The Blockchain 1.0: Currency

Karan Bharadwaj

October 7, 2016

Contents

1 Understanding the Blockchain

1.1 Introduction

Over the past few years, an important innovation colloquially known as the "blockchain" has emerged as a potentially disruptive technology. The core of the innovation is built around the concept of a distributed cryptographic database. The database, also referred to as the ledger, is maintained by a network of computers. The ledger makes it possible for the entire network to create, evolve and keep track of an immutable record of transactions. The most successful blockchain application thus far has been Satoshi Nakamoto's cryptocurrency known as Bitcoin, which he outlined in his seminal paper, "Bitcoin: A Peer-to-Peer Electronic Cash System" in 2008. This powerful technology has so far been been implicated in the numerous unregulated cyrptocurrencies that exist online. Financial institutions are only beginning to understand the potential applications of blockchains in conventionally regulated industries.

1.2 Features

1.2.1 Distributed

The blockchain is also referred to as a distributed ledger. This is both because a large number of nodes on the blockchain network posses their own copy of the entire ledger and that they are connected to one another. This creates a distributed arrangement of nodes. This is an important design choice because when a block of transactions is ready to be uploaded onto the blockchain, the individual nodes participate in a voting mechanism to decide which node, called the 'leader', makes the addition. This entire process is based upon a networking paradigm derived from BFT or Byzantine Fault Tolerance algorithms(see section 1.3.1). All nodes in this distributed system have access to a subset of all available nodes and are able to re-connect and update their individual ledger in the case of any downtime. Certain groups of nodes are stable nodes and often given the label 'genesis' nodes to demarcate the fact that the first block was created at that particular node and shared with other nodes.

1.2.2 Cryptographic Security

The blockchain is designed such that transactions are created using public key encryption(see section 1.3.2). Each account has a private key associated with it. The public key derived from the private key is used to sign transactions. The private key establishes ownership of the token transferred by the transaction. The use of public key encryption means that if someone were to get hold of the transaction data through malicious means, they would be unable to derive any information from it because the private key for any account is never exposed. The data stays encrypted and is of no use to the attacker. There is a second layer of SSL encryption that transfers the transaction data between nodes. So complete privacy is guaranteed at both the level of transaction data as well

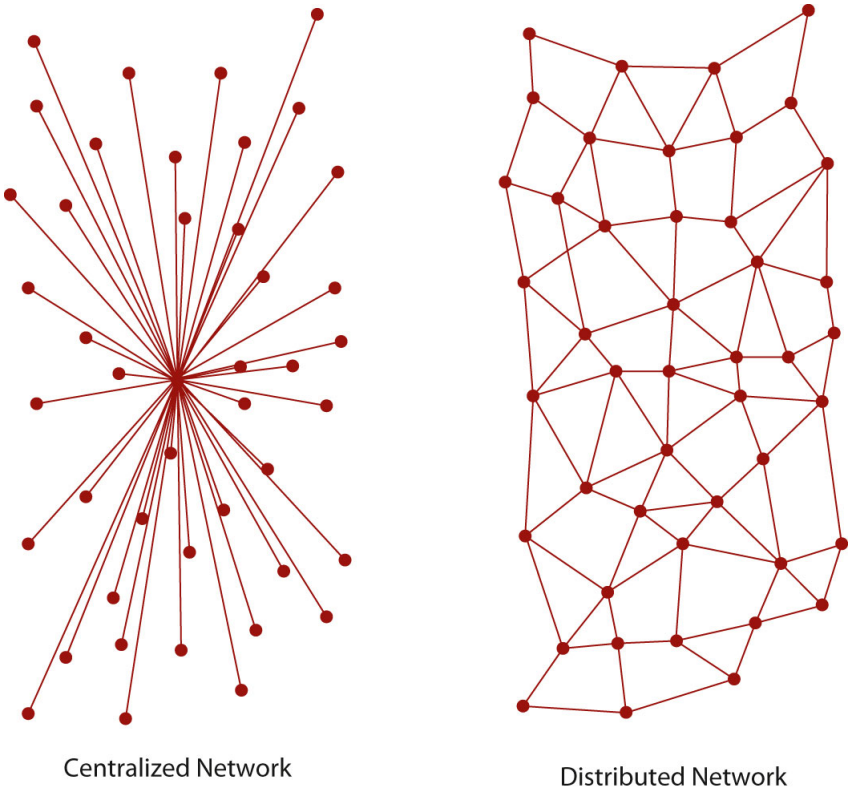

Figure 1: Networks visualized

as the transmission of the transaction data. Any entities who must be able to lawfully decrypt the data can be permissioned appropriately. This could include the payment service players(PSPs), auditors, or agencies involved in anti money laundering monitoring. Certain observer nodes can be configured for this purpose.

