HZ BOOKS
华章IT

区块链技术丛书

[PACKT] PUBLISHING

基于以太坊的区块链项目开发必备参考，微软亚洲研究院区块链负责人、IBM区块链社区发起人倾情翻译

区块链项目开发指南

[印度] 纳拉扬·普鲁斯蒂（Narayan Prusty）◎著
朱轩彤 闫莺 董宁◎译

BUILDING BLOCKCHAIN PROJECTS

Develop Real-time Practical DApps Using Ethereum and JavaScript

机械工业出版社
China Machine Press

作者简介

纳拉扬 · 普鲁斯蒂（Narayan Prusty）

在 18 岁时就创建了一个 MP3 搜索引擎，他是一位身兼多职的开发人员，主攻区块链和 JavaScript，倾向于使用以太坊、比特币、超级分类、IPFS 等构建去中心化的应用程序。其所编写的可扩展应用程序广泛应用于印度、新加坡、美国等国家的初创公司、企业以及政府部门。Narayan Prusty 目前供职于迪拜的阿联酋国家银行的区块链企业。他著有《Learning ECMAScript 6》和《Modern JavaScript Applications》。

区块链
技术丛书

区块链项目开发指南

[印度] 纳拉扬·普鲁斯蒂 (Narayan Prusty)◎著
朱轩彤 闫莺 董宁◎译

BUILDING BLOCKCHAIN PROJECTS

Develop Real-time Practical DApps Using Ethereum and JavaScript

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

区块链项目开发指南 /（印度）纳拉扬·普鲁斯蒂（Narayan Prusty）著；朱轩彤，闫莺，董宁译 . —北京：机械工业出版社，2017.11（2018.2 重印）
（区块链技术丛书）
书名原文：Building Blockchain Projects: Develop Real-time Practical DApps Using Ethereum and JavaScript

ISBN 978-7-111-58400-1

I. 区… II. ①纳… ②朱… ③闫… ④董… III. 电子商务－支付方式－研究 IV. F713.36

中国版本图书馆 CIP 数据核字（2017）第 273099 号

本书版权登记号：图字 01-2017-6963

区块链项目开发指南

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：吴晋瑜 责任校对：殷 虹
印 刷：中国电影出版社印刷厂 版 次：2018 年 2 月第 1 版第 2 次印刷
开 本：186mm × 240mm 1/16 印 张：13
书 号：ISBN 978-7-111-58400-1 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

The Translator's Words 译者序

从去年开始，“区块链”成了一个高热度的关键字，受到各行各业的关注。越来越多的人渴望用区块链这一变革性技术来解决商业中的关键问题。自然，有更多的人渴望深入了解和运用区块链技术甚至开发自己的区块链应用。最近一年多，总是有很多朋友和学生问我：“如何学习以太坊？有什么资料推荐吗？”通常我的回答就是：“从以太坊白皮书和黄皮书看起吧。”显然，仅仅精读两篇文章是不够的，要想对区块链有进一步的认识，还需要更多的知识储备。但是目前国内很难找到一本系统介绍区块链技术和开发平台的书籍。作为在区块链领域具有较高知名度和丰富从业经验的专家团队，我们非常希望能给大家提供一套系统的学习资料。

作为这场区块链技术热潮的弄潮儿，以太坊是最先进区块链技术的代表。以太坊的社区和开发工具都相对比较完善和活跃。正因如此，很多企业级区块链解决方案都在积极地拥抱以太坊。但是很遗憾，系统介绍以太坊的中文资料非常匮乏。首次接触到《Building Blockchain Projects》这本英文书后，我们便感觉它是关于以太坊在广度上难得的技术资料，于是想尽快呈现给国内的读者。在翻译过程中，我们在保证表述流畅的同时，对原著的内容进行了验证，并对其中的错误进行了修正。因此，本书应该比英文原版书更加易懂和准确。

在本书中，读者将了解如何编写智能合约、如何用 JavaScript 开发以太坊程序，以及如何为区块链创建端到端应用。

本书具有如下特点：

- “学生导向”，跟着这本书可以由浅及深地学习以太坊技术应用。
- 给出了多个真实的以太坊智能合约编写示例，可帮助初学者迅速上手编写代码。

❑ 通俗易懂，讲解细致，方便自学。

在翻译本书的同时，我们的团队没有停止前进的脚步。我们不断努力，以求在技术深度上更进一步。读者掌握本书的内容后，可以阅读我们即将于近期出版的其他关于以太坊和 Hyperledger 的书，以加深对区块链的关键技术的认识。详情请见我们的微信公众号“智链 ChainNova”。

