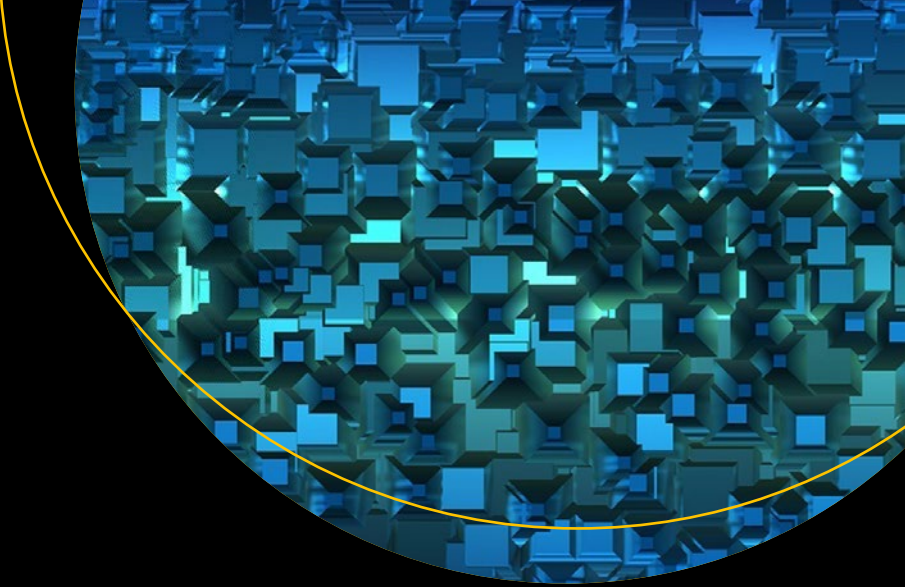




Introducing Ethereum and Solidity

Foundations of Cryptocurrency and
Blockchain Programming for Beginners

Chris Dannen

Apress®

Introducing Ethereum and Solidity

Foundations of Cryptocurrency
and Blockchain Programming for
Beginners



Chris Dannen

Apress®

Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners

Chris Dannen
Brooklyn, New York, USA

ISBN-13 (pbk): 978-1-4842-2534-9
DOI 10.1007/978-1-4842-2535-6

ISBN-13 (electronic): 978-1-4842-2535-6

Library of Congress Control Number: 2017936045

Copyright © 2017 by Chris Dannen

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image, we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director: Welmoed Spahr
Editorial Director: Todd Green
Acquisitions Editor: Louise Corrigan
Development Editor: James Markham
Technical Reviewer: Massimo Nardone
Coordinating Editor: Nancy Chen
Copy Editor: Sharon Wilkey
Compositor: SPi Global
Indexer: SPi Global
Artist: SPi Global

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail rights@apress.com, or visit <http://www.apress.com/rights-permissions>.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484225349. For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper

*Many thanks to Brandon Buchanan, Christopher McClellan,
Dr. Solomon Lederer, and the entire Iterative Instinct team
for their support and enthusiasm. Thanks also to
Joseph Lubin and the team at ConsenSys for
acting as a sounding board during
the writing of this book.*

Contents at a Glance

About the Author	xix
About the Technical Reviewer	xxi
■ Chapter 1: Bridging the Blockchain Knowledge Gap.....	1
■ Chapter 2: The Mist Browser	21
■ Chapter 3: The EVM	47
■ Chapter 4: Solidity Programming	69
■ Chapter 5: Smart Contracts and Tokens	89
■ Chapter 6: Mining Ether.....	111
■ Chapter 7: Cryptoeconomics Survey	139
■ Chapter 8: Dapp Deployment.....	149
■ Chapter 9: Creating Private Chains	159
■ Chapter 10: Use Cases.....	165
■ Chapter 11: Advanced Concepts.....	173
Index.....	181

