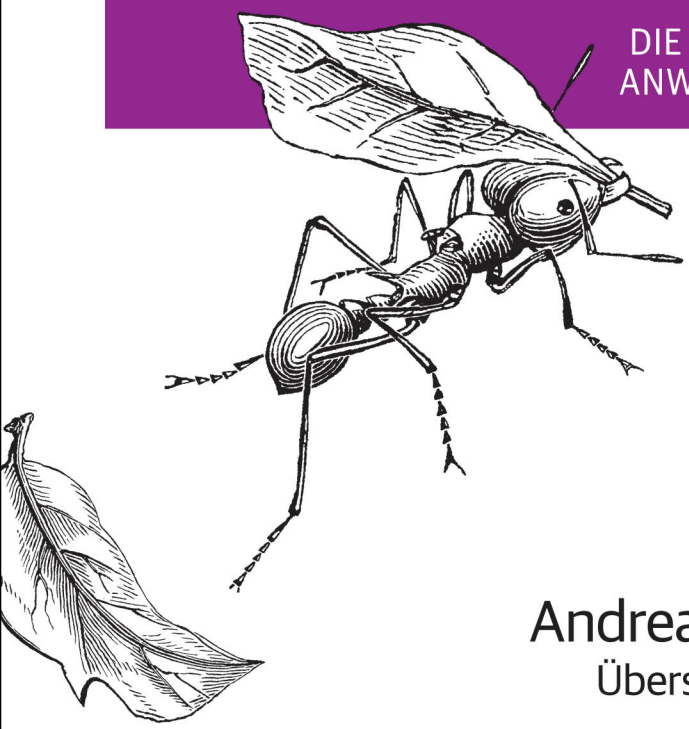
O'REILLY®

2. Auflage
US-Bestseller

Bitcoin & Blockchain

Grundlagen und Programmierung

DIE BLOCKCHAIN VERSTEHEN
ANWENDUNGEN ENTWICKELN



Andreas M. Antonopoulos
Übersetzung von Peter Klicman

2. AUFLAGE

Bitcoin und Blockchain – Grundlagen und Programmierung

Die Blockchain verstehen, Anwendungen entwickeln

Andreas M. Antonopoulos

Deutsche Übersetzung von
Peter Klicman

O'REILLY®

Andreas M. Antonopoulos

Lektorat: Ariane Hesse
Übersetzung: Peter Klicman
Korrektorat: Sibylle Feldmann, www.richtiger-text.de
Herstellung: Susanne Bröckelmann
Umschlaggestaltung: Michael Oréal, www.oreal.de
Satz: III-satz, www.drei-satz.de
Druck und Bindung: M.P. Media-Print Informationstechnologie GmbH, mediaprint-druckerei.de

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über *http://dnb.d-nb.de* abrufbar.

ISBN
Print: 978-3-96009-071-7
PDF: 978-3-96010-171-0
ePub: 978-3-96010-172-7
mobi: 978-3-96010-173-4

Dieses Buch erscheint in Kooperation mit O'Reilly Media, Inc. unter dem Imprint »O'REILLY«.
O'REILLY ist ein Markenzeichen und eine eingetragene Marke von O'Reilly Media, Inc. und wird mit Einwilligung des Eigentümers verwendet.

2. Auflage 2018

Wieblinger Weg 17
69123 Heidelberg

Authorized German translation of the English edition of Mastering Bitcoin, 2nd Edition (Second Release: 2017-07-21), ISBN 9781491954386 © 2017 Andreas M. Antonopoulos, LLC.
This translation is published and sold by permission of O'Reilly Media, Inc., which owns or controls all rights to publish and sell the same.

5 4 3 2 1 0

Dedicated to my mum, Theresa (1946–2017)
She taught me to love books and question authority
Thank you, mum

Inhalt

Vorwort

Ein Bitcoin-Buch schreiben

Mitte 2011 stolperte ich das erste Mal über Bitcoin. Meine erste Reaktion war: »Pfft! Nerd-Geld!«, und ich ignorierte es für weitere sechs Monate, ohne seine Bedeutung zu erkennen. Diese Reaktion habe ich bei vielen der klügsten Menschen, die ich kenne, beobachtet, was mich ein bisschen tröstet. Als ich in einer Mailinglistendiskussion das zweite Mal über Bitcoin stolperte, entschied ich mich, das Whitepaper von Satoshi Nakamoto zu lesen, um die maßgebliche Quelle zu studieren und herauszufinden, worum es denn da eigentlich ging. Ich erinnere mich immer noch an den Moment, als ich diese neun Seiten gelesen hatte und begriff, dass Bitcoin nicht einfach eine digitale Währung, sondern ein Vertrauensnetzwerk ist, das die Basis für weit mehr als nur Währungen sein konnte. Die Erkenntnis, dass das »kein Geld, sondern ein dezentralisiertes Vertrauensnetzwerk« ist, schickte mich auf eine viermonatige Reise, in der ich jedes Quäntchen an Informationen über Bitcoin, das ich finden konnte, aufsaugte. Es hatte mich gepackt, und wie besessen verbrachte ich täglich zwölf Stunden und mehr vor dem Bildschirm, in denen ich las, schrieb, programmierte und so viel lernte, wie ich konnte. Nachdem ich aus diesem Zustand wieder erwachte, war ich zehn Kilogramm leichter und hatte mich entschieden, zukünftig an Bitcoin zu arbeiten.

Zwei Jahre später, nachdem ich eine Reihe kleiner Start-ups gegründet hatte, um verschiedene Bitcoin-bezogene Dienste und Produkte zu erforschen, entschied ich, dass es an der Zeit wäre, mein erstes Buch zu schreiben. Bitcoin hatte mich in einen Kreativitätsrausch versetzt und meine Gedanken bestimmt. Das war die aufregendste Technologie, der ich seit Beginn des Internets begegnet war – Zeit also, meine Leidenschaft für diese faszinierende Technologie mit einem breiteren Publikum zu teilen.

Leserkreis

Dieses Buch richtet sich hauptsächlich an Entwickler. Wenn Sie eine Programmiersprache beherrschen, lehrt Sie dieses Buch, wie kryptografische Währungen funktionieren, wie man sie nutzt und wie man Software entwickelt, die mit ihnen

arbeitet. Die ersten Kapitel eignen sich ebenfalls als ausführliche Einführung in Bitcoin für Nichtprogrammierer, also für diejenigen, die die innere Funktionsweise von Bitcoin und Kryptowährungen verstehen wollen.

Warum sind Ameisen auf dem Cover?

Die Blattschneiderameise ist eine Spezies, die in einem Kolonie-Superorganismus ein hochkomplexes Verhalten zeigt. Doch jede einzelne Ameise agiert nach einem Satz einfacher Regeln, die durch soziale Interaktion und das Ausschütten chemischer Duftstoffe (Pheromone) gesteuert wird. Laut (englischer) Wikipedia bilden Blattschneiderameisen nach dem Menschen die größten und komplexesten Tiergesellschaften. Blattschneiderameisen essen keine Blätter, vielmehr nutzen sie sie, um einen Pilz anzubauen, der die zentrale Futterquelle der Kolonie bildet. Diese Ameisen betreiben also Landwirtschaft!

Zwar bilden Ameisen eine kastenbasierte Gesellschaft und haben eine Königin, die für den Nachwuchs sorgt, doch es gibt weder eine zentrale Autorität noch einen Anführer. Das hochgradig intelligente und komplexe Verhalten, das eine aus mehreren Millionen Ameisen bestehende Kolonie zeigt, ist eine emergente Eigenschaft der Interaktion von Individuen in einem sozialen Netzwerk.

Die Natur demonstriert, dass ein dezentralisiertes System robust, komplex und unglaublich ausgereift sein kann, ohne eine zentrale Autorität, eine Hierarchie oder komplexe Teile zu benötigen.

Bitcoin ist ein kunstvolles dezentralisiertes Vertrauensnetzwerk, das eine Vielzahl finanzieller Prozesse unterstützen kann. Dennoch folgt jeder Knoten im Bitcoin-Netzwerk nur einigen wenigen einfachen mathematischen Regeln. Die Interaktion zwischen vielen Knoten führt zu diesem ausgeklügelten Verhalten, nicht die Komplexität eines einzelnen Knotens oder das in ihn gesetzte Vertrauen. Wie eine Ameisenkolonie ist das Bitcoin-Netzwerk ein robustes Netzwerk einfacher Knoten, die einfachen Regeln folgen, um erstaunliche Dinge ohne zentrale Koordinierung zu erreichen.

Verwendete Konventionen

Im Buch folgen wir diesen typografischen Konventionen:

Kursivschrift
: Wird für neue Begriffe, URLs, E-Mail-Adressen, Dateinamen und Dateierweiterungen verwendet.

`Nichtproportionalschrift`
: Wird für Programmlistings verwendet. Im normalen Fließtext werden damit Programmelemente wie Variablen- oder Funktionsnamen, Datenbanken, Datentypen, Umgebungsvariablen, Anweisungen und Schlüsselwörter hervorgehoben.

`Nichtproportionalschrift fett`
: Wird für Befehle oder andere Eingaben verwendet, die Sie wortwörtlich eingeben müssen.

`Nichtproportionalschrift kursiv`
: Wird für Text verwendet, der durch benutzereigene oder durch den Kontext bestimmte Werte ersetzt wird, und für die Kommentare in Listings, um eine bessere Lesbarkeit zu gewährleisten..

Mit diesem Symbol wird ein Tipp oder ein Vorschlag angezeigt.

Mit diesem Symbol wird ein allgemeiner Hinweis angezeigt.

Hiermit wird eine Warnung angezeigt.