1.2.3 Immutable

As suggested by it's name, the blockchain is a long collection of blocks that link to each other in a specific way. In most implementations, each new block that is added to the blockchain contains a pointer to the previous block called the parent block. This pointer is included in a specific area of the block, sometimes called the header. In the case of the Bitcoin implementation, a block is identified by a SHA256 hash of the header. The header also contains the pointer to the previous block- the previous block hash. This arrangement links the newly created block with it's parent block and so on till the very first block called the genesis block.

What this arrangement accomplishes is that if someone made an attempt to

maliciously alter a transaction somewhere in the blockchain, they would change the blocks hash. Since the block's hash is included in the block following it(the child block), they would then have to change the child's hash as well. The higher the position of the block that is altered, the more child blocks succeeding it have to be altered. This quickly becomes an impossible task since new blocks are being added constantly and any malicious individual or small group(of nodes) cannot compete with the honest nodes.

This gives rise to the property of immutability. Which means that the state object, once created, is unchangeable. While there are ways to change the blockchain, as in the case of hacks or upgrades, these changes are difficult to implement and require structural modifications rather than changes to the block transactions themselves because of the very same property of immutability.

1.3 Concepts

1.3.1 Byzantine Fault Tolerance

Byzantine fault tolerance is a property of computer systems that are faced with failures resulting from the Byzantine General's Problem.

The Byzantine General's Problem refers to a hypothetical scenario where consensus must be reached between participants. The problem statement is that a number of generals of the Byzantine army must coordinate their movements on the battlefield. The options available to them is to 'attack' or 'retreat'. The idea being that a coordinated 'attack' or 'retreat' is better than some generals attacking and being defeated and some retreating.

The circumstances are complicated because the generals are separated from each other and must send messages to communicate. The messengers sent out by the generals might entirely fail in delivering the message. Or certain generals could be traitorous. In such a case where there are nine generals and four apiece vote for 'retreat' and 'attack', the ninth general can vote for a losing strategy or perhaps send different messages to either group. In either scenario the rogue general has compromised the system. This analogy holds well for nodes in a distributed network and helps envision strategies to create robust fault tolerance.

Let us considers the scenario of three generals A, B, and C, and their exchange of messages. If we suppose that A is malicious and sends conflicting messages to both B and C, then it is impossible for B and C to determine who the traitor is given that they can communicate with each other. One conclusion to draw from this is that the number of nodes that are faulty must be less than a third of the total number of nodes. In other words, the strength of an algorithm that can deal with Byzantine Faults is measured in terms of the number of faulty nodes or processes it can tolerate.