Contents

About the Author	xix
About the Technical Reviewer	xxi
■ Chapter 1: Bridging the Blockchain Knowledge Gap	1
Blockchain Roll Call!.....	1
What Ethereum Does.....	2
Three Parts of a Blockchain	4
Ethereum Assumes Many Chains	5
This Is a Scam, Just Like Bitcoin!	5
Ether as a Currency and Commodity	6
Gresham's Law	6
The Path to Better Money	7
Cryptoeconomics and Security.....	7
Back to the Good Old Days	8
Cryptochaos.....	8
The Power Is in the Protocol	8
You Can Build Trustless Systems.....	9
What Smart Contracts (Really) Do	10
Objects and Methods for Value.....	10
Just Add Commerce	11
Content Creation.....	11
Where's the Data?	12
What Is Mining?	12
Ether and Electricity Prices.....	12

Going Inside the EVM	13
The Mist Browser	13
Browser vs. Wallet or Keychain	13
Solidity Is Kind of Like JavaScript, But	13
What Ethereum Is Good For.....	14
A Critical Take	14
State of Smart Contract Development Today	15
Deciding Where You Fit In.....	16
A Note to New Programmers	17
Ethereum Is Free and Open Source	17
The EVM Is Here to Stay	17
What You Can Build Today	18
Private and Public Chains	18
The Promise of Decentralized Databases	19
What's Next: New Ways of Working	20
Summary	20
■ Chapter 2: The Mist Browser	21
Wallets as a Computing Metaphor	22
Your Address Is What?	22
Where Is My Ether?	23
The Bank Teller Metaphor.....	24
In Cryptocurrency, You Hold Your Own Assets.....	24
Visualizing Ethereum Transactions	24
Breaking with Banking History	26
How Encryption Leads to Trust.....	26
System Requirements	28

More about Eth.guide and This Book	28
Tools for Developers	29
CLI Nodes.....	29
Recommended: Using Parity with Geth	30
Finally, into the Mist!	30
Downloading and Installing Mist	30
Configuring Mist	32
Finding Your New Address.....	36
Sending and Receiving Ether.....	36
Understanding Ethereum Account Types	38
Backing Up and Restoring Your Keys.....	39
Using Paper Wallets.....	40
Using Mobile Wallets	40
Working with Messages and Transactions	42
So, What Is a Blockchain?	43
Paying for Transactions	43
Understanding Denominations	44
Getting Ether.....	44
Anonymity in Cryptocurrency	45
Blockchain Explorers	45
Summary	46
■ Chapter 3: The EVM	47
The Central Bank Network of Yesterday.....	47
What are Virtual Machines, Exactly?	48
The Role of the Ethereum Protocol in Banking	48
Anyone Can Make a Banking Platform	48

What the EVM Does	49
EVM Applications Are Called Smart Contracts.....	51
The Name “Smart Contracts”	51
The EVM Runs Bytecode	51
Understanding State Machines	51
Digital vs. Analog	51
“State-ments”	52
Data’s Role in State	53
How the Guts of the EVM Work.....	53
The EVM Constantly Checks for Transactions.....	54
Creating a Common Machine Narrative of What Happened.....	54
Cryptographic Hashing	55
What Hash Functions (or Hash Algorithms) Do	55
Blocks: The History of State Changes.....	55
Understanding Block Time.....	56
The Drawbacks of Short Blocks.....	56
“Solo Node” Blockchain	56
Distributed Security	57
Mining’s Place in the State Transition Function	57
Renting Time on the EVM	58
Hello, Gas	58
Why Is Gas So Important?.....	59
Why Isn’t Gas Priced in Ether?.....	59
Fees as Regulation	59
Working with Gas	60
Gas Specifics	60
How Gas Relates to Scaling the System.....	60