Codebeispiele

Die Beispiele sind in Python bzw. C++ geschrieben und verwenden die Kommandozeile Unix-artiger Betriebssysteme wie Linux oder macOS. Alle Code-Snippets finden Sie im Github-Repository (*https://github.com/bitcoinbook/bitcoinbook*) im *code*-Unterverzeichnis des Main-Repository. Laden Sie sich den Buchcode herunter, probieren Sie die Codebeispiele aus oder senden Sie Korrekturen über GitHub.

Alle Code-Snippets können für die meisten Betriebssysteme mit einer minimalen Installation der Compiler und Interpreter für die entsprechenden Sprachen repliziert werden. Wenn nötig, stellen wir grundlegende Installationsanweisungen und schrittweise Beispiele der Ausgaben bereit.

Einige der Code-Snippets wurden für den Druck aufbereitet. In diesen Fällen wurden die Zeilen mit einem Backslash-Zeichen (\) gefolgt von einem Newline-Zeichen getrennt. Wenn Sie mit den Beispielen arbeiten, sollten Sie diese beiden Zeichen entfernen und die Zeilen wieder zusammenfassen. Die Ergebnisse sollten dann denen der Beispiele entsprechen.

Alle Code-Snippets verwenden wann immer möglich reale Werte und Berechnungen. Sie können sich also von Beispiel zu Beispiel vorarbeiten und kommen immer zu den gleichen Ergebnissen wie das Buch. Die privaten Schlüssel und die zugehörigen öffentlichen Schlüssel etwa sind alle echt. Sämtliche Beispieltransaktionen,

Blöcke und Blockchain-Referenzen wurden in die Blockchain eingetragen und sind Teil des öffentlichen »Kassenbuchs«, d.h., man kann sie sich auf jedem Bitcoin-System ansehen.

Verwendung der Codebeispiele

Dieses Buch ist dazu gedacht, Ihnen bei der Erledigung Ihrer Arbeit zu helfen. Im Allgemeinen dürfen Sie den Code in diesem Buch in Ihren eigenen Programmen oder Dokumentationen verwenden. Solange Sie den Code nicht in großem Umfang reproduzieren, brauchen Sie uns nicht um Erlaubnis zu bitten. Zum Beispiel benötigen Sie nicht unsere Erlaubnis, wenn Sie ein Programm unter Zuhilfenahme mehrerer Codestücke aus diesem Buch schreiben. Eine Frage mit einem Zitat oder einem Codebeispiel aus dem Buch zu beantworten, erfordert ebenfalls keine Genehmigung. Signifikante Teile von Beispielcode aus dem Buch für die eigene Produktdokumentation zu verwerten, ist dagegen genehmigungspflichtig.

Wir freuen uns über eine Quellenangabe, verlangen sie aber nicht unbedingt. Zu einer Quellenangabe gehören normalerweise Autor, Titel, Verlagsangabe, Veröffentlichungsjahr und ISBN, hier also: »Andreas M. Antonopoulos, *Mastering Bitcoin*, O'Reilly Media, Inc. 2017, ISBN 978-1-491-95438-6«.

Einige Auflagen dieses Buchs werden unter einer Open-Source-Lizenz wie CC-BY-NC (*https://creativecommons.org/licenses/by-nc/4.0/*) angeboten. In diesem Fall gelten die Bedingungen dieser Lizenz.

Sollten Sie befürchten, dass Ihre Verwendung der Codebeispiele gegen das Fairnessprinzip oder die Genehmigungspflicht verstoßen könnte, nehmen Sie bitte unter *permissions@oreilly.com* Kontakt mit uns auf.

Bitcoin-Adressen und -Transaktionen in diesem Buch

Die Bitcoin-Adressen, Transaktionen, Schlüssel, QR-Codes und Blockchain-Daten in diesem Buch sind größtenteils real. Das bedeutet, dass Sie die Blockchain durchgehen und den größten Teil real nachverfolgen können. Sie können also die Blockchain durchsuchen, sich die in den Beispielen enthaltenen Transaktionen genau ansehen und sie mit Ihren eigenen Skripten/Programmen abrufen.

Beachten Sie aber, dass die in diesem Buch zur Generierung von Adressen verwendeten privaten Schlüssel entweder in diesem Buch abgedruckt oder »verbrannt« wurden. Wenn Sie also Geld an diese Adressen senden, ist es für immer verloren, oder es kann von jedem abgeschöpft werden, der die hier abgedruckten privaten Schlüssel kennt.

Bitte senden Sie keinesfalls Geld an irgendeine der in diesem Buch verwendeten Adressen! Ihr Geld landet bei einem anderen Leser oder ist für immer verloren.

Den Autor kontaktieren

Sie erreichen mich, Andreas M. Antonopoulos, über meine persönliche Website: *https://antonopoulos.com/*

Informationen zu *Mastering Bitcoin*, zur Open Edition und zu Übersetzungen finden Sie hier: *https://bitcoinbook.info/*

Folgen Sie mir auf Facebook: *https://facebook.com/AndreasMAntonopoulos*

Folgen Sie mir auf Twitter: *https://twitter.com/aantonop*

Folgen Sie mir auf LinkedIn: *https://linkedin.com/company/aantonop*

Mein herzlicher Dank an alle meine Förderer, die meine Arbeit durch monatliche Spenden unterstützen. Meine Patreon-Page finden Sie hier: *https://patreon.com/aantonop*

Danksagungen

Dieses Buch spiegelt die Bemühungen und Beiträge vieler Menschen wider. Ich bin sehr dankbar für die Hilfe, die ich von Freunden, Kollegen, aber auch völlig Fremden erhalten habe, die mich dabei unterstützt haben, diesen technischen Leitfaden zu Kryptowährungen und Bitcoin zu schreiben.

Es ist unmöglich, zwischen der Bitcoin-Technologie und der Bitcoin-Community zu unterscheiden, und dieses Buch ist ebenso ein Produkt dieser Community wie ein Buch über die Technologie. Meine Arbeit an diesem Buch wurde vom Anfang bis zum Ende von der Community befürwortet, angefeuert und unterstützt. Neben vielem anderen ermöglichte mir dieses Buch, über zwei Jahre Teil dieser wundervollen Community zu sein, und ich bin mehr als dankbar, in dieser Community akzeptiert worden zu sein. Eine große Menge an Menschen haben das Buch beeinflusst, und es sind viel zu viele, um sie beim Namen zu nennen. Es sind Menschen, die ich auf Konferenzen, Events, Seminaren, Meet-ups, beim Pizza-Plausch oder bei privaten Treffen kennengelernt habe, ebenso wie bei Twitter, auf reddit, bitcointalk.org und GitHub. Jede Idee, Analogie, Frage, Antwort und Erläuterung in diesem Buch wurde an irgendeinem Punkt durch die Community inspiriert, getestet und verbessert. Ich danke euch allen für die Unterstützung. Ohne euch hätte es dieses Buch nie gegeben, und ich bin euch für immer dankbar.

Der Weg zum Autor begann natürlich lange vor dem ersten Buch. Meine erste Sprache war Griechisch (und damit war auch mein erster Unterricht in Griechisch). Deshalb ich belegte im ersten Jahr an der Universität einen Schreibkurs. Ich danke meiner damaligen Lehrerin Diana Kordas, die mir in diesem Jahr dabei half, Selbstvertrauen und Fertigkeiten zu sammeln. Später schrieb ich für das *Network World Magazine* und entwickelte meine Fertigkeiten als technischer Autor im Bereich Data Center. Ich danke John Dix und John Gallant, die mir meinen ersten Job als Kolumnist bei *Network World* gaben, sowie meinem Lektor Michael Cooney und meinem

Kollegen Johna Till Johnson, die meine Kolumnen lektorierten und für eine Veröffentlichung aufbereiteten. Vier Jahre lang 500 Wörter pro Woche zu schreiben, sorgten für ausreichend Erfahrung, um ernsthaft über ein Dasein als Autor nachzudenken.

Vielen Dank auch an diejenigen, die mich unterstützten, nachdem ich meinen Buchvorschlag bei O'Reilly eingereicht hatte, indem sie Empfehlungen aussprachen und sich den Entwurf genauer ansahen. Mein Dank geht an John Gallant, Gregory Ness, Richard Stiennon, Joel Snyder, Adam B. Levine, Sandra Gittlen, John Dix, Johna Till Johnson, Roger Ver und Jon Matonis. Besonderer Dank geht an Richard Kagan und Tymon Mattoszko, die frühe Fassungen prüften, und an Matthew Taylor, der diese Fassung lektorierte.

Dank an Cricket Liu, Autor des O'Reilly-Titels *DNS and BIND*, der mich bei O'Reilly vorgestellt hat. Ein Dank auch an Michael Loukides und Allyson MacDonald von O'Reilly, die Monate daran arbeiteten, dass dieses Buch Wirklichkeit wurde. Allyson war besonders aufmerksam, wenn Abgabefristen verstrichen und Ergebnisse fehlten. Bei der zweiten Ausgabe gab Timothy McGovern die Richtung vor, Kim Cofer übernahm das Lektorat, und Rebecca Panzer sorgte für viele neue Diagramme.

Die ersten Entwürfe der ersten Kapitel waren die schwersten, schlicht weil Bitcoin ein kompliziertes Thema ist. Sobald ich einen Aspekt herauspickte, musste ich direkt schon wieder das große Ganze betrachten. Wiederholt blieb ich hängen und war frustriert, wenn ich versuchte, ein Thema leicht verständlich rüberzubringen, indem ich eine Geschichte um ein schwieriges technisches Thema herum erzählen wollte. Letztendlich entschied ich mich dafür, die Geschichte des Bitcoins über die Geschichten derjenigen zu erzählen, die Bitcoins nutzen. Das Buch zu schreiben, wurde dadurch erheblich einfacher. Ich schulde meinem Freund und Mentor Richard Kagan Dank, der mir dabei half, die Geschichte zu entwirren und meine Schreibblockaden zu überwinden. Ich danke Pamela Morgan, die frühe Fassungen jedes Kapitels der ersten und zweiten Auflage Korrektur las und die richtigen Fragen stellte. Mein Dank geht auch an die Entwickler der »San Francisco Bitcoin Developers Meetup«-Gruppe sowie an Taariq Lewis und Denise Terry, die dabei halfen, das frühe Material zu testen. Dank ebenfalls an Andrew Naugler für den Entwurf der Infografiken.

