2018

Blockchain: A New Type of Database

Brent Marshall
Parkland College

2018

Blockchain

A NEW TYPE OF DATABASE

BRENT MARSHALL

CSC 159
Mark Schudel
Spring 2018

Introduction

The blockchain is a new technology that seems to have people all over the world talking about it. But what is it? And what can it do? This paper will explore both the current uses of block chain technology and its potential uses. We will begin with definitions, followed by the history and uses of blockchains.

The basics

Let's begin by discussing the basic technology behind the blockchain and how it functions.

Hashes are one way encryptions that produce an output of a set length. Once something is hashed, you can never get the original data back but with hashes the same input always gives the same result. Often times this is used to verify a file's integrity. An important characteristic is that hashes are difficult to predict. You have to run the hash algorithm in order to see what the hash will be.

If you take the word "hello" and run it through a hashing algorithm, in this case SHA256 (a type of hashing algorithm) you will get the following result every single time.

2CF24DBA5FB0A30E26E83B2AC5B9E29E1B161E5C1FA7425E73043362938B9824

The hash will always be the same length (64 characters for SHA256) no matter what input is ran through it.

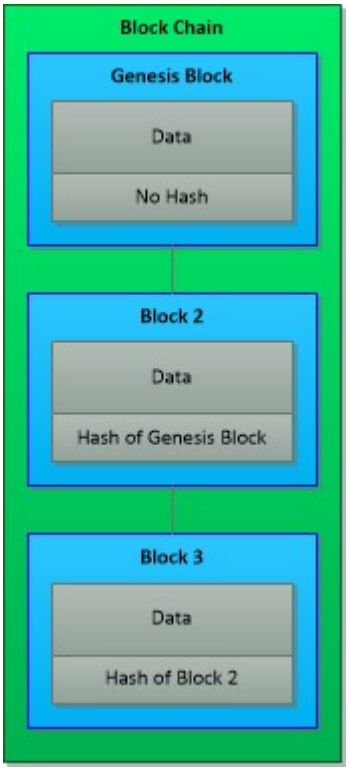
Input	SHA256 Output.
hello	2CF24DBA5FB0A30E26E83B2AC5B9E29E1B161E5C1FA7425E73043362938B9824
Hello	185F8DB32271FE25F561A6FC938B2E264306EC304EDA518007D1764826381969
Hello!	334D016F755CD6DC58C53A86E183882F8EC14F52FB05345887C8A5EDD42C87B7
He llo!	C183B0C10561C265A21997103E56984E93EDDC252EFFBDF9D26B871D06BB27A9

Because hashes change so much with even the slightest alteration to their input, they are commonly used to verify that something has not been modified. Let's say Bob has written a paper for a school project and wants to make sure it doesn't get changed. So Bob runs a hashing algorithm on his

paper and gets a hash that he takes with him. Later, his friend Tom comes by and decides to play a joke and changes something in Bob's paper. Later on Bob comes home and runs the hashing algorithm on his paper and discovers that the new hash is different. Bob now knows something has changed in his paper and promptly goes looking for Tom.

Hashes have a set length and it is possible, though very unlikely, to have two inputs that generate the same hash. This is referred to as a hash collision. The MD5 hashing algorithm was compromised in this way in 2008 when a team of hackers connected 200 Sony Playstation 3s together and demonstrated that they could create collisions at will. Newer hashing algorithms such as the SHA256 are much more robust and resistant to these kind of attacks.

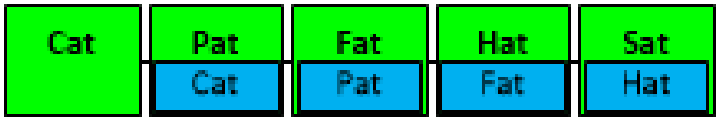
Understanding hashing algorithms is key to understanding what a blockchain is. A blockchain is a series of blocks that are linked together in a sequence using hashes. In each of these blocks you can store information that can be retrieved later just like in a database. A blockchain starts with a genesis block and each subsequent block must possess a hash of the previous block.



In the diagram on the right we have a simple blockchain. At the top is the genesis block whiich is the first block in the chain. The genesis block, being the first block in the chain can have data stored with it but will not have any hashes of a previous block since there aren't any previous blocks. Block 2 will also have data and a hash of the genesis block since the genesis block came before it. And lastly block 3 will have data and a hash of block 2. Block 3 does not have a hash of the genesis block; instead, it has a hash of the entire second block including the hash of the genesis block.