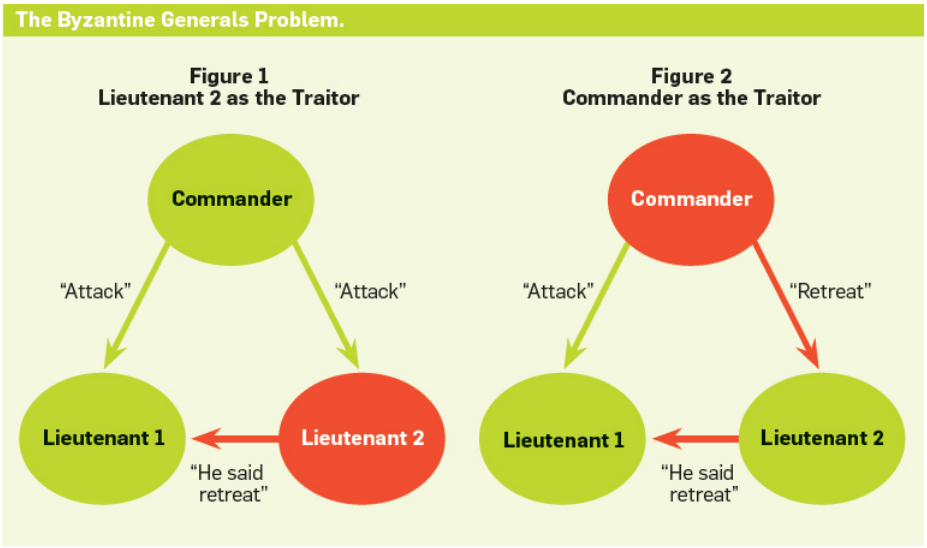

Figure 2: Byzantine Fault Tolerance

There are a number of implementations of Byzantine Fault Tolerance in practice today. Proof-of-work is a very well established implementation of Byzantine Fault Tolerance used in public blockchains like Ethereum and Bitcoin. A derivative of the network consensus algorithm Raft, hardened with fault tolerance, has been designed by the JP Morgan team which created the permissioned blockchain called Juno. The Ripple consensus algorithm is another implementation of Byzantine Fault Tolerance in the context of permissioned blockchains used most notably by the Intel Sawtooth blockchain.

1.3.2 Public Key cryptography

Public key or asymmetric cryptography is a sub-field of information theory that underpins much of modern day computer security. The mathematical underpinning of public key cryptography lies in families of mathematical functions that are injective and irreversible. A function that is injective or one-to-one, maps an input to a unique output. So functions of this nature will never have multiple inputs pointing to one output or vice versa. This is a prerequisite for functions used in cryptography. The way public key cryptography works is that a key pair containing two keys, private and public, are generated for an account. The public key for that account is used to encrypt transactions transferring tokens into the account. The private key is used to decrypt the transactions and allows the owner of the account to further spend the tokens. This is how the bitcoin blockchain works but there can be differences depending on the function family or implementation. The other requirement of the mathematical function is called reversibility.

Mathematical functions are called irreversible when they are easy to calculate

in one direction but practically impossible to calculate in the other direction. For instance the cryptographic algorithm called RSA necessitates the multiplication of two large primes, which is easy and straightforward to do. Getting those two primes from the product, on the other hand, is extremely difficult since it requires factoring a very large number. RSA is implemented in the SSL protocol that is very commonly used in networking. Elliptical curve cryptography uses a different class of functions that are less computationally intensive than those used in RSA. It has been getting attention from computer security experts as a possible replacement for RSA. Eleptical curve cryptography is the implementation of choice in Bitcoin.

1.4 Consensus

The state of a blockchain network is defined as the set of all the accounts that it contains. Network consensus mechanisms determine the state transition rules that determine changes to the state of the network. Blockchain networks are prone to Byzantine Faults and originally state transition rules were a part of the consensus mechanism. Ethereum is a notable exception to this where the state transition rules are entirely separate from the consensus mechanism.

There are three major categories of network consensus that need to be appropriately met by any network algorithm if it is designed to be implemented as a consensus mechanism in a blockchain: correctness, agreement, and Utility.

1.4.1 Correctness

Correctness refers to the ability of a distributed system to ascertain whether transactions carried out are correct or fraudulent. In the case traditional financial institutions, a fiduciary network of institutions that guarantee trust and the correctness of the transaction is established through digital signatures. In the context of a distributed system, there is no institution or a group of institutions to guarantee trust. In fact, in a number of blockchain implementations some or all the members may be anonymous. In order to establish correctness,

1.4.2 Agreement

A distributed system has a number of nodes which contribute to maintaining the state. In other words, the distributed system must have a shared or global truth that all nodes can agree on. While this concept is related to the idea of correctness, agreement refers to the ability of the distributed system to protect against the same correct transaction made multiple times. This is also referred to as the problem of double spending. Double spending was such a big problem that it essentially crippled the earliest digital currency ventures. It was the bitcoin blockchain that solved what was fundamentally a problem of agreement. Agreement is the existence of a single global set of recognized transactions in the system.

1.4.3 Utility

A distributed system with the properties of correctness and utility will still be unusable if the latency of the network is very high. Even with robust correctness and utility, if the settlement time is a week, the distributed system cannot be used because of it's impracticability. While utility has other components, latency is the most quantitative measure.