Während ich das Buch schrieb, machte ich frühe Fassungen auf GitHub verfügbar und lud dazu ein, diese zu kommentieren. Über 100 Kommentare, Vorschläge, Korrekturen und Beiträge sind daraufhin eingegangen. Für diese Beiträge bedanke ich mich explizit in »Early Release Draft (GitHub-Beiträge)« auf Seite XXI. Zuallererst gilt mein Dank meinen freiwilligen GitHub-Lektoren Ming T. Nguyen (erste Auflage) und Will Binns (zweite Auflage), die auf GitHub unermüdlich Pull-Requests kuratiert, verwaltet und aufgelöst, Reports veröffentlicht und Bug-Fixes vorgenommen haben.

Sobald die erste Fassung stand, wurde sie mehrfach von technischen Korrektoren überarbeitet. Vielen Dank an Cricket Liu und Lorne Lantz für deren sorgfältiges Korrekturlesen sowie ihre Kommentare und die Unterstützung.

Verschiedene Bitcoin-Entwickler steuerten Codebeispiele, Korrekturen und Kommentare bei. Dank an Amir Taaki und Eric Voskuil für Beispielcode und viele gute Kommentare, Chris Kleeschulte für den Bitcore-Anhang, Vitalik Buterin und Richard Kiss für Codebeiträge und ihre Hilfe bei der Mathematik elliptischer Kurven, Gavin Andresen für Korrekturen und Kommentare, Michalis Kargakis für Kommentare und Beiträge sowie Robin Inge für Fehlerkorrekturen der zweiten Auflage. Auch bei der zweiten Auflage erhielt ich wieder Hilfe von vielen Bitcoin-Core-Entwicklern, darunter Eric Lombrozo, der Segregated Witness entmystifizierte, Luke Dashjr, der mir beim Kapitel über Transaktionen half, Johnson Lau, der (unter anderem) das Kapitel zu Segregated Witness Korrektur las, und viele andere. Ich danke Joseph Poon, Tadge Dryja und Olaoluwa Osuntokun, die Lightning Networks erklärten, meinen Text Korrektur lasen und Fragen beantworteten, wenn ich nicht weiterkam.

Meine Liebe für Wörter und Bücher verdanke ich meiner Mutter Theresa, die mich in einem Haus aufzog, in dem Bücher jede Wand mit Beschlag belegten. Meine Mutter kaufte mir 1982 auch meinen ersten Computer, obwohl sie sich selbst als technophob beschrieb. Mein Vater Menelaos, ein Bauingenieur, der sein erstes Buch im Alter von 80 Jahren veröffentlichte, lehrte mich logisches und analytisches Denken und schürte meine Vorliebe für Wissenschaft und Technik.

Ich danke euch allen für eure Unterstützung während meiner Reise.

Early Release Draft (GitHub-Beiträge)

Viele Beitragende lieferten Kommentare, Korrekturen und Ergänzungen zum Early Release Draft auf GitHub. Ich danke euch allen für euren Beitrag zu diesem Buch.

Nachfolgend eine Liste wichtiger GitHub-Beitragender mit deren GitHub-IDs in Klammern:

- Alex Waters (alexwaters)
- Andrew Donald Kennedy (grkvlt)
- bitcoinctf
- Bryan Gmyrek (physicsdude)
- Casey Flynn (cflynn07)
- Chapman Shoop (belovachap)
- Christie D'Anna (avocadobreath)
- Cody Scott (Siecje)
- coinradar
- Cragin Godley (cgodley)
- dallyshalla
- Diego Viola (diegoviola)
- Dirk Jäckel (biafra23)
- Dimitris Tsapakidis (dimitris-t)
- Dmitry Marakasov (AMDmi3)
- drstrangeM
- Ed Eykholt (edeykholt)
- Ed Leafe (EdLeafe)

- Edward Posnak (edposnak)
- Elias Rodrigues (elias19r)
- Eric Voskuil (evoskuil)
- Eric Winchell (winchell)
- Erik Wahlström (erikwam)
- effectsToCause (vericoin)
- Esteban Ordano (eordano)
- ethers
- fabienhinault
- Frank Höger (francyi)
- Gaurav Rana (bitcoinsSG)
- genjix
- halseth
- Holger Schinzel (schinzelh)
- Ioannis Cherouvim (cherouvim)
- Ish Ot Jr. (ishotjr)
- James Addison (jayaddison)
- Jameson Lopp (jlopp)
- Jason Bisterfeldt (jbisterfeldt)
- Javier Rojas (fjrojasgarcia)
- Jeremy Bokobza (bokobza)
- JerJohn15
- Joe Bauers (joebauers)
- joflynn
- Johnson Lau (jl2012)
- Jonathan Cross (jonathancross)
- Jorgeminator
- Kai Bakker (kaibakker)
- Mai-Hsuan Chia (mhchia)
- Marzig (marzig76)
- Maximilian Reichel (phramz)
- Michalis Kargakis (kargakis)
- Michael C. Ippolito (michaelcippolito)
- Mihail Russu (MihailRussu)
- Minh T. Nguyen (enderminh)
- Nagaraj Hubli (nagarajhubli)
- Nekomata (nekomata-3)
- Robert Furse (Rfurse)
- Richard Kiss (richardkiss)
- Ruben Alexander (hizzvizz)
- Sam Ritchie (sritchie)
- Sergej Kotliar (ziggamon)
- Seiichi Uchida (topecongiro)
- Simon de la Rouviere (simondlr)
- Stephan Oeste (Emzy)
- takaya-imai
- Thiago Arrais (thiagoarrais)
- venzen
- Will Binns (wbnns)
- wintercooled
- wjx
- Wojciech Langiewicz (wlk)
- yurigeorgiev4

Glossar

Dieses Glossar umfasst viele der im Zusammenhang mit Bitcoin und Blockchain verwendeten Begriffe. Die Begriffe kommen im gesamten Buch ständig vor.

Adresse
: Eine Bitcoin-Adresse ist ein aus Buchstaben und Ziffern bestehender String wie beispielsweise 1DSrfJdB2AnWaFNgSbv3MZC2m74996JafV. Eigentlich handelt es sich um die base58check-codierte Version eines 160-Bit-Public-Key-Hash. Genau wie Sie jemanden bitten, Ihnen eine E-Mail an Ihre E-Mail-Adresse zu schicken, senden Ihnen andere Bitcoins an Ihre Bitcoin-Adressen.

Belohnung (Reward)
: In jedem neuen Block enthaltene Menge an Bitcoin, die der Miner erhält, der die Lösung für den Proof-of-Work gefunden hat, momentan (März 2018) 12,5 BTC pro Block.

Bestätigungen
: Ist eine Transaktion in einen Block aufgenommen worden, hat sie ihre erste Bestätigung erhalten. Sobald ein weiterer Block geschürft wird, hat die Transaktion zwei Bestätigungen und so weiter. Bei sechs oder mehr Bestätigungen gilt eine Transaktion als unumkehrbar.

BIP
: Bitcoin Improvement Proposals. Eine Reihe von Vorschlägen/Anträgen, die die Mitglieder der Bitcoin-Community eingereicht haben, um den Bitcoin zu verbessern. Zum Beispiel ist BIP-21 ein Vorschlag zur Verbesserung des Bitcoin-URI-Schemas (Uniform Resource Identifier).

Bitcoin
: Der Name einer Währungseinheit (des Coins), des Netzwerks und der Software.

Block
: Eine Gruppierung von Transaktionen, die mit einem Zeitstempel und einem Fingerprint des vorherigen Blocks markiert sind. Der Block-Header wird gehasht, um den Proof-of-Work zu erzeugen und damit die Transaktionen zu

validieren. Gültige Blöcke werden der Haupt-Blockchain durch Netzwerkkonsens hinzugefügt.

Block, veralteter (stale)
: Erfolgreich geschürfter Block, der nicht in die momentan beste Blockchain aufgenommen wurde, weil ein anderer Block der gleichen Höhe die Chain als Erster erweitert hat.

Blockchain
: Eine Liste validierter Blöcke. Jeder Block ist (bis zurück zum Genesis-Block) mit seinem Vorgänger verlinkt.

byzantinischen Generäle, Problem der
: Ein zuverlässiges Computersystem muss mit Fehlern einer oder mehrerer seiner Komponenten umgehen können. Eine fehlerhafte Komponente kann ein häufig übersehenes Verhalten zeigen, nämlich das Senden widersprüchlicher Informationen an unterschiedliche Teile des Systems. Das Problem des Umgangs mit dieser Art Fehler wird abstrakt oft als das Problem der byzantinischen Generäle beschrieben.

Coinbase
: Ein spezielles Feld, das als Input für Coinbase-Transaktionen verwendet wird. Mit der Coinbase kann man seinen Anspruch auf die Blockbelohnung geltend machen und bis zu 100 Byte beliebige Daten eintragen. Nicht zu verwechseln mit der Coinbase-Transaktion.

Coinbase-Transaktion
: Die erste Transaktion eines Blocks. Sie wird immer von einem Miner erzeugt und enthält eine einzelne Coinbase. Nicht zu verwechseln mit Coinbase.

Cold Storage
: Das Offlinespeichern von Bitcoins (oder zumindest eines Teils davon). Cold Storage erreicht man durch die Erzeugung privater Schlüssel und deren Offlinespeicherung in einer sicheren Umgebung. Cold Storage ist für jeden wichtig, der Bitcoins hält. Ist Ihr Computer online, ist er für Hackerangriffe anfällig. Er sollte deshalb nicht zur Speicherung signifikanter Bitcoin-Mengen genutzt werden.