Accounts, Transactions, and Messages.....	61
Externally Owned Accounts	61
Contract Accounts	61
Transactions and Messages	62
Characteristics of Transactions	62
Characteristics of Messages	62
Estimating Gas Fees for Operations	63
Opcodes in the EVM	64
Summary	67
■ Chapter 4: Solidity Programming	69
Primer.....	69
Global Banking Made (Almost) Real	70
Extra-Large Infrastructure	70
Worldwide Currency?	70
Complementary Currency.....	71
The Promise of Solidity.....	71
Browser Compiler	72
Learning to Program the EVM	72
Easy Deployment	73
The Case for Writing Business Logic in Solidity.....	74
Code, Deploy, Relax.....	74
Design Rationale	74
Writing Loops in Solidity	75
Expressiveness and Security	76
The Importance of Formal Proofs.....	76
Historical Impact of a Shared Global Resource	76
How Attackers Bring Down Communities	77
Hypothetical Attack Written in Solidity	77

Automated Proofs to the Rescue?	78
Determinism in Practice	78
Lost in Translation.....	78
Testing, Testing, Testing	79
Command Line Optional!	79
Formatting Solidity Files	81
Tips for Reading Code	81
Statements and Expressions in Solidity	82
What Is an Expression?	82
What Is a Statement?	82
Functions, Public and Private	82
Value Types	83
Booleans.....	83
Signed and Unsigned Integers.....	83
Addresses.....	83
Members of Addresses.....	83
Address-Related Keywords	84
Less-Common Value Types	84
Complex (Reference) Types	84
Global Special Variables, Units, and Functions	85
Block and Transaction Properties	85
Operators Cheat Sheet.....	86
Global Functions	87
Exceptions and Inheritance	88
Summary	88
■ Chapter 5: Smart Contracts and Tokens	89
EVM as Back End.....	89
Smart Contracts to Dapps.....	90

Assets Backed by Anything	90
Bartering with Fiat Currency.....	90
Ether as Glass Beads.....	90
Cryptocurrency Is a Measure of Time.....	91
Asset Ownership and Civilization	92
Coins are Collectibles	93
The Function of Collectibles in Human Systems	94
Early Counterfeiting.....	95
Jewelry and Art as Money	95
The Step Toward Banknotes	95
Platforms for High-Value Digital Collectibles	96
Tokens Are a Category of Smart Contract	97
Tokens as Social Contracts.....	97
Tokens Are a Great First App.....	98
Creating a Token on the Testnet	98
Getting Test Ether from the Faucet.....	99
Registering Your Tokens	106
Deploying Your First Contract	107
Same House, Different Address	108
Playing with Contracts	110
Summary	110
■ Chapter 6: Mining Ether.....	111
What's the Point?	111
Ether's Source	112
Defining Mining	112

Versions of the Truth.....	113
Difficulty, Self-Regulation, and the Race for Profit	114
How Proof of Work Helps Regulate Block Time	115
What's Going on with the DAG and Nonce?.....	116
All This for Faster Blocks?.....	117
Making Fast Blocks Work	117
How Ethereum Uses Stale Blocks	118
Uncle Rules and Rewards.....	119
The Difficulty Bomb.....	119
Miner's Winning Payout Structure	120
Limits on Ancestry	120
The Block Processing Play by Play	120
Evaluating the Ancestry of Blocks and Transactions.....	121
How Ethereum and Bitcoin Use Trees	122
Merkle-Patricia Trees.....	122
Contents of an Ethereum Block Header.....	123
Forking	123
Mining Tutorial.....	124
Installing Geth on macOS	125
Installing Geth on Windows	125
Getting Comfortable with the Command Line.....	125
Installing Geth on Ubuntu 14.04	126
Executing Commands in the EVM via the Geth Console.....	128
Launching Geth with Flags.....	131
Fire Up Your Miner!	132

Mining on the Testnet.....	134
GPU Mining Rigs.....	134
Mining on a Pool with Multiple GPUs	136
Summary.....	136
■ Chapter 7: Cryptoeconomics Survey	139
How We Got Here.....	139
New Technologies Create New Economies.....	140
Rules of the Game	141
Why Is Cryptoeconomics Useful?	141
Understanding Hashing vs. Encryption.....	142
Encryption	142
Hashing.....	143
Why the Speed of Blocks Matters	144
Ether Issuance Scheme.....	144
Common Attack Scenarios	145
Social Proof Between Machines	146
Security as the Network Scales.....	146
More About Cryptoeconomics	147
Summary.....	147
■ Chapter 8: Dapp Deployment	149
Seven Ways to Think About Smart Contracts	150
Dapp Contract Data Models.....	150
How an EVM Back End Talks to a JS Front End	151
JSON-RPC.....	151
Web 3 Is Here (Almost).....	152