1.5 Summary

The blockchain is made up of different technologies derived from different fields of computer science. Consensus mechanisms are derived from computer networking. Cryptographic security is derived from information theory. The wallet, or where the user interacts with his tokens, lives on different devices including smartphones. In some paradigms it can be useful to think of the blockchain as a protocol that can be implemented over existing networking infrastructure, the internet being the most important. The blockchain industry must work towards standardization of internal components. This standardization will involve large established firms, like IBM, creating consensus mechanism standards, a smart contract core upon which derived contracts will be written, database implementation on which the ledger is built, and cryptographic security.

2 Smart Contracts

Contracts are essentially agreements between two parties that contain precise rules about what actions are to be undertaken by whom and under which circumstances. To establish trust between such two parties, a number of institutions facilitate transactions. Most financial industries are heavily dependent on such 'middle men'. The blockchain provides a trustless mechanism that can automate contracts. Automation of contracts would include immutability with respect to the contract rules. Once the contract rules have been agreed upon, neither party can change rules or renege on them. It also includes using third party sources to ascertain if conditions dependent on external factors have been met. Making middle men obsolete also carries with it the added advantage of reduced settlement times as well as reduced operational costs.

Smart contracts have applications in a large number of use cases beyond the conventional currency based transactional model that the blockchain is most well known for:

- Public records such as land holdings, vehicle registrations, and various licenses.
- Identification such as drivers licenses, passports, and identity cards.
- Private records such as bets, contracts, loans, and wills.
- Intangible assets such as patents, trademarks, and copyrights.

- Financial transactions such as private equity, mutual funds, and derivatives.

3 The Indian Context

3.1 Financial Inclusion

The last couple of years has seen the government initiate massive schemes like the Jan Dhan Yojna to provide the large unbanked population in India with financial services. According to a recent report by PriceWaterCoopers released last year, the unbanked population has halved to 233 million in 2015 from 557 million in 2011. While the numbers seem impressive, the fact that about half these added accounts have a balance of zero, makes this endeavor by the government far less successful. According to government sources only about 27% of villages have banking services readily accessible. The success of mPesa in Kenya and bKash in Bangladesh could point to a cost effective and efficient method to extend financial inclusion to all segments of Indian society. The possibility of implementing mobile banking solutions necessitates a more thorough analysis of which underlying technology to implement. Blockchain technology may be the ideal solution to build mobile banking infrastructure on.

The RBI has been making structural changes to the architecture of the banking industry. With the advent of payment banks in addition to universal banks, the RBI has introduced non-traditional banking institutions and might yet add more. All of these banking institutions will have independent mechanisms of database management and settlement. Even though the current mood in the financial world with respect to the application of Blockchain technology to the banking sector lies in permissioned or private blockchains, inter-bank settlement and ledger maintenance could one day be carried out on the blockchain. Even intra-bank settlement can benefit greatly from the distributed ledger architecture of the blockchain.

The scenarios outlined above make a compelling case for why distributed ledger technology, or the blockchain, is extremely relevant to the financial industry in India today. Any attempt to service the massive underbanked population via expansion of infrastructure would be prohibitively expensive. By using the blockchain as the core protocol to carry out mobile banking, banks in India could target these populations relatively easily. This is because Blockchain technology has significantly lower cost of deployment, an immutable shared ledger, and robust security mechanisms. By reworking the service model followed by all banks today from the ground up, banking facilities could be served on mobile devices rather than physical locations resulting in massively reduced infrastructure costs. While financial institutions balk at the term 'shared ledger', in the permissioned blockchain paradigm, shared can be defined to only include the parties involved. The extent of how 'shared' the ledger is can be controlled depending on the use case. For instance the parties involved in a transaction, the payment service provider, and the regulatory bodies could all have

access to a shared ledger without any other individual or institution having access. And finally blockchain ledgers are built upon technology derived from cryptographic breakthroughs in theoretical Computer Science over the past few decades. Transactions between two parties are signed by digital signatures using cryptographic techniques and the transmission of these transactions is further encrypted ensuring robust digital security.