Colored Coins
: Ein Open-Source-Bitcoin-2.0-Protokoll, das es Entwicklern erlaubt, digitale Vermögenswerte (Assets) auf der Bitcoin-Blockchain aufzusetzen und so die Funktionalität des Bitcoins über eine Währung hinaus zu erweitern.

Difficulty
: Eine netzwerkweit gültige Einstellung, die festlegt, wie viel Rechenleistung nötig ist, um den Proof-of-Work zu berechnen.

Difficulty Retargeting
: Eine netzwerkweite Neuberechnung der Difficulty. Erfolgt einmal alle 2.016 Blöcke und berücksichtigt die Hashing-Leistung der vorangegangenen 2.016 Blöcke.

Difficulty Target
: Eine Difficulty, bei der alle Berechnungen im Netzwerk im Schnitt alle zehn Minuten einen Block finden.

Double-Spending
: Double-Spending bedeutet, Geld erfolgreich zweimal ausgeben zu können. Bitcoin schützt sich vor Double-Spending, indem es jede Transaktion verifiziert, die in die Blockchain aufgenommen wird. Dabei wird sichergestellt, dass die Inputs der Transaktion nicht bereits eingelöst wurden.

ECDSA
: Elliptic Curve Digital Signature Algorithm, kurz ECDSA, ist ein kryptografischer Algorithmus, über den Bitcoin sicherstellt, dass die Mittel nur von ihren rechtmäßigen Besitzern ausgegeben werden können.

Extra Nonce
: Als die Difficulty anstieg, sind Miner häufig alle 4 Milliarden möglichen Werte für die Nonce durchgegangen, ohne einen Block gefunden zu haben. Weil das Coinbase-Skript zwischen 2 und 100 Bytes Daten speichern kann, begannen die Miner damit, diesen Platz für eine zusätzliche Nonce zu nutzen und so einen wesentlich größeren Bereich von Block-Header-Werten nach einem gültigen Block abzusuchen.

Fork
: Ein Fork tritt ein, wenn zwei oder mehr Blöcke die gleiche Blockhöhe aufweisen, sodass sich die Blockchain »teilt« (engl. Fork). Das passiert üblicherweise, wenn zwei oder mehr Miner nahezu gleichzeitig einen Block finden. Kann auch als Teil eines Angriffs vorkommen.

Gebühren
: Der Sender einer Transaktion fügt eine Gebühr für die Verarbeitung seiner Transaktion hinzu.

Geheimer Schlüssel (Secret Key oder auch Private Key)
: Die geheime Zahl, die die Bitcoins freigibt, die an die dazugehörige Adresse gesendet wurden. Ein solcher geheimer Schlüssel ist z. B.: 5J76sF8L5jTtzE96r66Sf8cka9y44wdpJjMwCxR3tzLh3ibVPxh.

Genesis-Block
: Der erste Block in der Blockchain, der zur Initialisierung der Kryptowährung verwendet wurde.

Hard Fork
: Ein Hard Fork ist eine permanente Teilung der Blockchain. Tritt häufig ein, wenn veraltete Nodes ihre UTXO-Datenbank nicht mehr aktualisieren können.

Hardware-Wallet
: Eine Hardware-Wallet ist eine spezielle Bitcoin-Wallet, die die privaten Schlüssel des Nutzers auf einer sicheren Hardware speichert.

Hash
: Der digitale Fingerabdruck einer binären Eingabe.

Hashlock
: Ein Hashlock ist eine Art Schuld, die das Einlösen eines Outputs erst erlaubt, wenn bestimmte Daten öffentlich werden. Hashlocks haben die nützliche Eigenschaft, dass alle anderen Hashlocks, die über den gleichen Schlüssel gesichert sind, ebenfalls geöffnet werden können, sobald dieser Hashlock bekannt ist. Auf diese Weise lassen sich mehrere Outputs erzeugen, die durch den gleichen Hashlock geschützt sind und zur gleichen Zeit eingelöst werden können.

HD-Protokoll
: Das Hierarchical Deterministic (HD) Key Creation and Transfer Protocol (BIP-32). Erlaubt die hierarchische Erzeugung von Child-Schlüsseln aus einem einzigen Parent-Schlüssel.

HD-Wallet
: Wallets, die das HD-Protokoll (BIP-32) verwenden.

HD-Wallet-Seed
: HD-Wallet-Seed oder Root-Seed ist ein (potenziell kurzer) Wert, der als Seed-Wert zur Generierung des privaten Master-Keys und Master-Chain-Codes verwendet wird.

HTLC
: Ein Hashed Time Lock Contract, kurz HTLC, ist eine Klasse von Hashlocks und Timelocks nutzenden Zahlungen, bei der der Empfänger einer Zahlung diese vor einer festgelegten Frist durch eine kryptografische Zahlungsbestätigung bescheinigt. Anderenfalls kann die Zahlung nicht mehr an ihn überwiesen werden, und der Betrag geht an den Zahlungspflichtigen zurück.

Konsens
: Konsens besteht, wenn mehrere Nodes (üblicherweise ein Großteil der Nodes im Netzwerk) die gleichen Blöcke in der lokal validierten »besten« Blockchain haben. Nicht zu verwechseln mit den Konsensregeln.

Konsensregeln
: Die Regeln für die Blockvalidierung, denen alle Full Nodes folgen, um den Konsens mit anderen Nodes zu erhalten. Nicht zu verwechseln mit Konsens.

KYC
: Know Your Customer (KYC), zu Deutsch etwa »kenne deinen Kunden«, ist der Prozess, mit dem ein Unternehmen die Identität seiner Kunden verifiziert. Der Begriff beschreibt auch die gesetzlichen Vorgaben, die diesen Prozess regeln.

LevelDB
: LevelDB ist ein als Bibliothek ausgelegter festplattenorientierter Schlüssel/Wert-Speicher. Die Bibliothek ist Open Source und für viele Plattformen verfügbar.

Lightning Network
: Lightning Network ist eine vorgeschlagene Implementierung von HTLCs (Hashed Time Lock Contracts) mit bidirektionalen Zahlungskanälen, bei denen Zahlungen über mehrere Peer-to-Peer-Zahlungskanäle sicher geroutet werden können. Das erlaubt den Aufbau eines Netzwerks, bei dem jeder Peer im Netzwerk jeden anderen Peer bezahlen kann, selbst wenn es keinen direkten Kanal zwischen den beiden gibt.

Locktime
: Locktime, oder etwas technischer nLockTime, ist der Teil einer Transaktion, der festlegt, zu welcher Zeit oder zu welchem Block eine Transaktion frühestens in die Blockchain eingefügt werden kann.

Mempool
: Der Bitcoin-Mempool ist die Sammlung aller Transaktionsdaten eines Blocks, die von den Bitcoin-Nodes verifiziert, aber noch nicht bestätigt wurden.

Merkle Root
: Die »Wurzel« eines Merkle Tree, d. h. der Stamm aller gehashten Paare in einem Baum. Block-Header müssen eine gültige Merkle Root für alle Transaktionen in einem Block enthalten.

Merkle Tree
: Ein Baum, der durch das Hashing gepaarter Daten (der »Blätter«) erzeugt wird. Diese Paare werden dann immer weiter gehasht, bis nur noch ein einziger Hash übrig bleibt: die Merkle Root. Beim Bitcoin sind die Blätter fast immer Transaktionen eines einzelnen Blocks.

Miner
: Eine Netzwerk-Node, die durch wiederholtes Hashing einen gültigen Proof-of-Work für neue Blöcke findet.

Multisignatur
: Die Multisignatur (Multisig) verlangt mehr als einen Schlüssel für die Autorisierung einer Bitcoin-Transaktion.

Netzwerk
: Ein Peer-to-Peer-Netzwerk, das Transaktionen und Blöcke an jede Bitcoin-Node im Netzwerk propagiert.

Nonce
: Die Nonce in einem Bitcoin-Block ist ein 32-Bit-Feld (4 Byte) und wird so gesetzt, dass der Hash des Blocks eine bestimmte Anzahl führender Nullen enthält. Die restlichen Felder dürfen nicht verändert werden, da sie eine definierte Bedeutung haben.

Off-Chain-Transaktionen
: Eine Off-Chain-Transaktion bewegt Werte außerhalb der Blockchain. Während eine On-Chain-Transaktion – die üblicherweise einfach als »Transaktion« bezeichnet wird – die Blockchain modifiziert und darauf angewiesen ist,

dass die Blockchain deren Gültigkeit validiert, verwendet eine Off-Chain-Transaktion andere Methoden, um eine Transaktion festzuhalten und zu validieren.

Opcode
: Operationscodes (kurz Opcodes) der Bitcoin-eigenen Skriptsprache (Script), die in Pubkey- oder Signaturskripten Daten ablegen oder bestimmte Funktionen durchführen.

Open-Assets-Protokoll
: Das Open-Assets-Protokoll ist ein einfaches, aber leistungsfähiges Protokoll, das auf der Bitcoin-Blockchain aufsetzt. Es erlaubt die Emission und den Transfer selbst definierter »Assets« (Vermögenswerte). Das Open-Assets-Protokoll ist eine Weiterentwicklung des Colored-Coins-Konzepts.

OP_RETURN
: Ein Opcode, der in einem der Outputs einer OP_RETURN-Transaktion verwendet wird. Nicht zu verwechseln mit einer OP_RETURN-Transaktion.

OP_RETURN-Transaktion
: Ein Transaktionstyp, der standardmäßig seit Bitcoin Core 0.9.0 (und höher) weitergeleitet und geschürft wird. Fügt beliebige Daten zu einem nachweislich nicht einlösbaren Pubkey-Skript hinzu, das von Full Nodes nicht in deren UTXO-Datenbank gespeichert werden muss. Nicht zu verwechseln mit dem OP_RETURN-Opcode.