This unbroken chain of blocks possesses a unique property and that is called immutability. It gains this property because once the next block is created the previous block cannot be changed without breaking the blockchain and causing a snowball effect that will invalidate every hash in the chain.

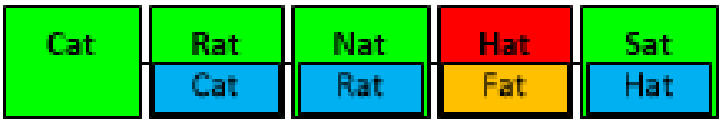
Let's see this in action. The following Block chain has 5 blocks, each storing data with the 4 blocks after the genesis block containing the simplified "hash" of the previous block.



Each block contains a stored hash (in blue) of the previous block. Block 2 contains a stored hash of block 1 (Cat), block 3 contains a stored hash of block two (Pat) and so on. Now let's go ahead and try

to change block 2.

We have changed block 2 but now the hash of block 2 is now “Rat”, and the stored hash in block 3(Pat) is invalid because they don’t match. So now in order to get things to be valid we need to change the hash in block 3(Pat) to match block 2(Rat).



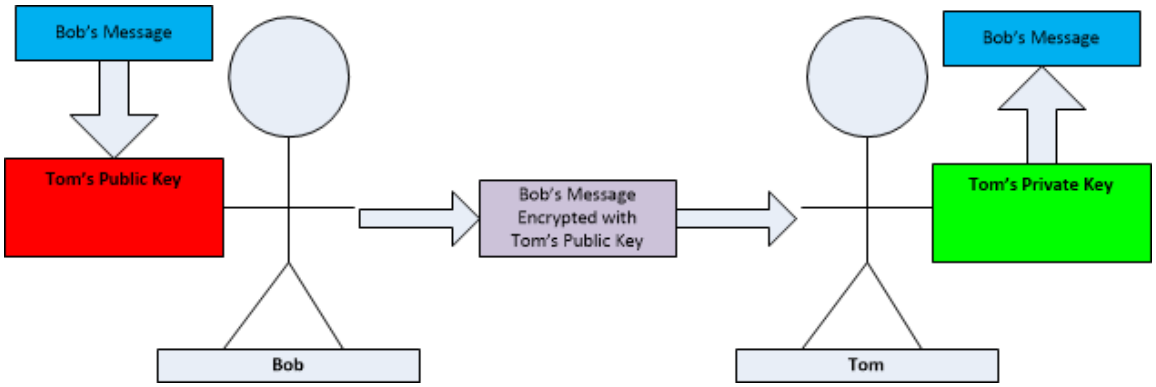
Now the stored hash in block 3(Rat) has been changed to match block 2. Since the stored hash itself is a part of the block data, the hash of block 3 has been changed from “Fat” to “Nat”. Since the stored hash in block 4 is “Fat” the blockchain is broken once again. Now we need to change the stored hash in block 4(Fat) to match the hash of block 3(Nat).

Now the stored hash in block 5(Hat) no longer matches the hash of block 4(Bat) making it invalid. So this process needs to continue all the way till the end of the chain in order to make it valid.

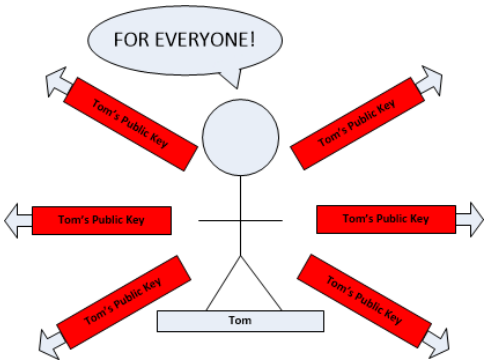
This is what is meant by immutability. When data is added to the blockchain it cannot be changed or edited later on. If a person tries to change something then they have to rebuild every single block after.

Another important technology to understand is Public/Private key cryptography. This is a type of cryptography that utilizes an asymmetric encryption. This is widely used on the internet in order to secure connections between two points so that people cannot read the traffic.