3.2 UPI

The NPCI(National Payment Corporation of India) was formed in 2009 by the RBI to address certain concerns it had with respect to banking infrastructure and public accessibility to it. The NPCI most recently launched the Unified Payments Interface making explicit three major design principles. The first is adoption. This ties in directly with the preceding section on financial inclusion. Innovative technology is meaningless if it is inaccessible to the masses or inconvenient. A major hurdle to accessibility of banking technology can hypothetically be overcome by implementing mobile banking infrastructure that can be served on phones. The second is security. Security at all levels of implemented technologies is paramount to the banking industry. Racked by malicious attacks, the need to protect transactional data is of utmost importance. The third is cost. With mobile technology so widespread, investing in mobile banking technology made infinitely more sense than in creating traditional banking infrastructure. More over operational costs of banking at large scale are significant and innovation to drive down those costs can result in sizable savings.

How is the blockchain relevant in this scenario? Firstly, adoption of new technologies in regulated financial industries is always difficult. Innovating with technology that the banks use to serve it's customers becomes much easier with the creation of the UPI. In this context, blockchain technology can be implemented to conduct mobile banking transaction without regulatory hurdles. In other words, regulators in the banking industry are looking to evolve mobile banking and this provides the perfect opportunity to innovate with newer technologies like the blockchain. Secondly, date security is a paramount concern for all stakeholders in the banking ecosystem. The blockhain was designed as an open or public network and the reason it continues to thrive(bitcoin) is because of robust data security mechanisms. Running mobile banking infrastructure on the blockchain would give Indian banks the ability to secure transactional data better than traditional network security implementations. And finally, operational costs are greatly reduced if authentication, verification, and settlement all happen in a distributed system rather than a centralized one. It's not surprising that both NPCI and RBI have constituted committees to understand the blockchain better. However the specifics of how the blockchain gets implemented are far less concrete at this point.

4 Private vs Public

4.1 Bitcoin

Private or permissioned blockchains share significant similarities with bitcoin. The state of the system is immutable in both the bitcoin blockchain and private blockchains. The general category of consensus mechanism is a BFT(Byzantine Fault Tolerance) algorithm. The specific algorithm, however, is almost guaranteed to be different because the consensus mechanism of proof-of-work, used in bitcoin, has limited usefulness in the private blockchain paradigm. Proof-of-work is a consensus mechanism where large processor farms carry out repetitive calculations to win a numeric lottery(the winning numbers are called the 'difficulty'). The miner who wins the lottery is allowed to pay himself a fee and add his block of transactions to the blockchain. The reason this guarantees BFT(Byzantine Fault Tolerance) is because if transactions are edited within the blockchain, the network's hashing rate(which determines mining power) is going to be much higher than that of the individual miner or mining pool and therefore the 'honest' transactions are going to be on the longest chain of blocks. Admittedly this is not straightforward but takeaway from this recalls what was discussed in Section 1.3.1, that a significant fraction of the nodes(miners) must be dishonest to introduce 'wrong' transactions in the blockchain(51% for bitcoin). In the context of private blockchains, this entire process is unnecessary because the nodes that do the 'mining'(adding transactions to a block) are all known. This is why private is used interchangeable with permissioned in the context of the blockchain. Even though the nodes which add transactions to blocks are known/permissioned, the concerns discussed in Section 1.3.1 still persist. Robust consensus mechanisms that are Byzantine Fault Tolerant must still be implemented. The good news with doing away with proof-of-work is that the utility(latency) of the system is drastically improved.

Smart contracts are far more powerful in private blockchain implementations. The bitcoin scripting language is intentionally primitive(eg. Turing-incomplete) and the inbuilt data frame, upon which a script is run, has very rigid categories. Mostly about quantities of bitcoin and the spending conditions. While some people have devised ways to encode data in some sub-fields of the data frame, the solutions are very inefficient and inelegant. Private blockchains on the other hand can have sophisticated scripting capabilities built in at different levels of the blockchain. Smart contracts defining transaction rules can be written at the level of the wallet implementation. While encoding business logic by itself is a significant improvement over the bitcoin smart contract abilities, the time stamping for the execution of the smart contract can be delinked from the timestamping at the moment of block creation. So in the case of bitcoin, a smart contract is dependent on when the block is mined for execution whereas in a private blockchain the smart contract can be encoded with execution instructions including when it must/can be executed.