Experimenting with the JavaScript API	153
Using Geth for Dapp Deployment.....	153
Using Meteor with the EVM	154
Install Web3.js to Build an Ethereum-Enabled Web Application	154
Executing Contracts in the Console.....	155
How Contracts Expose an Interface.....	155
Recommendations for Prototyping	156
Third-Party Deployment Libraries	156
Summary	157
■ Chapter 9: Creating Private Chains	159
Private and Permissioned Chains.....	159
Setting Up a Local Private Chain	160
Optional Flags to Use with New Chains.....	162
Private Blockchains in Production Usage.....	162
Summary	163
■ Chapter 10: Use Cases	165
Chains Everywhere.....	165
The Internet of Ethereum Things	166
Retail and E-Commerce	167
Community and Government Financing	167
Human and Organizational Behavior	168
Financial and Insurance Applications.....	169
Inventory and Accounting Systems	170
Software Development.....	171

Gaming, Gambling, and Investing	171
Summary	172
■ Chapter 11: Advanced Concepts.....	173
Who Is Leading Software Developers Toward Decentralization?	173
Vitalik's Best Technical Blog Posts	174
The Ethereum Release Schedule.....	174
Whisper (Messaging)	175
Swarm (Content Addressing)	175
What the Future Holds.....	176
Other Interesting Innovations	177
Full Ethereum Roadmap	177
Frontier Release (2015)	177
Homestead Release (2016).....	178
Metropolis (2017)	178
Serenity (2018)	178
Summary	178
Index.....	181

About the Author



Chris Dannen is a partner and founder at Iterative Instinct, a hybrid investment fund focused on cryptocurrency trading and seed-stage venture investments. He first began working with Bitcoin and Ethereum as a miner, and became gradually more enthralled in researching how smart contracts could be used to automate business logic and create new kinds of experiences with software. He was formerly a corporate strategist for Fortune 500 companies. A self-taught programmer in Objective-C and JavaScript, he holds one computer hardware patent. This is his fourth book. Chris is an avid traveler who has trekked across 20 countries, bicycled from Rome to Barcelona in 30 days, and summited Mount Fuji in under six hours. He was formerly a senior editor at *Fast Company* and today consults on technical content for major publishers such as *Quartz* and *Bloomberg*. He graduated from the University of Virginia and now resides in New York, NY.

About the Technical Reviewer



Massimo Nardone has more than 22 years of experience in security, web/mobile development, and cloud and IT architecture. His true IT passions are security and Android. He has been programming and teaching how to program with Android, Perl, PHP, Java, VB, Python, C/C++, and MySQL for more than 20 years. He holds a master of science degree in computing science from the University of Salerno, Italy.

He has worked as a project manager, software engineer, research engineer, chief security architect, information security manager, PCI/SCADA auditor and senior lead IT security/cloud/SCADA architect for many years. Technical skills include security, Android, cloud, Java, MySQL, Drupal, Cobol, Perl, web and mobile development, MongoDB, D3, Joomla, Couchbase, C/C++, WebGL, Python, Pro Rails, Django CMS, Jekyll, Scratch, and more.

He currently works as chief information security officer (CISO) for CargotecOyj. He worked as visiting lecturer and supervisor for exercises at the Networking Laboratory of the Helsinki University of Technology (Aalto University). He holds four international patents (PKI, SIP, SAML and Proxy areas).

Massimo has reviewed more than 40 IT books for various publishing companies. He is the coauthor of *Pro Android Games* (Apress, 2015).

