IN THIS CHAPTER

- Getting to know Ethereum
- Understanding ether and trading
- Exploring DAOs and ICOs
- Setting up an Ethereum development environment
- Developing blockchain applications

Chapter 1
Introducing Ethereum

Blockchain technology is the most disruptive technology introduced in our generation, and Ethereum is by far the most popular blockchain implementation in use today. You can't read many technology articles or blogs without seeing something about how blockchain changes everything. Although some claims seem to be a little far-fetched, blockchain technology really is a game-changer.

Blockchain, which first burst on the scene in 2008, has gained global notoriety for what it has already changed and what is coming. At first, blockchain was all about a new type of electronic currency. But now, partially thanks to Ethereum, blockchain is so much more than a new way to pay for things. It's a new way to think about things. It enables people and businesses to conduct business without many of the obstacles that have existed in trade relations for centuries.

In this book you learn about what blockchain is and why it is viewed as so radical. You discover how powerful Ethereum is in diverse domains and how you can harness its promise and power in your own organizations. If you want to learn what Ethereum is and how it can work for you without having to trudge through hundreds of pages of theory and background, this is the book for you.

Describing Blockchain Technology

You learn a lot more about blockchain technology in Chapter 2, but before you meet Ethereum, you need to know a little of Ethereum's backstory.

TIP

If you already know what blockchain is, this section will be like watching yet another depiction of why Bruce Wayne became Batman. Feel free to skim it and move on to the next section. There are only so many ways you can kill Thomas and Martha Wayne.

Blockchain technology was introduced to support a new type of digital currency that you can trade in a trustless environment. Traditional currency exchanges require a trusted third party between the parties. Even if a buyer provides coins or bills to a seller at the point of transaction, some government provides the guarantee of the currency's value. There is always a middleman. If the exchange involves a payment card or check, other financial institutions participate to handle the transfer of funds between parties.

In 2008, Satoshi Nakamoto published a paper that changed everything. Nakamoto's paper described a new way to store and distribute data with verifiable integrity among a group of nodes that don't trust one another. You learn more about how Nakamoto's proposal works, and about bitcoin, the cryptocurrency proposed in the paper, in Chapter 2. At this point, the most important takeaway is that this paper showed how to take the requirement for a trusted (and omnipotent) central authority out of the mix. Using this new technology, called blockchain, application developers can create environments in which nodes that do not trust one another can share data that they can trust.

The idea is based on several concepts that are simple to consider but difficult to put into practice. First, data is logically presented as a ledger. The data isn't really stored that way; you can just think of it as a ledger. A ledger is a way of recording data as transactions occur. One interesting feature of this ledger is that you can only add data to it. You can't change anything after you've added it. So, the only two operations you can perform on this ledger are add and read. We refer to the "add only" property as the *immutability property.* In short, blockchains are immutable. As you'll see, immutability is crucial for the technique to work.

Another feature is that data is added to this ledger in blocks. *Blocks* are collections of transactions, each with an owner's address. *Addresses* are the unique IDs of accounts in our ledger system. When there are enough transactions to make a new block, some of the blockchain participants begin a process of adding a new block to the ledger. Each new block is linked to the previous block, making a chain. That's where the term blockchain comes from. A *blockchain* is basically a bunch of blocks where each block is connected to its predecessor.

Then, the entire set of blocks, or the entire blockchain, is shared with other participants. These participants are called *nodes.* These nodes communicate with one another and each stores an exact copy of the blockchain. Many blockchain networks are made up of thousands of nodes, and keeping all of the copies of the blockchain in *sync* (that is, ensuring that every copy is the same) is another revolutionary feature of blockchain technology.

Blockchain technology is built on a *democracy governance mode.* Before any new block is added to the blockchain, a majority of nodes must agree that the new block is valid. All nodes agree to accept the majority decision. That's how the blockchain stays in sync. Nodes essentially vote on all new blocks. Different blockchains use different voting methods, but one of the more common ones requires nodes to solve very hard mathematical puzzles to earn the right to add a new block to the blockchain. As an incentive for doing the hard work, the node that solves the puzzle first gets a reward. The reward encourages nodes to pitch in and help do the hard work of solving verification puzzles.