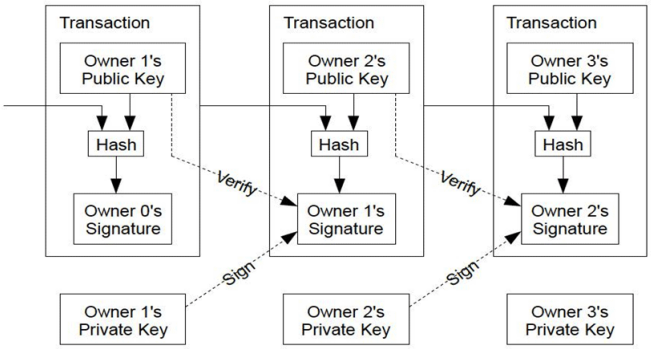

Figure 3: Bitcoin blockchain

4.2 Ethereum

Ethereum was designed as a generalized blockchain in contrast to bitcoin. It is built around the idea of smart contracts and ether, which is the computational currency that determines how long the smart contracts run. Ethereum is a very technically advanced implementation of the blockchain and the organization is run by very good developers who are not afraid to experiment. While this mentality is very conducive for technical experimentation, there are significant security challenges that arise from how the ethereum ecosystem is set up. For instance developers who wish to build their applications on ethereum are often less conscientious about the quality of their code and can introduce vulnerabilities in their implementations. Most notably the DAO hack in recent times damaged the reputation of ethereum significantly. The DOA was shut down and the reimbursement process was extremely messy.

While private implementations are possible, the ethereum blockchain is a public one. Unsurprisingly the ethereum blockchain uses the proof-of-work consensus mechanism. This is in contrast with private blockchains which typically don't use proof-of-work.

5 Conclusion

The blockchain, on closer scrutiny, reveals itself to be a collection of different technologies that are combined together to facilitate transactions. The database or ledger records all transaction and more fundamentally is the store of the state of the entire system. Network consensus mechanisms allow state transition rules to be implemented allowing transactions to be written on the ledger. There are a number of choices for the structure of the ledger, kind of consensus mechanism, and even the distribution of the network. The kind of blockchain, private or public, often makes certain design choices more useful than others. The use-case

of the blockchain also affects what technologies are chosen. And sometimes even ideological persuasions can directly affect the technology stack in question.

From this still muddled picture emerges the conventional idea that a technology, or a technology stack, is only as useful as the problem it solves. So the focus is necessarily on identifying the problems that need a solution. And from this should follow the design choices that make up different blockchains.

References

[1] Nakamoto, Satoshi. Bitcoin: A Peer-to-Peer Electronic Cash System: Cryptography Mailing List, 2008.

[2] Swan, Melanie. Blockchain: Blueprint For A New Economy: Sebastopol, O'Reilly, 2015.

[3] Annexure T to RFP No. NPCI/RFP/2015-16/IT/001. Unified Payment Interface: API and Technology Specifications: 2015.

[4] Copeland, Christopher and Zhong, Hongxia. Tangaroa: a Byzantine Fault Tolerant Raft: Stanford, 2015.

[5] Castro, Miguel and Liskov, Barbara. Practical Byzantine Fault Tolerance: Proceedings of the Third Symposium on Operating Systems Design and Implementation, New Orleans, USA, 1999.

[6] Buterin, Vitalik. Ethereum: Platform Review, Opportunities and Challenges for Private and Consortium Blockchains. 2016.

[7] Brown, Richard Gendal , and James Carlyle, Ian Grigg, Mike Hearn. Corda: An Introduction. 2016.

[8] Figure 1: Networks visualized. http://images.flatworldknowledge.com/lule/lule-fig11_002.jpg

[9] Figure 2: Byzantine Fault Tolerance. http://deliveryimages.acm.org/10.1145/1540000/1538794/figs/uf1.jpg

[10] Figure 3: Bitcoin blockchain. Nakamoto, Satoshi. Bitcoin: A Peer-to-Peer Electronic Cash System: Cryptography Mailing List, 2008.