单组件	208	8.3.8 工具类开发	264
6.1.1 静态组件的实现	209	8.3.9 用户信息接口	265
6.1.2 实现组件的复用和封装	213	8.3.10 公告消息接口	270
6.1.3 应用二级菜单组件	219	8.3.11 建议	272
6.2 日历组件	220	8.4 客户端设计和开发	272
6.2.1 开发日历组件	221	8.4.1 客户端设计	272
6.2.2 应用日历组件	226	8.4.2 工具组件和服务	274
6.3 开源组件	227	8.4.3 添加依赖库	275
6.3.1 react-native-swiper	228	8.4.4 程序入口和登录	277
6.3.2 react-native-modal	231	8.4.5 联系人列表	286
第三部分 App 更新和上架篇		8.4.6 公告功能	294
第 7 章 热更新和上架	236	8.4.7 管理功能	299
7.1 动态更新	236	8.4.8 关于	316
7.1.1 初始化设置	236	8.4.9 建议	318
7.1.2 更新逻辑	238	第 9 章 基于 LBS 的应用开发	319
7.1.3 回滚策略	240	9.1 功能设计	319
7.2 App 上架	240	9.1.1 需求确定	319
7.2.1 证书生成	241	9.1.2 开发目录结构	320
7.2.2 注册应用	248	9.2 程序入口和工具模块	321
7.2.3 上传应用	249	9.2.1 注册应用程序	321
第四部分 实战篇		9.2.2 工具模块	322
第 8 章 企业内部通讯录应用开发	254	9.2.3 Nearby 组件入口	323
8.1 需求提出	254	9.3 列表组件开发	327
8.2 技术架构	255	9.3.1 通用列表组件开发	327
8.3 服务器端设计和开发	256	9.3.2 完成列表页	334
8.3.1 服务器端整体设计	257	9.4 详情页组件开发	336
8.3.2 用户数据模型设计	257	9.5 WebView 地图模块开发	338
8.3.3 公告数据模型设计	259	9.6 综合效果	341
8.3.4 服务路由设计	260	第 10 章 豆瓣 App	342
		10.1 豆瓣 API	342
		10.1.1 熟悉豆瓣 API	342

4 ▶ 目 录

10.1.2	图书、电影、音乐 API	343	10.3.5	WebView 封装	352
10.2	应用设计	345	10.3.6	搜索框封装	353
10.2.1	功能设计	345	10.4	功能开发	354
10.2.2	模块划分	345	10.4.1	入口组件	354
10.3	公共模块开发	347	10.4.2	图书列表页开发	356
10.3.1	工具类开发	347	10.4.3	图书详情页开发	361
10.3.2	服务列表	348	10.4.4	电影模块开发	363
10.3.3	Navigator 封装	349	10.4.5	音乐模块开发	368
10.3.4	公共头封装	350	10.5	完成豆搜 App	372

第一部分

基础语法篇

- 第1章 React Native简介
- 第2章 React Native开发基础

第 1 章

React Native简介

React Native一经Facebook开源，就引起了业界的关注，越来越多的开发者开始尝试在生产中使用它。React Native为JavaScript开发跨终端应用提供了更加丰富的想象空间。下面就开始我们的React Native开发之旅吧。

1.1 环境搭建

工欲善其事，必先利其器。首先，我们需要搭建开发环境（整本书都是以Mac OS X系统为基础的）。React Native主要依赖的环境如下所示。

- Mac OS X操作系统。
- 推荐使用Xcode 6.4或者更高版本。
- 安装Node.js 4.0或者最新版本。
- 建议使用Homebrew安装：watchman和flow。

下面我们一步步来安装开发环境。

1.1.1 安装Node.js

打开浏览器，在浏览器中输入地址<https://nodejs.org/>，此时打开的是Node.js的官网，从中可以看到最新的版本以及下载按钮。

这里下载的是node-v4.1.0.pkg。等待下载完成后，我们开始安装Node.js。双击node-v4.1.0.pkg文件，将弹出如图1-1所示的界面。