Blockchain and Ledgers and Cryptos – Oh My!

by Dr. Philip Pyburn
ppyburn@techpargroup.com

The basic concepts behind blockchain are relatively straightforward: a distributed database (the "ledger") that maintains a complete transaction history on multiple nodes of a network.

Confusingly, much that has been written about blockchain conflates this simple concept with the *applications* of blockchain to things like cryptocurrencies (Bitcoin, Ethereum, Ripple etc.), smart contracts, trans-border payments, and securities settlements. In this article, we will try to clarify the differences and provide a high-level understanding of how blockchain works.

Blockchain: Not your Grandfather's Distributed Database

So, what is it about blockchain that makes it more powerful than the distributed database technologies that have been around for years? While there are a number of technical differences, from a business point of view we believe three aspects of blockchain set it apart – and above – the conventional distributed database:

- Blockchain comprises an immutable and comprehensive transaction history that is simultaneously stored on every node on the network. Any change to a transaction on one node that is not validated by the other nodes will be rejected. Also, because every transaction is visible to every node in the network, there is no need for a central authority to validate transactions and enforce trust among all parties.

- The data contained within each block in a blockchain is strongly encrypted. This allows parties to a transaction to maintain anonymity while still giving the network the visibility needed to validate each transaction.

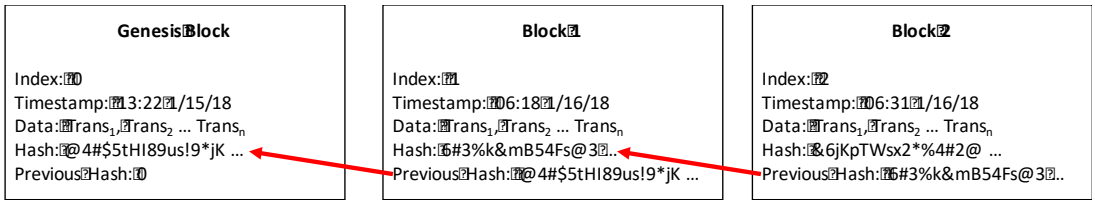
- The integrity of the blockchain is confirmed by a consensus of the nodes in the network. If a node is hacked and an attempt is made to modify a block (e.g. fraudulently change the ownership of an asset), the modified block will not match the same block as it is recorded on the ledgers of the other nodes. As a result, the change will be identified as invalid and will not be posted to the other ledgers. Depending upon the needs of the application, this consensus can require a plurality, a majority, a super-majority, or even "all but one" nodes.

In combination, these factors ensure that a reasonably large blockchain network is one of the most secure data stores imaginable and almost completely hack-proof. To successfully modify a transaction in a block on a blockchain would require the hacker to simultaneously decrypt records on hundreds or thousands of nodes, a task beyond even the most sophisticated current or contemplated capabilities. In addition, the more deeply embedded a block is within a chain (i.e., the older it is), the harder it is to hack because all of the subsequent blocks that have been added to the chain would have to be modified in addition to the data in the target block.

Creating a Blockchain

The programming to create a blockchain is actually quite straightforward, and at its simplest, involves just seven steps[1]:

1. **Blockchain Architecture**: Determine how the blockchain will be structured. At a minimum, each block will need to contain an index, a timestamp, the substantive transaction data, a hash value for the new block and the hash value from the previous block so each block can be linked in a chain.



2. **Cryptographic Algorithm**: A cryptographic hashing algorithm to be used to secure each block. The most widely used is SHA-256, which generates 32-character hash keys.

3. **Block Generation**: Code to create a block by determining the hash of the previous block and providing a mechanism for the end user to add the additional block data.

4. **Block Storage**: Code to create an array that stores an identical copy of the "chain of blocks" (i.e. the blockchain) on each node.

5. **Block Validation**: Code to validate an individual block or a chain of blocks to determine whether it should be added to the chain on each node.

6. **Conflict Resolution**: Code to resolve block and chain conflicts created, for example, when two nodes create a block with the same index.

7. **Communication**: Code to allow each node to share and synchronize the blockchain with the other nodes in the network.

While this may appear complex, a rudimentary blockchain called NaiveChain with all of this functionality was developed by Lauri Hartikka in a mere 200 lines of JavaScript.