Preface 前　言

区块链是一个防篡改的去中心化账本，其中包含不断增长的数据记录列表。每个用户都可以连接到网络，发送新的交易、验证交易和创建新的区块。

本书将阐释区块链的概念，讲述其如何保证数据真实性，以及如何使用以太坊创建现实世界的区块链项目。通过有趣的现实世界案例，读者将了解如何编写完全按照程序运行、没有欺诈、没有中心机构或者第三方干预的智能合约，并学习如何创建端到端的区块链应用。本书还将介绍加密货币中的密码学、以太币安全、挖矿、智能合约和 Solidity 等概念。

区块链是比特币中最有创造性的技术，是记录比特币交易的公共账本。

本书内容

第 1 章阐释 DApp 的概念，并简述其工作原理。

第 2 章阐释以太坊的工作原理。

第 3 章阐释如何编写智能合约和使用 geth 交互接口来部署合约，以及使用 web3.js 广播交易。

第 4 章介绍 web.3js 的概念及其导入方法、连接到 geth 的方法，并阐释了如何在 node.js 或者客户端 JavaScript 使用它。

第 5 章阐释如何创建钱包服务，以方便用户创建和管理以太坊钱包，甚至离线创建和管理钱包。我们将专门使用 LightWallet 库实现。

第 6 章展示如何使用 web3.js 编译智能合约，以及使用 web3.js 和 EthereumJS 部署智能合约。

第 7 章阐释如何使用 Oraclize 从以太坊智能合约发出 HTTP 请求，以访问万维网中的数据。我们还将学习访问存储在 IPFS 中的文件、使用字符串库处理字符串等方法。

第 8 章阐释如何使用 truffle。truffle 将使创建企业级 DApp 变得容易。我们将通过创建代币来学习 truffle。

第 9 章阐释创建联盟区块链的方法。

设备环境

Windows 7 SP1+、Windows 8、Windows 10 或者 Mac OS X 10.8+。

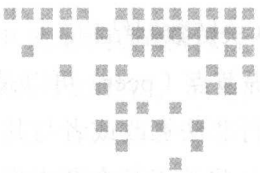
读者对象

本书适合想使用区块链和以太坊创建防篡改数据 (和交易) 应用的 JavaScript 开发人员阅读，也适合对密码学及其逻辑以及相关数据库感兴趣的人阅读。

下载实例代码

可以从 http://www.packtpub.com 下载本书的实例代码文件。如果您在其他地方购买了本书，可以访问 http://www.hzbook.com 注册并下载。

Contents 目　录



第 1 章 Chapter 1

去中心化应用

我们以前用过的所有互联网应用几乎都是中心化的，即每个应用的服务端由一个特定企业或个人所有。长期以来，开发人员创建中心化应用，用户使用中心化应用。但是中心化应用存在一些问题，包括不透明、有单点故障、不能防止网络审查等，导致几乎不可能创建某些特定类型的应用。为了解决这些问题，一项新的技术诞生了，它创建以网络为基础的**去中心化应用（DApp）**。在本章中，我们将学习去中心化应用。

在本章中，我们将讲解以下内容：

- 什么是 DApp。
- 去中心化、中心化和分布式应用之间的区别。
- 中心化和去中心化应用的优点和缺点。
- 概述一些最热门的 DApp 所使用的数据结构、算法和协议。
- 学习一些创建在其他 DApp 之上的流行 DApp。

1.1 什么是 DApp

DApp 是一种互联网应用，其后端在去中心化的点对点网络上运行，且其源代码是开源的。网络中不存在能够完全控制 DApp 的节点。

根据 DApp 的功能不同，使用不同的数据结构来存储应用数据。例如，比特币

DApp 使用区块链数据结构。

这些对等节点（peer）可以是网络中的任何计算节点，因此，发现和防止节点对应用数据进行非法篡改或者与其他人分享错误信息是一个重要挑战，所以需要对等节点之间有一些关于某个节点发布的数据是否正确的共识。在 DApp 中，没有一个中心服务器来协调节点，或者决定什么是对、什么是错，因此应对这个挑战确实不容易。一致性协议（concensus protocol）可用于解决这个问题。不同的 DApp 通常使用不同数据结构类型的共识协议，例如比特币使用工作量证明协议（PoW）来达成共识。

为了让用户 () 使用 DApp，每一个 DApp 都需要一个客户端（client）。使用 DApp 时，用户首先需要运行 DApp 中自己的节点服务端，然后将客户端连接至节点服务端。DApp 的节点只提供应用程序编程接口（Application Programming Interface，API），并允许开发者社区使用 API 开发多种客户端。一些 DApp 开发人员会提供一个官方的客户端。DApp 客户端应该是开源的，并可以被下载使用，否则整个去中心化的想法就失败了。