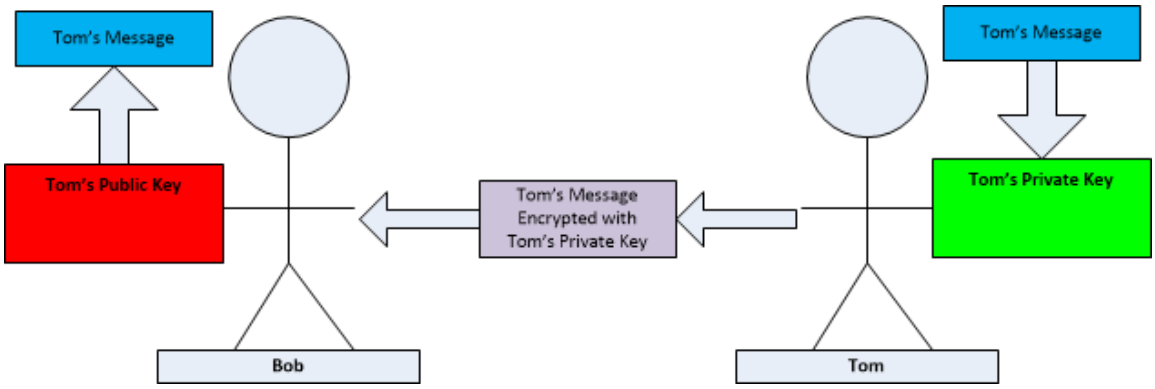
In a symmetric encryption there is only one key. That single key is used to both encrypt and decrypt messages. In an asymmetric encryption there are two keys. One key will lock what the other key will open and vice versa. These keys are referred to as private keys and public keys. If you encrypt a file with a private key then only the public key will be able to decrypt it and if a file is encrypted with a public key then only the private key can decrypt it. This has a number of uses, which we will show below.



If Bob uses Tom's public key to encrypt a file he can sure that only Tom can read the message even if everyone else also has Tom's public key. A message encrypted with a public key cannot be decrypted by anything other than the private Key.



This means that Tom can give his public key to everyone he wants to without worrying about his messages being leaked.



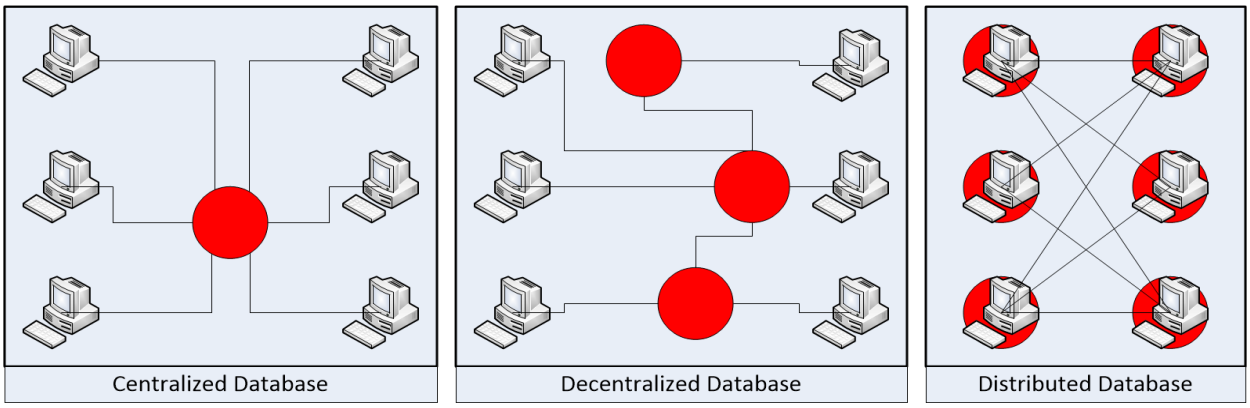
If Tom encrypts a message with his private key Bob can read the message and be certain that

Tom was the one who sent this message. If the message was sent by someone else besides Tom, then Tom's public key wouldn't be able to unlock it.

Private/public key cryptography is used on the blockchain for the transfer of units of cryptocurrency from one user to another. By signing a message to send a certain amount of currency with your private key and your recipient's public key you can both ensure that only they will get the message and that you actually sent it. The transaction can be placed on the blockchain still encrypted with you private key. Since only your public key can decrypt the message, it serves as proof that you sent the message to the recipient. This also allows the blockchain can decrypt the message with your key and then verify that you actually have the funds.

All of these transactions happen out in the open where everyone can see them. However the only data known about any user is the public key. So if you keep the ownership of your public key to yourself then you can maintain your privacy. The software that contains your private key is called a wallet and there are many of these available online and many of them support multiple cryptocurrencies.

The most exciting property of the block chain is the ability to keep this immutable database distributed and be trusted even if the individuals holding the ledgers cannot. Let's explore the different properties of each of these database types.