CHAPTER 1



Bridging the Blockchain Knowledge Gap

Acclimating to the fast-moving blockchain world can be challenging. This book is your guide. Before we get started, let's define some of the terms you'll encounter ahead

A *blockchain* is a fully-distributed, peer-to-peer software network which makes use of cryptography to securely host applications, store data, and easily transfer digital instruments of value that represent real-world money. *Cryptography* is the art of communication via coded messages. In Bitcoin and Ethereum, cryptography is used to conjure one secure computing environment out of thousands of similar machines, running with no central authority and no single owner. With that kind of potential, it's obvious why the technology has been subject to unprecedented speculation, hype, confusion, and prognostication.

The term "Ethereum" can be used refer to three distinct things: the Ethereum protocol, the Ethereum network created by computers using the protocol, and the Ethereum project funding development of the aforementioned two. On the heels of Bitcoin, Ethereum has become its own macrocosm, attracting enthusiasts and engineers from numerous industries. Many of civilization's most nagging imperfections could become the domain of blockchain's killer apps, and the Ethereum protocol (which was derived from Bitcoin, and extended) is widely considered to be the network where these "distributed" apps will spring up. For developers, designers, and product managers, there's no better time to begin prototyping applications for the Ethereum network.

Blockchain Roll Call!

Two big groups of thinkers are interested in blockchain systems, and Ethereum more specifically: application developers interested in building products and services, and nonprogrammers who are curious about the potential of Ethereum, perhaps owing to work or interest in financial services, consulting, insurance, law, game creation,

government, logistics, product design, or IT.¹ This book is similarly cross-disciplinary. It provides a contextual guide for programmers and non-programmers alike to develop ideas about what to build, and how to build it. It fills a gap between computer science, economics, financial services, and where necessary, banking history.

For programmers, the challenging thing about Ethereum isn't usually the code; like most open source software projects, this one has on-ramps for people who already program in other environments. Rather, the challenge is wrapping your head around the concept of "cryptoeconomics," or the system of incentives and disincentives which secure the network.

For nonprogrammers, the challenge is divining how the ecosystem will develop, and how you fit in. Claims that blockchains will modernize the banking system, revolutionize insurance, and lay waste to counterfeiting may be overblown—but by how much?²

What Ethereum Does

In the abstract, open source blockchain networks such as Ethereum and Bitcoin are kits that allow you to pop up an economic system in software, complete with account management and a native unit of exchange to pass between accounts. Kind of like the game Monopoly. People call these native units of exchange *coins*, *tokens*, or *cryptocurrencies*, but they're no different from tokens in any other system: they're a form of money (or *scrip*) that is usable only within that system.

Blockchains work something like mesh networks or local area networks (LANs); they are merely connected to other "peer" computers running the same software. When you want to make one of these peer-to-peer (P2P) networks accessible through a web browser, you need to use special software libraries such as Web3.js to connect an application's front end (the GUI you see in a browser), via JavaScript APIs, to its back end (the blockchain).

In Ethereum, you can take this concept one step further by easily writing financial contracts with other users inside the system. As you'll see, these financial contracts are called *smart contracts*.

The key component is this idea of a Turing-complete blockchain. ... As a data structure, it works kind of the same way that Bitcoin works, except the difference in Ethereum is, it has this built-in programming language.

—Vitalik Buterin, inventor of Ethereum³

In Ethereum, smart contracts are written in the programming language Solidity, which you'll learn about in Chapter 4. Turing completeness was an advantage that many developers quickly latched onto, but more important is Ethereum's ability to save state. In computing, a simple definition of a *stateful* system is one that can detect changes to information and remember them over time.

¹Ethereum Blog, "Visions, Part 1: The Value of Blockchain Technology," <https://blog.ethereum.org/2015/04/13/visions-part-1-the-value-of-blockchain-technology/>, 2015.

²American Banker, "Blockchain Won't Make Banks Any Nimbler," www.americanbanker.com/bankthink/blockchain-wont-make-banks-any-nimbler-1079190-1.html, 2016.

³YouTube, "Technologies That Will Decentralize the World," www.youtube.com/watch?v=er-k3ehpFaM&feature=share, 2016.