Part of the puzzle solution involves creating a mathematical hash of the previous block. By storing the previous block's hash in the current block, every node can quickly determine if any block has changed. Each node periodically scans the blockchain to ensure that nothing has changed. This is how nodes can be sure that the blockchain is the same across the entire network. And, because no block can change after it is added to the blockchain, you never have to worry about overwriting data.

Putting it all together, a blockchain makes it possible to share a set of data with many nodes that you don't trust. You can trust the democracy of the network, though. As long as you can trust that more than half of the nodes on the blockchain network are going to be honest, you can trust the blockchain.

The last big advantage to blockchain technology is that you can put rules of operation in blocks on the blockchain as well. These rules are called *smart contracts*. A smart contract is just a program that lives in a blockchain block and governs how data is added to the blockchain. Because all blockchain data is immutable, even the smart contract code is immune from changes. That's how you can exchange currency without a bank. As long as there are rules that dictate how a currency exchange is carried out, transaction data can be recorded on the blockchain and be part of the permanent ledger.

For example, suppose you want to buy a car. You have enough digital currency in your blockchain account to buy the car, and the car owner has the car's title stored in the blockchain. You can offer to buy the car and if the seller accepts your offer, a smart contract handles the transaction. The smart contract would verify that the title is owned by the seller and that you have enough money in your account to

make the purchase. If those two requirements are met, the smart contract would transfer the sales amount into the seller's account and transfer the title to your account. Without any middleman, you have purchased a car and paid for it without carrying a wad of cash around.

TIP

Of course, you really purchased a title to a car. Blockchain handles digital assets. You still have to physically get the keys and the car from the seller.

Introducing Ethereum

Bitcoin was the first blockchain technology application. It was revolutionary and defined the first widely used digital currency, called *cryptocurrency*. The *crypto* part of the name refers to the use of cryptographic hashes to ensure the integrity of the blockchain. The shared ledger literally keeps a copy of every cryptocurrency transaction that gets verified by all nodes. Using this approach, bitcoin created a permanent record of every exchange of their cryptocurrency. And, because account owners are identified only by an address, bitcoin has always enjoyed a measure of anonymity.

TECHNICAL STUFF

Although bitcoin addresses aren't linked directly to people, many exchanges have records of identities that are related to addresses. At some point, you have to exchange your cryptocurrency for real currency. That switchover point is where many law enforcement officials focus when they're trying to track down criminals using cryptocurrency.

As bitcoin became more and more popular, researchers began to see more applications for blockchain technology beyond cryptocurrency. In 2013, Vitalik Buterin, the cofounder of *Bitcoin Magazine,* published a whitepaper that proposed a new, more functional blockchain implementation. This new proposal was for the Ethereum blockchain. After gaining interest and attracting technical and financial support, the Ethereum Foundation, a Swiss non-profit organization, was founded and became the developer of Ethereum.

Ethereum wasn't created just to exchange cryptocurrency. In fact, it was designed from the beginning to be different. The core features of Ethereum are the smart contract and ether. *Ether* is the native cryptocurrency that Ethereum supports, although you can create your own tokens to exchange value in many other forms. Smart contracts provide an execution environment that ensures integrity across all nodes. Any code that executes on one node executes the same way on all nodes. This guarantee makes it possible to deploy a wide range of applications across untrusted environments.

The foundational guarantees Ethereum provides support many types of value exchanges without the concern about fraud, censorship, or any involvement by a third party. When you interact with an Ethereum application, you don't have to rely on any intermediary to broker your transactions. You don't need a bank, wholesaler, or transaction broker to provide trust. As a result of Ethereum's *disintermediation,* you can often complete transactions faster, with far lower service fees and without requiring approval from external authorities.