Output
: Output, Transaktions-Output oder TxOut ist das Ergebnis einer Transaktion und besteht aus zwei Feldern: einem Wertfeld zur Übertragung von null oder mehr Satoshis und einem Pubkey-Skript, das die Bedingungen festlegt, zu denen diese Satoshis eingelöst werden können.

P2PKH
: Transaktionen an eine Bitcoin-Adresse enthalten ein P2PKH- oder Pay-to-Public-Key-Hash-Skript. Ein Output, der an ein P2PKH-Skript gekoppelt ist, kann freigegeben (eingelöst) werden, indem man einen Public Key und eine mit dem dazugehörigen privaten Schlüssel erzeugte digitale Signatur vorlegt.

P2SH
: P2SH, oder Pay To Script Hash, ist eine mächtige neue Art von Transaktion, die den Einsatz komplexer Transaktionsskripte stark vereinfacht. Bei P2SH ist das komplexe Skript, das die Bedingungen für die Freigabe des Outputs (Redeem-Skript) festlegt, nicht im Locking-Skript enthalten. Stattdessen enthält das Locking-Skript nur einen Hash auf das Redeem-Skript.

P2SH-Adresse
: P2SH-Adressen sind Base58Check-codierte 20-Byte-Hashes eines Skripts. P2SH-Adressen verwenden das Versionspräfix 5, was Base58Check-codierte Adressen ergibt, die mit einer 3 beginnen. P2SH-Adressen verstecken die

gesamte Komplexität, d. h., die eine Zahlung vornehmende Person bekommt das Skript nicht zu sehen.

P2WPKH
: Die Signatur eines P2WPKH (Pay to Witness Public Key Hash) enthält die gleichen Informationen wie ein P2PKH, steht aber im Witness- und nicht im scriptSig-Feld. Der scriptPubKey wird ebenfalls modifiziert.

P2WSH
: Der Unterschied zwischen P2SH und P2WSH (Pay to Witness Script Hash) besteht darin, dass der kryptografische Beweis aus dem scriptSig- in das Witness-Feld verschoben wird. Der scriptPubKey wird ebenfalls modifiziert.

Paper-Wallet
: Im engeren Sinne ist eine Paper-Wallet ein Dokument, das alle Daten enthält, die notwendig sind, um eine beliebige Anzahl privater Bitcoin-Schlüssel zu erzeugen. Für die meisten Menschen ist sie eine Möglichkeit, Bitcoins (samt Paper Keys und Freigabecodes) offline in einem physischen Dokument zu speichern.

Pool-Mining
: Beim Pool-Mining tragen mehrere Clients zur Generierung eines neuen Blocks bei und teilen sich die Erlöse entsprechend der beigetragenen Leistung.

Proof-of-Stake
: Proof-of-Stake (PoS) ist eine Methode, nach der das Blockchain-Netzwerk einer Kryptowährung versucht, einen netzwerkweiten Konsens zu erzielen. Beim Proof-of-Stake müssen die Nutzer den Besitz einer bestimmten Menge der Währung (ihren »Anteil«, engl. Stake) nachweisen.

Proof-of-Work
: Ein Stück Daten, dessen Auffinden einen beträchtlichen rechnerischen Aufwand verlangt. Beim Bitcoin müssen Miner eine numerische Lösung für den SHA256-Algorithmus finden, die eine netzwerkweite Zielvorgabe, das sogenannte Difficulty Target, erfüllt.

RIPEMD-160
: RIPEMD-160 ist eine kryptografische 160-Bit-Hashfunktion. RIPEMD-160 ist eine erweiterte Version von RIPEMD mit einem Hashergebnis von 160 Bit. Man erwartet, dass sie für die nächsten zehn Jahre (oder mehr) sicher ist.

Satoshi Nakamoto
: Satoshi Nakamoto ist der Name, der von jener Person (oder den Personen) verwendet wurde, die den Bitcoin entworfen und die ursprüngliche Referenzimplementierung entwickelt hat. Als Teil der Implementierung hat er auch die erste Blockchain-Datenbank entwickelt, wobei er als Erster das Double-Spending-Problem für Digitalwährungen gelöst hat. Seine wahre Identität ist bis heute nicht bekannt.

Script
: Bitcoin verwendet ein Skripting-System für Transaktionen. Script orientiert sich an Forth und ist eine einfache, stackbasierte Sprache, die von links nach rechts verarbeitet wird. Sie verzichtet bewusst auf Schleifen und ist daher nicht Touring-vollständig.

ScriptPubKey (alias Pubkey Script)
: ScriptPubKey, oder Pubkey Script, ist ein in Outputs enthaltenes Skript, das die Bedingungen festlegt, die erfüllt werden müssen, um die Satoshis einlösen zu können. Die Daten, die zur Erfüllung dieser Bedingungen benötigt werden, können in einem Signaturskript bereitgestellt werden.

ScriptSig (aka Signaturskript)
: ScriptSig, oder Signaturskript, steht für die Daten, die der Einlösende generiert. Diese Daten werden fast immer als Variablen verwendet, um die Bedingungen eines Pubkey-Skripts zu erfüllen.

Segregated Witness
: Segregated Witness ist eine Erweiterung des Bitcoin-Protokolls. Sie trennt die Signaturdaten von den Bitcoin-Transaktionen ab. Segregated Witness ist ein Soft Fork, der die Regeln des Bitcoin-Protokolls technisch restriktiver handhabt.

SHA
: Der Secure-Hash-Algorithmus, kurz SHA, ist eine Familie kryptografischer Hashfunktionen, die vom National Institute of Standards and Technology (NIST) veröffentlicht wurde.

Soft Fork
: Ein Soft Fork ist eine temporäre Teilung der Blockchain. Tritt auf, wenn Miner nicht aktualisierte Nodes nutzen, die einer neuen Konsensregel nicht folgen. Nicht zu verwechseln mit Fork, Hard Fork, Software Fork oder Git Fork.

SPV (Simplified Payment Verification)
: SPV, oder Simplified Payment Verification, ist eine Methode, um zu verifizieren, ob bestimmte Transaktionen in einem Block enthalten sind, ohne den gesamten Block herunterladen zu müssen. Diese Methode wird von einigen leichtgewichtigen Bitcoin-Clients genutzt.

Timelocks
: Ein Timelock verhindert, dass Bitcoins vor einem bestimmten Zeitpunkt (oder einer bestimmten Blockhöhe) eingelöst werden können. Timelocks spielen bei vielen Bitcoin-Verträgen eine wichtige Rolle, einschließlich Zahlungskanälen und gehashten Timelock-Verträgen.

Transaktion
: Einfach ausgedrückt die Übertragung von Bitcoin von einer Adresse an eine andere. Genauer formuliert, ist eine Transaktion eine signierte Datenstruktur,

die den Transfer von Werten ausdrückt. Transaktionen werden über das Bitcoin-Netzwerk übertragen, von Minern gesammelt und in Blöcken eingetragen, um permanent in der Blockchain festgehalten zu werden.

Transaktionspool
: Ungeordnete Liste von Transaktionen, die nicht in Blöcken der Haupt-Chain enthalten sind, für die es aber Input-Transaktionen gibt.

Turing-Vollständigkeit
: Eine Programmiersprache ist »Turing-vollständig«, wenn sie bei ausreichend Zeit und Speicherplatz jedes Programm ausführen kann, das auch eine Turing-Maschine ausführen kann.

UTXO (Unspent Transaction Output)
: UTXO ist ein nicht eingelöster Transaktions-Output, der in einer neuen Transaktion als Input verwendet werden kann.

Verwaister Block
: Ein Block, dessen Parent-Block von der lokalen Node noch nicht verarbeitet ist und der daher noch nicht validiert werden konnte.

Verwaiste Transaktionen
: Transaktionen, die nicht in den Pool aufgenommen werden können, weil ein oder mehrere Input-Transaktionen fehlen.

Wallet
: Software, die ihre Bitcoin-Adressen und geheimen Schlüssel vorhält. Fungiert als elektronische Geldbörse, über die Sie Bitcoin senden, empfangen und speichern können.

WIF (Wallet Import Format)
: WIF, oder Wallet Import Format, ist ein Datenaustauschformat, das den Export und Import privater Schlüssel erlaubt. Ein Flag zeigt an, ob ein komprimierter öffentlicher Schlüssel verwendet wird oder nicht.

Zahlungskanäle
: Ein Mikrozahlungs- oder Zahlungskanal steht für eine Klasse von Techniken, die es den Nutzern ermöglichen, eine Vielzahl von Bitcoin-Transaktionen auszuführen, ohne alle Transaktionen in der Bitcoin-Blockchain festhalten zu müssen. Bei einem typischen Zahlungskanal werden nur zwei Transaktionen in die Blockchain eingetragen, während dazwischen eine (nahezu) unbeschränkte Anzahl von Zahlungen zwischen den Parteien durchgeführt werden kann.

KAPITEL 1

Einführung

Was ist Bitcoin?

Bitcoin ist eine Sammlung von Konzepten und Technologien, die ein Ökosystem für digitales Geld bilden. Währungseinheiten namens Bitcoin werden genutzt, um Werte zu speichern und sie zwischen den Teilnehmern des Bitcoin-Netzwerks zu übertragen. Bitcoin-Nutzer kommunizieren miteinander über das Bitcoin-Protokoll. Das geschieht hauptsächlich über das Internet, andere Transportprotokolle sind aber auch möglich. Der Bitcoin-Protokollstack steht als Open-Source-Software zur Verfügung und ist auf einer Vielzahl von Geräten (einschließlich Laptops und Smartphones) lauffähig, d. h., der Zugang zu dieser Technik gestaltet sich einfach.