但是建立客户端架构比较麻烦，如果用户不是开发人员，就更麻烦。因此，客户端通常作为服务和 / 或节点形式出现，以便让使用 DApp 的过程更容易。

什么是分布式应用？

分布式应用是指应用分布在多个服务端上，而非只有一个服务端。当应用数据和通信量变得巨大，且应用的停机时间难以承受时，分布式是必要的。在分布式应用中，数据在多个服务端中备份，以具有较高可用性。中心化应用可能是分布式的，也可能不是分布式的，但去中心化应用肯定是分布式的。例如 Google、Facebook、Slack、DropBox 等是分布式的，而简单的投资组合网站或者个人微博通常不是分布式的，除非通信量很大。

1.1.1 去中心化应用的优点

中心化应用的一些优点如下：

- DApp 能容错，没有单点故障，因为它们默认是分布式的。
- 防止某单一机构的干扰。因为没有一个中心机构，任何第三方机构无法向中心机构施压逼迫其删除一些内容。甚至没有单一机构能关闭应用的域名或者

IP 地址，因为 DApp 不是通过一个特定的 IP 地址或者域名访问的。或许某些机构可以通过 IP 地址追踪网络中的单个节点并关闭它，但是如果网络很庞大，则几乎不可能关闭应用。

- 用户容易相信该应用。因为它不是由某个通过欺骗用户来牟利的机构所控制的。

1.1.2　去中心化应用的缺点

显然，每个系统都不是完美的。去中心化应用的一些缺点如下：

- 修改 bug 或者更新 DApp 很困难，因为网络中的每一个节点都需要更新其节点软件。
- 一些应用要求验证用户身份（即 KYC），却没有中心化的机构来验证用户身份，开发应用时会遇到问题。
- 创建去中心化应用比较困难，因为它们应用复杂的协议达成共识，且必须从最开始就自行创建并扩大规模。所以我们不能仅仅实现一个想法，然后不断添加功能，使其规模扩大。
- 应用通常独立于第三方 API，以获取或者存储数据。DApp 不能依赖中心化应用 API，但是可以依赖其他 DApp。因为目前 DApp 的生态圈还不太大，所以创建起来比较困难。尽管 DApp 理论上可以依赖其他 DApp，但在实践中紧密融合 DApp 仍比较困难。

1.2　去中心化自治组织

一般来说，被签署的文件可以代表组织，而且政府能对它们产生影响。根据组织类型的不同，组织可能有股东，也可能没有股东。

去中心化自治组织（Decentralized Autonomous Organization，DAO）是由计算机程序代表的组织（即组织根据程序中写明的规则运行），完全透明，完全由股东控制，不受政府影响。

为了达到这些目标，我们需要把 DAO 作为 DApp 来开发。因此，我们可以说 DAO 是 DApp 的一个子类。

Dash 和 DAC 是 DAO 的一些例子。

什么是去中心化自治公司（DAC）？

DAC 和 DAO 尚无很明显的差别。很多人认为它们是一样的，有些人则在 DAO 为股东谋取利益时将 DAC 定义为 DAO。

1.3 DApp 中的用户身份

DApp 的主要优点之一是它一般能保证用户的匿名性，但是许多应用要求用户必须经过身份验证这个过程才能使用应用。因为 DApp 中没有中央机构，验证用户身份成了一个挑战。

在中心化应用中，人们要求用户提交特定的扫描文件、OTP 验证等，再验证用户身份。该过程称为 Know Your Customer（KYC）。但是由于 DApp 中没有人负责验证用户身份，所以 DApp 不得不自己验证用户身份。DApp 显然不能理解和验证扫描文档，也不能发送短信，因此需要用户提供那些它们可以理解和验证的数字标识。主要问题是几乎没有 DApp 有数字身份，只有少数人知道如何得到数字身份。

数字身份有多种形式。目前最受推崇、最热门的形式就是数字证书。数字证书（也称为公钥证书或者标识证书）是一个用来证明公钥所有权的电子文档。基本上，一个用户拥有私钥（private key）、公钥（public key）和数字证书（digital certificate）。私钥是秘密的，用户不应当与其他人分享；公钥可以与其他人分享；数字证书包含公钥和谁拥有公钥的信息。显然，生产这种证书并不难，因此数字证书总是由用户可以信任的授权机关颁发。数字证书有一个加密部分是用证书颁发机构（certificate authority）的私钥加密的。为了验证证书的真实性，我们只需要使用证书颁发机构的公钥解码该部分，如果成功解码，那么证书就是合法的。

即使用户成功获得了数字身份并得到 DApp 验证，还是有一个关键问题，那就是有各种各样的数字证书颁发机构。为了验证一个数字证书，我们需要该证书颁发机构的公钥。掌握所有证书颁发机构的公钥、更新 / 添加新的证书颁发机构的公钥是很困难的。因此，数字身份验证程序通常在客户端里，这样可以方便更新。但仅把验证程序转移到客户端并不能彻底解决问题，因为有很多颁发数字证书的机构，跟踪所有机构并把它们加到客户端是很麻烦的。