Ethereum is a comprehensive, decentralized application platform that expands the range of capabilities beyond what was possible before blockchain technology. Whereas legacy solutions to data and process sharing required third-party authorities to enforce integrity, Ethereum provides process and data integrity, along with disintermediation. The possibilities are just beginning to be explored.

Exploring Ethereum's Consensus, Mining, and Smart Contracts

Ethereum provides integrity in the way it implements immutability and smart contracts. Immutability isn't actually a blockchain guarantee. You can change data in any block — even after other blocks are added to the blockchain. However, as soon as you change a block, that block and all subsequent blocks fail integrity checks and your node is out of sync. Instead of saying that the blockchain is immutable, it is more accurate to say that any changes (mutations) to the blockchain are easily and immediately detected.

Ethereum is based on democracy. Each node gets an equal vote. Every time nodes get a new block to add to the blockchain, they validate the block and its transactions, and then vote whether to accept or reject the block. If several different blocks are submitted by different nodes, only one of the blocks can receive votes from a majority. The block that gets more than half of the network node's votes gets to join the blockchain as its newest block.

One of the first problems is to determine when a new block is ready for the blockchain. When too many conflicting blocks are submitted, the voting process slows down. Ethereum makes it hard to add new blocks to keep the number of new block collisions low and to make voting faster. Ethereum uses a consensus protocol called *Proof of Work (PoW),* which sets the rules for validating and adding new blocks. PoW makes add blocks to the blockchain difficult but profitable.

Ethereum defines ether as its cryptocurrency. You can transfer ether between accounts or earn it by doing the hard work of adding blocks to the Ethereum blockchain. The Ethereum PoW mechanism requires that nodes find a number that, when combined with the block's header data, produces a cryptographic hash value that matches the current target, which is a value that is adjusted to keep new block production at a steady rate. Finding a hash value that matches the current target is hard. You have to try on average more than a quadrillion values to find the right one. That's the point. Using a PoW mechanism makes it so hard to submit a block that fewer blocks are submitted, which reduces the number of collisions. The node that finds the right value gets a small ether payment for the effort. This process is called *mining,* and the node that wins the prize is that block's *miner.*

Mining regulates the speed at which new blocks get submitted as candidate blocks, and results in a number that is easy to validate. Finding the right number to solve the puzzle is difficult, but verifying the number is fast and easy. Another interesting aspect of mining is that each block's header contains a hash from the previous block. Ethereum nodes use the hash to easily detect unauthorized block changes. If a block changes, the hash result doesn't match and the block becomes invalid.

Mining is also a way to make money using blockchain technology. Mining has become competitive, and most of today's miners invest in high-performance hardware with multiple GPUs to carry out the complex operations. To keep the mining process fair, Ethereum uses a complexity value that makes the mining process even harder as miners get faster. Adjusting the complexity allows Ethereum to regulate the new block frequency to an average of one new block every 14 seconds.

The glue that holds the Ethereum environment together is the smart contract. Ethereum is much more than just a financial ledger, and smart contracts provide much of its rich functionality. Each Ethereum node runs a copy of the *Ethereum virtual machine (EVM).* The EVM runs smart contract code in a way that guarantees that smart contracts execute the same way on all nodes and produce the same output. Running smart contract code is not optional. Smart contracts execute based on specific rules and cannot be subverted or halted. The EVM smart contract guarantees provide a stable platform for automated transaction processing that you can trust. Smart contracts provide the primary power of the Ethereum environment.

One of the known weaknesses with software is that attackers can sometimes bypass its controls and carry out unintended actions. That type of attack is more difficult in Ethereum, primarily due to its smart contract implementation. Attackers can't directly attack the blockchain and make unauthorized changes because any such changes will be immediately detected. The next most likely attack vector is the smart contract interface to the blockchain data. Ethereum guarantees that

smart contract code, which is translated into *bytecode* before it is written to the blockchain, executes on every EVM instance the same way. Also, the EVM determines when code executes and what code executes. Attackers have few opportunities to leverage smart contract code, which makes Ethereum an even more secure environment.