Nutzer können Bitcoin über das Netzwerk transferieren und damit das tun, was man auch mit normalem Geld macht: Güter kaufen und verkaufen, Geld an Menschen oder Organisationen überweisen oder jemandem einen Kredit gewähren. Bitcoin kann an speziellen Börsen gekauft, verkauft und gegen andere Währungen getauscht werden. Bitcoin ist in gewissem Sinn das perfekte Geld für das Internet, da es schnell und sicher ist und keine Grenzen kennt.

Im Gegensatz zu traditionellen Währungen ist Bitcoin vollständig virtuell. Es gibt keine Münzen im herkömmlichen Sinn und auch keine digitalen Münzen. Die Münzen (also die Coins) sind in Transaktionen enthalten, die Werte vom Sender zum Empfänger transferieren. Bitcoin-Nutzer verfügen über Schlüssel, die den Besitz von Bitcoins im Bitcoin-Netzwerk nachweisen. Mit diesen Schlüsseln können sie Transaktionen signieren, um den Betrag freizugeben und an einen neuen Eigentümer zu transferieren. Die Schlüssel werden häufig in einer digitalen Geldbörse (der *Wallet*) auf dem Computer oder Smartphone des Nutzers gespeichert. Der Besitz des Schlüssels, mit dem eine Transaktion signiert werden kann, ist die einzige Voraussetzung, um Bitcoins auszugeben, d. h., die Kontrolle liegt vollständig in den Händen der Nutzer.

Bitcoin ist ein verteiltes Peer-to-Peer-System. Daher gibt es keinen »zentralen« Server oder Kontrollpunkt. Bitcoins werden in einem als *Mining* bezeichneten Prozess erzeugt, bei dem darum gerungen wird, wer als Erster die Lösung eines mathema-

tischen Problems findet, während die Bitcoin-Transaktionen verarbeitet werden. Jeder Teilnehmer am Bitcoin-Netzwerk (d. h. jeder, auf dessen Gerät der vollständige Bitcoin-Protokollstack läuft) kann als *Miner* fungieren und die Rechenleistung seines Computers nutzen, um Transaktionen zu verifizieren und festzuhalten. Im Schnitt ist alle zehn Minuten jemand in der Lage, die Transaktionen der letzten zehn Minuten zu verifizieren, und wird dafür mit neuen Bitcoins belohnt. Im Grunde dezentralisiert das Mining die Geldausgabe und die Abrechnung (das *Clearing*), wodurch eine Zentralbank überflüssig wird.

Das Bitcoin-Protokoll enthält fest eingebaute Algorithmen, die die Mining-Funktion innerhalb des Netzwerks regeln. Der Schwierigkeitsgrad (die *Difficulty*) der Rechenaufgabe, die die Miner lösen müssen, wird dynamisch so angepasst, dass im Durchschnitt alle zehn Minuten jemand erfolgreich ist, und zwar unabhängig davon, wie viele Miner (und wie viel Rechenleistung) gerade an der Lösung arbeiten. Das Protokoll halbiert alle vier Jahre die Geschwindigkeit, mit der neue Bitcoins erzeugt werden, und beschränkt die Gesamtzahl der Bitcoins auf etwas unter 21 Millionen. Das führt dazu, dass die im Umlauf befindlichen Bitcoins einer einfach vorhersagbaren Kurve folgen, nach der die 21 Millionen im Jahr 2140 erreicht werden. Durch die sinkende Geschwindigkeit der Ausgabe ist die Währung Bitcoin auf lange Sicht deflationär. Darüber hinaus kann der Bitcoin nicht »aufgeblasen« werden, indem man neue Coins über oder unter der erwarteten Ausgaberate »druckt«.

Hinter den Kulissen ist Bitcoin auch der Name eines Protokolls, eines Peer-to-Peer-Netzwerks und einer Innovation in Sachen Distributed Computing. Tatsächlich ist die Währung Bitcoin nur die erste Anwendung dieser innovativen Technik. Bitcoin repräsentiert den Höhepunkt jahrzehntelanger Forschung zu den Themen Kryptografie und verteilte Systeme. Sie fasst vier Schlüsselinnovationen in einer einmaligen und leistungsfähigen Kombination zusammen. Bitcoin besteht aus:

- einem dezentralisierten Peer-to-Peer-Netzwerk (dem Bitcoin-Protokoll),
- einem öffentlichen Kassenbuch (der Blockchain),
- einer Reihe von Regeln für die unabhängige Validierung von Transaktionen und die Geldausgabe (Konsensregeln) sowie
- einem Mechanismus, mit dem ein globaler, dezentralisierter Konsens zur jeweils gültigen Blockchain erreicht wird (Proof-of-Work-Algorithmus).

Als Entwickler sehe ich Bitcoin als eine Art Internet des Geldes, als Netzwerk für die Verteilung von Werten und die Sicherung des Eigentums an digitalen Vermögenswerten mithilfe verteilter Berechnungen. Hinter Bitcoin steht viel mehr, als es auf den ersten Blick scheint.

In diesem Kapitel wollen wir einige der wesentlichen Konzepte und Begriffe erläutern, uns die notwendige Software beschaffen und Bitcoin für einfache Transaktionen nutzen. In den folgenden Kapiteln sehen wir uns dann schrittweise die tieferen Schichten der Technik an, die Bitcoin möglich machen, und untersuchen das Innenleben des Bitcoin-Netzwerks und -Protokolls.

Digitale Währungen vor Bitcoin

Das Aufkommen brauchbarer digitaler Währung ist eng mit den Entwicklungen in der Kryptografie verknüpft. Das ist nicht weiter überraschend, wenn man die Herausforderungen betrachtet, vor denen man steht, wenn man Bits nutzt, um Werte zu repräsentieren, die gegen Güter und Dienste getauscht werden können. Für jeden, der digitales Geld akzeptiert, stellen sich drei grundlegende Fragen:

1. Kann ich sicher sein, dass das Geld echt und nicht gefälscht ist?
2. Kann ich sicher sein, dass digitales Geld nur einmal ausgegeben werden kann (das sogenannte »Double-Spending-Problem«)?
3. Kann ich sicher sein, dass niemand außer mir dieses Geld für sich beansprucht?

Die Herausgeber von Papiergeld bekämpfen das Fälschungsproblem mit immer ausgefeilteren Papieren und anspruchsvoller Drucktechnik. Physikalisches Geld verhindert das Problem des doppelten Ausgebens ganz einfach, weil eine Banknote nicht an zwei Orten gleichzeitig sein kann. Natürlich wird konventionelles Geld häufig digital gespeichert und überwiesen. In diesen Fällen werden Fälschungen und Double-Spending verhindert, indem alle elektronischen Transaktionen durch zentrale Instanzen verarbeitet werden, die eine globale Übersicht über alle im Umlauf befindlichen Währungen haben. Bei digitalem Geld, das nicht auf esoterische Tinten oder Hologramme zurückgreifen kann, bildet Kryptografie die Basis für das Vertrauen in die Legitimität eines Besitzanspruchs. Insbesondere kryptografische digitale Signaturen ermöglichen es einem Nutzer, ein digitales Gut oder eine Transaktion zu signieren und so das Eigentum an diesem Gut zu beweisen. Mit der richtigen Architektur können digitale Signaturen auch verwendet werden, um das Double-Spending-Problem in den Griff zu bekommen.

Als die Kryptografie in den späten 1980ern einer breiteren Masse zur Verfügung stand und besser verstanden wurde, versuchten viele Forscher, Kryptografie zum Aufbau digitaler Währungen zu nutzen. Diese frühen Projekte gaben digitales Geld heraus, das durch eine nationale Währung oder ein Edelmetall wie Gold gedeckt war.

Zwar funktionierten diese frühen digitalen Währungen, doch sie waren zentralisiert und dementsprechend von Regierungen und Hackern einfach anzugreifen. Frühe digitale Währungen nutzten (genau wie das traditionelle Bankensystem) eine zentrale Abrechnungsstelle, um alle Transaktionen in regelmäßigen Intervallen abzuwickeln. Leider gerieten die meisten dieser aufstrebenden digitalen Währungen ins Visier besorgter Regierungen und wurden letztendlich auf dem Rechtsweg aus dem Weg geschafft. Einige gingen spektakulär unter, als das Mutterunternehmen unvermittelt abgewickelt wurde. Um gegen Interventionen durch Antagonisten gewappnet zu sein, war eine *dezentralisierte* digitale Währung nötig, um einen zentralen Angriffspunkt zu vermeiden. Bitcoin ist ein solches System, es wurde bereits von Grund auf dezentralisiert entworfen. Es kommt vollständig ohne zentrale Autorität oder einer zentralen Kontrollstelle aus, die angegriffen oder geschädigt werden könnte.

Geschichte des Bitcoins

Bitcoin wurde 2008 in einem Papier mit dem Titel »Bitcoin: A Peer-to-Peer Electronic Cash System«[1] vorgestellt, das unter dem Pseudonym Satoshi Nakamoto (siehe Anhang A) veröffentlicht worden war. Nakamoto kombinierte verschiedene frühere Erfindungen wie b-money und Hashcash, um ein vollständig dezentralisiertes Electronic-Cash-System zu entwickeln, das völlig unabhängig war von einer zentralen Instanz für Geldausgabe und Abrechnung sowie die Validierung von Transaktionen. Die Kerninnovation war die Nutzung eines verteilten Rechensystems (das als *Proof-of-Work-Algorithmus* bezeichnet wird), um alle zehn Minuten eine globale »Wahl« durchzuführen, die dem dezentralisierten Netzwerk zu einem *Konsens* über den Zustand der Transaktionen verhilft. Das löst auf elegante Weise das Double-Spending-Problem, bei dem eine einzelne Währungseinheit zweimal ausgegeben werden kann. Bis dahin war das Double-Spending-Problem eine Schwäche digitaler Währungen, die dadurch gelöst wurde, dass alle Transaktionen über eine zentrale Abrechnungsstelle verarbeitet wurden.