---

[1] The fundamental elements of a blockchain are taken from a blog by Lauri Hartikka; https://medium.com/@lhartikk/a-blockchainblockchainblockchain-in-200-lines-of-code-963cc1cc0e54

Public vs. Private Blockchains

Blockchains come in two flavors: Public and Private.

Private Blockchains

A private blockchain comprises only nodes that have permission to run the applications and participate in the network. For example, a consortium of banks might develop a blockchain application to share anti-money laundering (AML) information to improve detection and reduce costs. In a private blockchain, the operators of each node in the network (in this case, the banks) must pay for the cost of the application and for the costs of operating each node. As a result, private blockchains are most suitable for applications where the operators of the nodes derive benefits from their participation in the form of increased revenue or decreased costs. In addition, some executives believe that a private blockchain is more secure because all of the participants are known and vetted, but for the reasons noted earlier, the blockchain distributed database/ledger creates its own security, regardless of the trustworthiness of the nodes.

The most important shortcoming of private blockchains shows up in applications where there are significant network effects. For example, if an AML application for a consortium of 50 large banks can cut AML costs in half, then an application operating on a *public* blockchain with several thousand institutions (including non-banks) could cut costs by a factor of 10 or more.

Public Blockchains

The implications of these network effects lead us directly into a consideration of public blockchains. Unlike private blockchains, a public blockchain is open to anyone who is willing to download and operate an application. For example, to operate as a node on the Bitcoin network one need only download and set up the application to validate transactions and maintain copies of the Bitcoin ledger.

However, operating a public blockchain node involves some significant costs for which the operator of the node will want to be compensated. Because a complete history is maintained of every transaction since the chain was created, large blockchains require a significant amount of storage and the means to access that storage quickly. In addition, because a public blockchain is open to all comers, security, transaction validation and the integrity of the chain are more critical than with a private blockchain. Without getting into the cryptographic weeds, suffice it to say that just the electric power consumption of the required graphics processing units (GPU's) is enough to dissuade many casual players from participating.

So, how do the operators of a public blockchain get compensated?

As I noted at the outset, many people conflate blockchain and cryptocurrencies. And while it is true that most public blockchains run cryptocurrency applications, the currencies themselves are not technically required for a non-currency application of blockchain.

For cryptocurrency blockchains, the incentive to operate a network node is straightforward: The node that solves a cryptographic puzzle first, thereby earning the right to post the block to the ledger, receives a small fraction of the cryptocurrency as a reward. In this process, called "mining," the fractional cryptocurrency is not paid by any group or individual, but rather is created by the cryptocurrency application itself.[2]

For example, the Ethereum blockchain is used to support Ether cryptocurrency transactions (at current writing, worth about $1,000 USD per Ether). However, the Ethereum blockchain is also well suited to many kinds of self-executing contracts[3] where the Ether cryptocurrency itself is not required. But without Ether, what is the incentive to operate an Ethereum node? At least for the time being, public blockchain applications will be implemented alongside the cryptocurrency of that blockchain implementation.

Summary

Whether cryptocurrencies like Bitcoin, Ripple and Ether expand or implode is a question that is very much open for debate. But either way, it is important to remember that these cryptocurrencies are just one application of a technology that has enormous potential across a broad range of industries. Almost every situation where there are intermediaries involved in a business transaction are ripe for improvement using blockchain technology.

Our intent in this article was not to explore all of the nuances of blockchain, but rather to show how it is an important evolutionary step in the development of distributed database technologies. It provides the platform upon which applications can be developed that solve all sorts of real-world business problems. For us, the exciting challenge over the next couple of years will be identifying, analyzing and deploying blockchain applications that dramatically reduce costs and create new service opportunities that aren't possible today.

I would like to thank Lee Gruenfeld (lee@leegruenfeld.com) and Baruch Plagman (bplagman@techpargroup.com) for their conceptual and editorial contributions to this article. All errors and omissions are my own.

[2] One of the important differences between various cryptocurrencies is the extent to which the maximum number of coins or tokens is fixed. The Bitcoin application, for example, limits the total number of Bitcoins to 21 million. Other cryptocurrencies have no such limit.

[3] In 1996 computer scientist Nick Szabo introduced the concept of a "smart contract" where software, rather than attorneys and courts, would ensure that contracts were properly executed. For example, a "smart contract" blockchain application for real estate transactions might be used to record the receipt of funds, the distribution of funds to the various parties, and the transfer of title without the need for a settlement agent, attorney or land registry.