Buying, Spending, and Trading Ether

Ethereum runs on ether (ETH), its main cryptocurrency. The majority of all existing ether was pre-mined when Ethereum first went live on July 30, 2015. Miners continually create ether, but the amount of mined ether is less than 30 percent of all ether in existence. The lifecycle of Ethereum transactions requires that you first acquire ether to participate in Ethereum. Many exchanges support exchanging legal tender, also called *fiat currency,* for cryptocurrency, including ether. You can navigate to `https://99bitcoins.com/best-ethereum-exchange-review-comparison` for an independent comparison of several popular exchanges.

Before you can interact with the Ethereum blockchain, you need to create at least one account. Creating an Ethereum account is essentially just creating a cryptographic private and public key pair, and generating the associated address, which is based on your public key. The software that handles this process is called an Ethereum *wallet.* You learn about different options for Ethereum wallets in Chapter 6. You can use a wallet provided by an exchange or a standalone wallet. After you create your Ethereum account, you'll need to select an exchange to purchase ether.

After you select an exchange, you set up an exchange account and provide a funding source. Your main funding source is generally a bank account. The most common way to buy ether is to withdraw funds from your bank account and use that money to exchange for ether. Figure 1-1 shows the purchase ether web page for the coinbase.com exchange. Note that the funding source for this account is a Bank of America account.

You can also purchase ether using cash. A growing number of cryptocurrency ATMs allow you to exchange cash for different types of cryptocurrency. All you need is the private key you generated using your Ethereum wallet and cash. However, you will pay for this convenience. Cryptocurrency ATMs often use exchange rates that are less favorable than more traditional exchanges. One current service, localcoin ATM, works just like a regular ATM. Navigate to `https://localcoinatm.com` to see where you can find ATMs and how to use them. Figure 1-2 shows several steps in the process of purchasing ether with cash from an ATM.

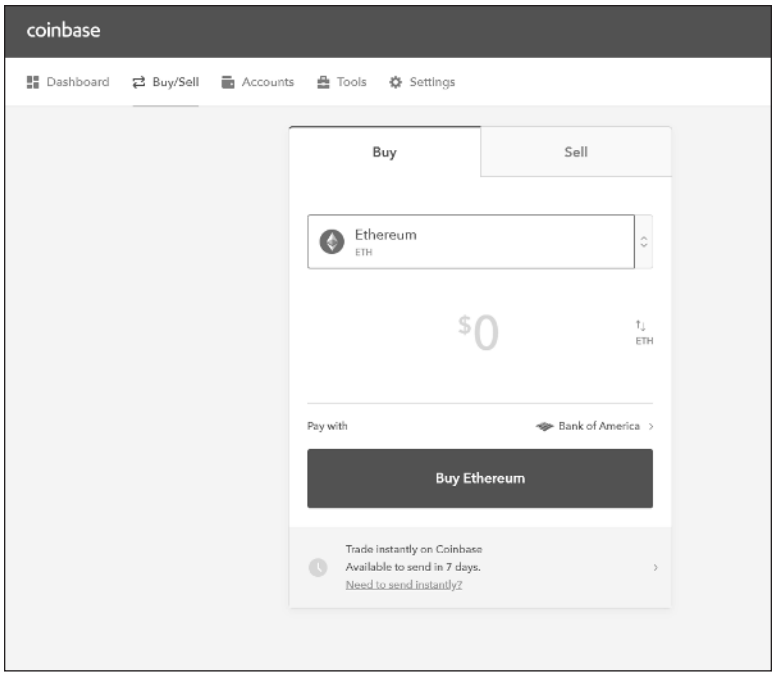

FIGURE 1-1: Purchase ether using coinbase.com.