Das Bitcoin-Netzwerk startete 2009 basierend auf einer Referenzimplementierung von Nakamoto, die seitdem von vielen anderen Programmierern überarbeitet wurde. Die Leistung der Proof-of-Work-Implementierung (Mining), die für die Sicherheit und Belastbarkeit des Bitcoins sorgt, ist exponentiell angestiegen und übertrifft mittlerweile die kombinierte Rechenleistung der Top-Supercomputer auf der Welt. Die Marktkapitalisierung des Bitcoins kratzt (je nach aktuellem Bitcoin-Dollar-Wechselkurs) an der 100-Milliarden-Dollar-Marke (Stand Oktober 2017). Die bisher größte durch das Netzwerk verarbeitete Transaktion war 150 Millionen Dollar schwer, wurde sofort übertragen und ohne Gebühren verarbeitet.

Satoshi Nakamoto zog sich im April 2011 zurück und übergab die Verantwortung für die Entwicklung des Codes und des Netzwerks an eine Gruppe von Freiwilligen. Die Identität der Person oder Personen hinter Bitcoin ist bisher nicht bekannt. Ungeachtet dessen kontrolliert weder Satoshi Nakamoto noch irgendwer sonst das Bitcoin-System. Es arbeitet auf vollständig transparenten mathematischen Prinzipien, Open-Source-Code und dem Konsens zwischen den Teilnehmern. Diese Erfindung ist für sich genommen schon bahnbrechend und hat bereits zu neuen Forschungen in den Bereichen Distributed Computing, Wirtschaftswissenschaften und Ökonometrie geführt.

1 »Bitcoin: A Peer-to-Peer Electronic Cash System,« Satoshi Nakamoto (*https://Bitcoin.org/Bitcoin.pdf*).

Eine Lösung für ein Distributed-Computing-Problem

Satoshi Nakamotos Erfindung liefert auch eine praktische und neue Lösung für ein Problem des Distributed Computing, das als »Problem der byzantinischen Generäle« bekannt ist. Kurz gefasst, besteht das Problem darin, sich über das Vorgehen oder den Zustand eines Systems zu einigen, in dem Informationen über ein unzuverlässiges und möglicherweise kompromittiertes Netzwerk ausgetauscht werden. Satoshi Nakamotos Lösung, die das Proof-of-Work-Konzept nutzt, um einen Konsens *ohne eine zentrale vertrauenswürdige Instanz* zu erzielen, stellt einen Bruch für das Distributed Computing dar und genießt unabhängig von Währungen eine breite Akzeptanz. Man kann auf diese Weise einen Konsens in dezentralisierten Netzwerken erreichen und so die Rechtmäßigkeit von Wahlen, Lotterien, Anlageregistern, digitalen notariellen Bestätigungen und vielem mehr bestätigen.

Bitcoin: Anwendungsfälle, Anwender und deren Geschichten

Bitcoin ist für die uralte Technik des Geldes eine Innovation. Im Kern erleichtert Geld den Tausch von (wie auch immer gearteten) Werten zwischen Menschen. Um Bitcoin und seine Anwendungsfälle vollständig verstehen zu können, wollen wir ihn daher aus der Perspektive der Menschen betrachten, die ihn nutzen. Jeder dieser Menschen sowie jede der hier für sie aufgeführten Geschichten illustriert einen oder mehrere spezifische Anwendungsfälle. Wir werden ihnen im Verlauf des Buchs immer wieder begegnen:

Endverbraucher
: Alice lebt in der Bay Area im nördlichen Kalifornien. Sie hat über ihre Techie-Freunde von Bitcoin gehört und möchte ihn nutzen. Wir begleiten sie dabei, wenn sie etwas über Bitcoins lernt, sich welche beschafft und dann einen Teil ihrer Bitcoins ausgibt, um sich einen Kaffee in Bobs Café in Palo Alto zu kaufen. Diese Geschichte führt uns aus der Perspektive eines Endverbrauchers in die Software, die Börsen und die grundlegenden Transaktionen ein.

Einzelhändler mit hochpreisigen Produkten
: Carol ist Besitzerin einer Kunstgalerie in San Francisco. Sie nutzt Bitcoins zum Verkauf teurer Bilder. Diese Geschichte zeigt uns die Risiken eines »51-%-Konsensangriffs« für einen Einzelhändler mit hochpreisigen Produkten auf.

Offshore-Verträge
: Bob, der Besitzer des Cafés in Palo Alto, baut eine neue Website auf. Er hat einen Vertrag mit einem indischen Webentwickler, Gopesh, geschlossen, der in Bangalore in Indien lebt. Gopesh hat einer Zahlung in Bitcoin zugestimmt.

Diese Geschichte untersucht die Nutzung des Bitcoins für das Outsourcing, für Dienstleistungsverträge und internationale Überweisungen.

Webshop
: Gabriel ist ein geschäftstüchtiger junger Teenager aus Rio de Janeiro, der in einem kleinen Webshop mit dem Bitcoin-Logo versehene T-Shirts, Becher und Aufkleber verkauft. Gabriel ist zu jung für ein eigenes Bankkonto, doch seine Eltern unterstützen seinen Unternehmergeist.

Gemeinnützige Spenden
: Eugenia ist Direktorin eines Kinderhilfswerks auf den Philippinen. Jüngst hat sie Bitcoin für sich entdeckt und möchte ihn nutzen, um eine ganz neue Gruppe fremder und einheimischer Spender für ihr Hilfswerk zu erreichen. Zudem untersucht sie Möglichkeiten, den Bitcoin dazu zu verwenden, Gelder schnell in bedürftige Gebiete zu schicken. Diese Geschichte zeigt, wie man den Bitcoin für globale Spendensammlungen über Währungs- und Ländergrenzen hinweg nutzt. Sie zeigt außerdem, wie ein offenes Kassenbuch für Transparenz bei Wohltätigkeitsorganisationen sorgen kann.

Import/Export
: Mohammed importiert Elektronik in Dubai. Er versucht, den Bitcoin zu nutzen, um Elektronik aus den USA und China in die Vereinigten Arabischen Emirate zu importieren. Er möchte auf diese Weise die Zahlungen für die Importe beschleunigen. Diese Geschichte zeigt, wie Bitcoins für große internationale B2B-Zahlungen genutzt werden können, die an physikalische Güter gebunden sind.

Bitcoin-Mining
: Jing studiert Informatik in Schanghai. Er hat seine Informatikkenntnisse genutzt und eine *Mining-Rig* gebaut, um sein Einkommen aufzubessern. Diese Geschichte untersucht die »industrielle« Basis des Bitcoins: die spezialisierte Ausrüstung, die verwendet wird, um das Bitcoin-Netzwerk abzusichern und neue Bitcoins zu erzeugen.

Jede dieser Geschichten basiert auf realen Menschen und realen Industrien, die momentan mithilfe des Bitcoins neue Märkte, neue Industrien und innovative Lösungen für globale ökonomische Fragen entwickeln.

Erste Schritte

Bitcoin ist ein Protokoll, auf das man über eine Clientanwendung zugreifen kann, die dieses Protokoll versteht. Eine *Bitcoin-Wallet* ist so etwas wie eine elektronische Geldbörse und die übliche Benutzerschnittstelle zum Bitcoin-System (so wie der Webbrowser die übliche Schnittstelle zum HTTP-Protokoll ist). Es gibt viele verschiedene Implementierungen von Bitcoin-Wallets, ebenso wie es die unterschiedlichsten Webbrowser gibt (z. B. Chrome, Safari, Firefox und Internet Explorer). Und genau wie wir unsere Lieblingsbrowser (Mozilla Firefox, Yay!) und die Schur-

ken darunter (Internet Explorer) haben, variieren auch Bitcoin-Wallets in Qualität, Performance, Sicherheit, Privatsphäre und Zuverlässigkeit. Es gibt ebenfalls eine Referenzimplementierung des Bitcoin-Protokolls, die auch eine Wallet umfasst. Diese ist als »Satoshi-Client« oder »Bitcoin Core« bekannt und leitet sich aus der ursprünglich von Satoshi Nakamoto geschriebenen Implementierung ab.

Wahl einer Bitcoin-Wallet

Bitcoin-Wallets sind mit die am aktivsten entwickelten Anwendungen des Bitcoin-Ökosystems. Es herrscht ein starker Wettbewerb, und während wahrscheinlich gerade jetzt eine neue Wallet entwickelt wird, werden verschiedene Wallets aus dem letzten Jahr nicht mehr aktiv gepflegt. Viele Wallets konzentrieren sich auf bestimmte Plattformen oder auf spezielle Anwendungen. Einige eignen sich besser für Einsteiger, während andere vollgepackt sind mit Features für fortgeschrittene Anwender. Die Wahl einer Wallet ist eine hochgradig subjektive Angelegenheit und hängt von der Nutzung und dem Wissen des Anwenders ab. Es ist daher unmöglich, ein bestimmtes Produkt oder Projekt zu empfehlen. Ungeachtet dessen können wir Bitcoin-Wallets entsprechend ihrer Plattform und ihrer Funktion kategorisieren und etwas Klarheit in Bezug auf die verschiedenen Arten von Wallets schaffen. Darüber hinaus ist der Transfer von Geld zwischen Bitcoin-Wallets einfach, günstig und schnell. Es lohnt sich also, verschiedene Wallets auszuprobieren, bis man die gefunden hat, die den eigenen Bedürfnissen am besten entspricht.

Bitcoin-Wallets lassen sich entsprechend ihrer Plattform wie folgt klassifizieren:

Desktop-Wallet
: Die Desktop-Wallet war die erste Form der Wallet, die als Referenzimplementierung entwickelt wurde. Viele Nutzer verwenden Desktop-Wallets aufgrund ihrer Features, der Autonomie und der von ihnen gebotenen Kontrolle. Der Betrieb auf verbreiteten Betriebssystemen wie Windows und Mac OS ist aber nicht ganz so sicher, weil diese Plattformen selbst häufig unsicher und schlecht konfiguriert sind.