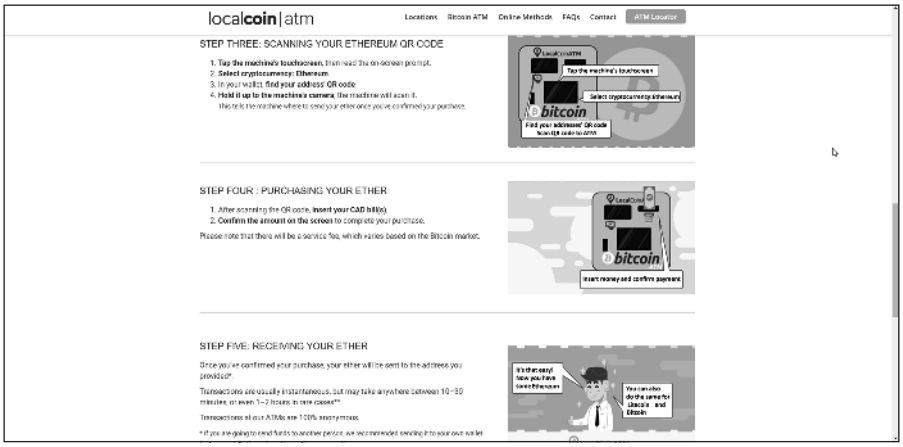

FIGURE 1-2: Purchasing ether with cash.

After you own ether, you can interact with other Ethereum accounts and send them some of your ether in exchange for good or services. Or you can simply hold on to your ether in hopes that is goes up in price. Ether, along with other cryptocurrencies, fluctuates in price continuously. Many investors buy and sell

cryptocurrencies as investments, just like trading fiat currencies or commodities. Figure 1-3 shows the main coinbase.com dashboard with popular cryptocurrency prices.

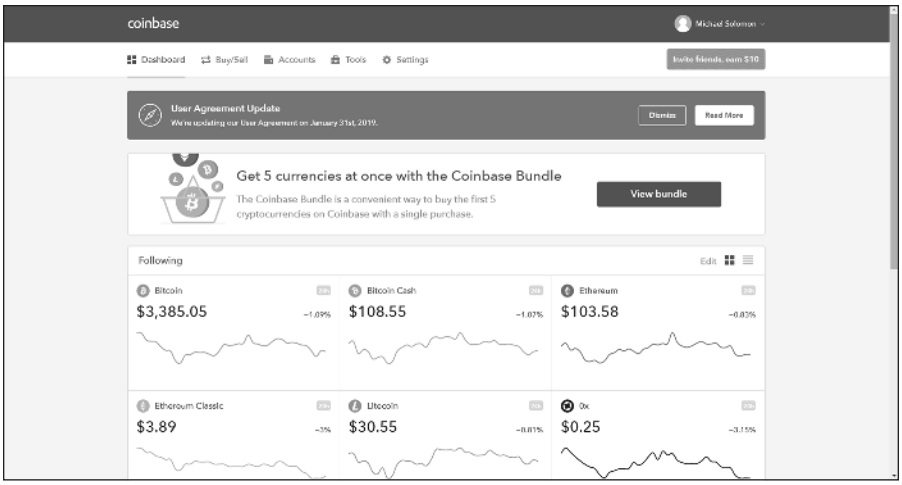

FIGURE 1-3: Current cryptocurrency prices.

At its highest price, ether sold for around $1,400. At the time of this writing, it was down near $100. Whether cryptocurrency is a good investment depends on your appetite for risk and belief in its long-term value.

In addition to buying and trading ether, you can spend it just like any other currency. Of course, you generally have to buy from a vendor that accepts ether. Several service providers make it easy to accept payments with ether, such as Pay with Ether. This company provides the software and the services to make it easy for vendors of any size to accept ether as payment. Visit `www.paywithether.com/` to find out more about this payment option.

TECHNICAL STUFF

There are ways to spend cryptocurrency at vendors that don't directly accept it. Several companies are planning to offer Visa cards that you fund with cryptocurrency. One company, Wirex, allows users to convert their cryptocurrency to USD, GPB, or EUR and use their card at any vendor that accepts Visa.

Cryptocurrency is growing rapidly, but only a small number of vendors accept it. If you really want to pay with ether or other cryptocurrencies, take a look at TenX. This company offers a popular Visa card funded by cryptocurrency. The card isn't available everywhere, but the company expects to increase its availability over time. Navigate to `https://tenx.tech/en` for more information on TenX and their cryptocurrency payment card.

Getting Started with DAO and ICO

Blockchain technology has given rise to new classes of organizations and opportunities. You'll often hear about *decentralized autonomous organization (DAO)* and *initial coin offering (ICO).* These terms simply describe endeavors that Ethereum makes possible. You'll read a lot about these terms as you learn more about Ethereum, so it makes sense to cover them here.

A *DAO* is an organization that operates only on the rules set forth in its smart contracts. In reality, most DAOs require some human interaction, but the majority of the functionality is automated. For example, assume in just a few years that autonomous vehicles (driverless cars) are more common. A DAO would be like a driverless Uber or Lyft car. The car waits for a passenger, and then drives to the pickup location when someone needs a ride. The autonomous car completes the trip and the passenger pays with cryptocurrency. The car just earned some money. However, the car's maintenance smart contract detects that the brakes need replacing. So the car drives itself to a mechanic and pays for new brakes, using the profits of previous rides. The autonomous vehicle does not need human interaction to carry out its primary business function or to get necessary service. The autonomous vehicle is the same idea as a DAO.

A DAO conducts business and engages in transactions without requiring human interaction. Today's DAOs are relatively simple, but it is expected that they will grow in complexity and eventually replace (or at least compete with) some existing human-based businesses.

Like all businesses, Ethereum-based or Ethereum-related businesses need funding to operate. Many traditional methods for raising funds exist, including soliciting private investors, securing loans, or selling shares in the company. In addition, Ethereum opens new options for funding businesses.

Businesses that use Ethereum often create their own tokens, also called coins, that represents value associated with the business ventures. Businesses sell these tokens to raise funds to launch the business. These *ICOs* essentially exchange one type of currency for a digital item of value. Tokens may represent an expected future value as ownership in a new venture or current value that entitles the holder to some benefit. Either way, tokens are similar in some ways to stock shares. An ICO is a popular method to fund a new blockchain-based business. If you want to learn more about the most popular ICOs, navigate to `www.coindesk.com/ico-tracker` to explore coindesk's ICO Tracker, shown in Figure 1-4.

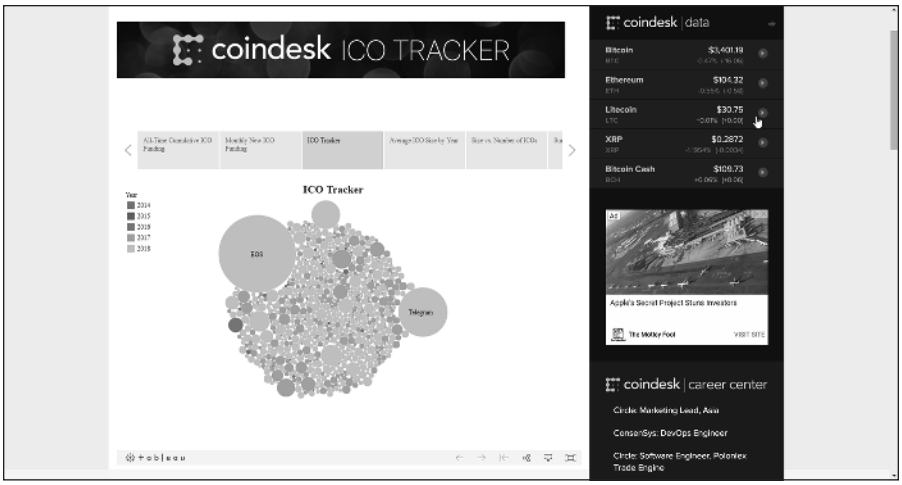

FIGURE 1-4: coindesk ICO Tracker.