Mobile Wallet
: Mobile Wallets sind die am weitesten verbreitete Form der Bitcoin-Wallets. Sie laufen auf Smartphone-Betriebssystemen wie Apple iOS und Android und sind daher eine gute Wahl für neue Nutzer. Viele stellen Einfachheit und eine unkomplizierte Anwendung in den Vordergrund, doch es gibt auch voll ausgestattete mobile Wallets für Poweruser.

Web-Wallet
: Der Zugriff auf Web-Wallets erfolgt über den Webbrowser, und die Benutzer-Wallets liegen auf den Servern einer dritten Partei. Dies ähnelt Webmail, da man vollständig von Servern eines Drittanbieters abhängig ist. Einige dieser Dienste arbeiten mit clientseitigem Code, der auf dem Browser des Benutzers ausgeführt wird, wodurch der Nutzer die Kontrolle über die Schlüssel behält.

Die meisten stellen allerdings einen Kompromiss dar, bei dem die Kontrolle über die Schlüssel der Nutzer übernommen wird, um eine einfache Nutzung zu ermöglichen. Es ist nicht empfehlenswert, größere Mengen an Bitcoin auf Systemen von Drittanbietern zu speichern.

Hardware-Wallet
: Hardware-Wallets sind Geräte, die eine sichere eigenständige Bitcoin-Wallet auf spezieller Hardware betreiben. Sie werden per USB über einen Webbrowser am Desktop gesteuert oder per *Near Field Communication* (NFC) von einem mobilen Gerät. Da alle Bitcoin-bezogenen Operationen auf dieser speziellen Hardware abgewickelt werden, betrachtet man diese Wallets als besonders sicher und zur Speicherung großer Mengen an Bitcoins für geeignet.

Paper-Wallet
: Die Bitcoins kontrollierenden Schlüssel können zur längerfristigen Speicherung auch ausgedruckt werden. Man nennt sie daher Paper-Wallets, auch wenn andere Materialien (Holz, Metall etc.) genutzt werden können. Paper-Wallets stellen eine einfache, aber sehr sichere Technik dar, um Bitcoins für eine längere Zeit zu speichern. Diese Offlinespeicherung wird häufig als *Cold Storage* bezeichnet.

Eine andere Möglichkeit der Kategorisierung von Bitcoin-Wallets ist ihr Grad an Autonomie und wie sie mit dem Bitcoin-Netzwerk interagieren:

Full-Node-Client
: Ein vollwertiger Client, oder kurz eine *Full Node* (also ein »vollwertiger Knoten«), ist ein Client, der die gesamte Historie aller Bitcoin-Transaktionen (jede Transaktion jedes Nutzers) vorhält, Wallets verwaltet und Transaktionen im Bitcoin-Netzwerk direkt initiieren kann. Eine Full Node deckt alle Aspekte des Protokolls ab und kann unabhängig die gesamte Blockchain und jede Transaktion validieren. Ein Full-Node-Client benötigt einiges an Ressourcen (z.B. mehr als 125 GByte Plattenplatz und 2 GByte RAM), bietet aber vollständige Autonomie und die unabhängige Verifikation von Transaktionen.

Leichtgewichtiger Client
: Ein »leichtgewichtiger« (Lightweight) Client, auch bekannt als SPV-Client (*Simple Payment Verification*), stellt die Verbindung zu den oben erwähnten Full Nodes her, um auf die Bitcoin-Transaktionsdaten zuzugreifen. Er speichert die Wallet des Nutzers aber lokal ab und kann unabhängig Transaktionen erzeugen, validieren und übertragen. Leichtgewichtige Clients interagieren ohne Vermittler direkt mit dem Bitcoin-Netzwerk.

API-Client
: Ein API-Client interagiert mit Bitcoins über ein System eines Drittanbieters mithilfe von APIs (*Application Programming Interfaces*), statt eine direkte Verbindung mit dem Bitcoin-Netzwerk herzustellen. Die Wallet kann beim Nut-

zer oder auf den Servern des Fremdanbieters liegen, doch alle Transaktionen laufen über den Drittanbieter.

Kombiniert man diese Kategorien, fallen viele Bitcoin-Wallets in nur wenige Gruppen, von denen die gängigsten der Desktop Full Client, die mobile leichtgewichtige Wallet und die Web-Wallet eines Fremdanbieters sind. Die Grenzen zwischen den verschiedenen Kategorien sind oft etwas unscharf, weil viele Wallets auf mehreren Plattformen laufen und mit dem Netzwerk auf unterschiedlichen Wegen kommunizieren können.

Für die Zwecke dieses Buchs werden wir die Verwendung einer Reihe von Bitcoin-Clients demonstrieren, die man herunterladen kann. Das umfasst neben der Referenzimplementierung (Bitcoin Core) auch mobile und Web-Wallets. Einige der Beispiele verlangen den Einsatz von Bitcoin Core, das neben einer Full Node auch APIs für die Wallet, das Netzwerk und Transaktionen zur Verfügung stellt. Wenn Sie sich die Programmierschnittstellen für das Bitcoin-System ansehen wollen, müssen Sie Bitcoin Core oder einen der alternativen Clients (siehe »Alternative Clients, Bibliotheken und Toolkits« auf Seite 53) verwenden.

Schnelleinstieg

Alice, die wir in »Bitcoin: Anwendungsfälle, Anwender und deren Geschichten« auf Seite 5 bereits vorgestellt haben, ist keine technisch versierte Nutzerin und hat erst jüngst über ihren Freund Joe von Bitcoin gehört. Während einer Party schwärmt Joe mal wieder von Bitcoin und bietet eine Demonstration an. Neugierig fragt Alice, wie sie mit Bitcoin beginnen kann. Joe hält eine mobile Wallet für neue Nutzer für die beste Möglichkeit und empfiehlt einige seiner Lieblings-Wallets. Alice lädt sich *Mycelium* für Android herunter und installiert es auf ihrem Telefon.

Während Alice Mycelium zum ersten Mal ausführt, richtet die Anwendung (wie viele Bitcoin-Wallets) automatisch eine neue Wallet für sie ein. Alice sieht die Wallet auf ihrem Bildschirm, wie in Abbildung 1-1 gezeigt. (Hinweis: Senden Sie keine Bitcoins an diese Adresse, sie wären für immer verloren!)

Der wichtigste Teil dieses Screenshots ist die *Bitcoin-Adresse* von Alice. In der Abbildung ist sie als langer String aus Buchstaben und Ziffern zu erkennen: `1Cdid9KFAaatwczBwBttQcwXYCpvK8h7FK`. Neben der Bitcoin-Adresse der Wallet sieht man einen QR-Code, eine Art Barcode, der die gleichen Informationen in einem Format enthält, das von einer Smartphone-Kamera gescannt werden kann. Der QR-Code ist das Quadrat mit einem Muster aus schwarzen und weißen Punkten. Alice kann die Bitcoin-Adresse oder den QR-Code in ihre Zwischenablage kopieren, indem sie den QR-Code oder den *Receive*-Button antippt. Bei den meisten Wallets wird beim Antippen des QR-Codes dieser auch vergrößert, sodass er von einer Smartphone-Kamera einfacher eingescannt werden kann.

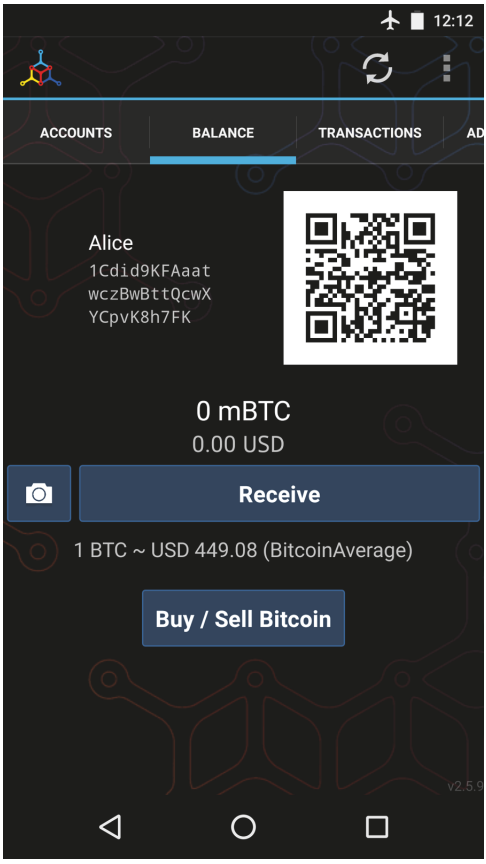


Abbildung 1-1: Die mobile Wallet Mycelium

Bitcoin-Adressen beginnen mit einer 1 oder einer 3. Wie E-Mail-Adressen können sie mit anderen Bitcoin-Nutzern geteilt werden, die ihnen dann Bitcoins direkt an ihre Wallet schicken können. Sicherheitsaspekte spielen bei Bitcoin-Adressen keine besondere Rolle. Man kann sie überall verteilen, ohne die Sicherheit des eigenen Kontos zu gefährden. Im Gegensatz zu E-Mail-Adressen können so viele Adressen erzeugt werden, wie man möchte, die alle Zahlungen an die eigene Wallet weiterleiten. Tatsächlich erzeugen viele moderne Wallets automatisch eine neue Adresse für jede Transaktion, um die Vertraulichkeit zu erhöhen. Eine Wallet ist einfach eine Sammlung von Adressen sowie der zugehörigen Schlüssel, die die darin enthaltenen Gelder freigeben.

Alice ist nun bereit, Zahlungen zu empfangen. Ihre Wallet-Anwendung hat einen zufälligen privaten Schlüssel erzeugt (der in »Private Schlüssel« auf Seite 60 ausführlicher beschrieben wird), zusammen mit der zugehörigen Bitcoin-Adresse. An