Exploring the Ethereum Ecosystem

The *Ethereum environment*, or ecosystem, is made up of several different parts. You've already learned about most of these pieces, but it helps to put those pieces together in one place. Starting from the lowest level of the blockchain, Ethereum is made up of the following components:

- **Blockchain:** The collection of data blocks that is the core of Ethereum. Each block contains data and smart contract code, and is cryptographically linked to its predecessor, creating a chain of blocks.
- **EVM:** The Ethereum virtual machine, which runs smart contract bytecode. Each Ethereum node runs an instance of the EVM.
- **Wallet:** Software, hardware, or physical paper that stores the public and private keys that correspond to an Ethereum account. The wallet stores the capability to access crypto-assets on the Ethereum blockchain.
- **Exchange:** A service the allows its users to exchange fiat currency for cryptocurrency.
- **Development environment:** The set of tools to write, compile, and perform unit tests on smart contract software.
- **Testing environment:** A simulated Ethereum blockchain used to perform integration testing on smart contracts and complete decentralized applications (dApps).
- **Client interface:** The client user interface's software and libraries used to interact with Ethereum smart contracts.

You learn much more about the details of each of these components in Chapter 4.

Delving into Development Tools

Developing decentralized applications (dApps) for Ethereum requires several types of tools. Each of these tools provides support for various phases of software development and is necessary to create dApps for the Ethereum blockchain environment. In Chapter 4, you find out the different categories of development tools and learn about several alternatives for each type of tool.

Tools that you'll use when developing Ethereum dApps are in the following categories:

- **Blockchain client:** When developing Ethereum dApps, you'll need to implement a local EVM. A blockchain client launches a local EVM and executes your smart contract code. It also interacts with your Ethereum blockchain.
- **Development and testing blockchain:** Deploying to the live Ethereum blockchain, also called *mainnet,* is the last step in the development process. Before deployment, you want to interact with a local version of an Ethereum blockchain. Because live blockchain access costs ether, you should carry out development and testing on a local blockchain to avoid costs and errors.
- **Compiler and testing framework:** After you have a local EVM and local blockchain, you'll need a way to interact with your smart contract code and place it on your test blockchain for test execution. A development and testing framework provides the tools you need to carry out development and testing tasks.
- **Source code editor/IDE:** Although you can use any text editor to write smart contract source code, an editor or integrated development environment (IDE) that is designed or extended to support Ethereum smart contract source code development will be very helpful. IDEs can increase developer efficiency and make it easier to create good smart contract code.

As you work through the exercises in this book, you'll learn more about each of these tools and install an example from each category.

Building Blockchain Apps

The process of building blockchain applications is not radically different from developing traditional applications. You have a few additional considerations and a few additional steps. As you work through the chapters in this book, you'll learn each of the tasks required to develop effective and efficient blockchain applications for the Ethereum environment.

The main differences between traditional applications and blockchain applications is the distributed, transparent nature of the data and the fact that writing data to the blockchain costs money. Distributed and transparent data means that you can't rely on the blockchain to provide any confidentiality. Anything you write to the blockchain is visible by users on any node. That should affect how you design your applications and data. Be careful when collecting data from users and storing it on the blockchain. Always assume that any blockchain data is available to the public.

The other main difference is that writing to the blockchain costs money. The development and testing processes normally occur with simulated blockchains that use only fake cryptocurrency, but live blockchain I/O has a real cost. Most developers aren't used to calculating computation and access costs. Because this is generally a new concept, many developers may put this consideration off until later in the development cycle. This decision is a mistake. It is always easier and cheaper to address blockchain access cost problems early in the development process. If you wait until the end, any changes will likely have cascading effects and costs.

As with any good development practices, do everything you can to fully understand what your users want and need. Design your software with the users in mind and always cater to their needs first. Then make you software meet their needs in ways that are the most effect and economical. Stick with good software design and development principles, while incorporating blockchain-specific considerations, and you'll be on your way to developing a good blockchain dApp.