With a centralized database you have a single point of failure and that is the database itself. It can be attacked or compromised. A hacker could break in and steal your data or utilize a denial of service attack denying you access to your data. Downtime due to maintenance and upgrades can keep you from your data when you need it. A natural disaster could occur destroying the database and your information with it. You also have to trust the people who control the database. People can be dishonest, deceived or threatened. An employee at the database can steal your data. An employee can also be tricked into giving your data away. The employees can also be threaten or coerced into giving up your data by either criminal and/or government entities.

With a decentralized database you have more than one point of failure. This makes it more robust and resistant to attacks and disasters, but you still must rely on the trusting the people who

control the database. So the human vulnerability still exists.

With a distributed database, every user has a local copy of the full database. If something happens to your copy then you can pull the most current version from your peers. It is easy to see how this can protect against attacks or disasters but what about the human problems? With a distributed database there is no central authority. No company, person, or institution controls the database. The protocols and algorithms of the blockchain itself are the authority. The blockchain running on many computers at once is the central authority of itself.

Since every user possesses a copy of the blockchain database one question may be posed “How do you determine which copy of the database is the correct one. The answer to that is simply which blockchain is the longest. The blockchain has a system in place to ensure that the longest blockchain is correct one. This system is referred to as Proof-of-Work.

Proof-of-Work is a system that causes the creation of new blocks to be a difficult and lengthy process. When a transaction is ready to be added to the blockchain it is broadcast to all of the peers. Each of these peers gather up a number of these new transactions and get to work creating a new block. Which and how many transactions each peer decides to add their block is determined by when they arrive and what settings the peer has. These peers cannot just run the block through a hashing algorithm and be done. This is because the hash created has to conform to some rules.

Inside each of each block is a value called a nonce. This value is incremented each time the block is hashed, generating a different hash each time. The goal is to continue hashing until a certain number of leading zeros (or similar goal) is achieved. This is a difficult task to accomplish, but very easy to check. It may take millions of hashes to find the correct nonce value, but only hash one to verify.

We will use the following code in PowerShell to demonstrate this. In this example we will only hash the nonce and keep our comparison in hexadecimal.

```
$hashLead = "0"

$SHA256 = new-object -TypeName System.Security.Cryptography.SHA256CryptoServiceProvider
$utf8 = new-object -TypeName System.Text.UTF8Encoding
$nonce = 0
$continue = $true

Measure-Command{
    while($continue){
        $hash = ([System.BitConverter]::ToString($SHA256.ComputeHash($utf8.GetBytes($nonce)))) -
replace "-"
        Write-host "Nonce:$nonce Hash:$hash"
        if(($hash.substring(0,$hashLead.Length) -like $hashLead)){
            $continue = $false
        }else{
            $nonce++
        }
    }
}
```

First we will quickly explain this code. `$hashLead` is the variable that we will use to set the rules for our hash. This is done by adding zeros in between the quotation marks (Such as"0", "00", "000", etc.). The line beginning with `$SHA256` and `$utf8` is simply setting up the tools to do the hashing. The line starting with `$nonce` is the value we will be hashing, it will start it at 0 and be increased after each hash. The line starting with `$continue` is set to `$true` and is used to keep our loop going until it is set to `$false` which will happen when the hash meets the criteria. The `Measure-Command` applet is used to measure and display the amount of time required to run the loop until it gets the hash required.

The main body is a loop. Each time through this loop the body will hash the `$nonce`. It will write out the current value of the `$nonce` and it's `$hash`. Then it will then test to see if the hash has the required number of leading 0s. If it doesn't it will add 1 to the `$nonce` and try again. If it does then it will change `$continue` to `$false` causing the loop to exit followed by `Measure-Command` reporting the time.

Each time we run the code we will increase the number of zeros in the `$hashLead` variable.

Zeros*	Nonce	Time	Hash
0	39	64 milliseconds	0B918943DF0962BC7A1824C0555A389347B4FEBDC7CF9D1254406D80CE44E3F9
00	286	286 milliseconds	00328CE57BBC14B33BD6695BC8EB32CDF2FB5F3A7D89EC14A42825E15D39DF60
000	886	843 milliseconds	000F21AC06ACEB9CDD0575E82D0D85FC39BED0A7A1D71970BA1641666A44F530
0000	88484	79705 milliseconds (1M 19s 705milli)	0000A456E7B5A5EB059E721FB431436883143101275C4077F83FE70298F5623D
00000	596138	531872 milliseconds (8M 51s 872milli)	00000691457F4F0CE13E187B9AB4FDA6D42C8647752909B8F71F9DBD8F6BD4AB

*Hexadecimal

As we can see each increase of the required leading zeros drastically increases the difficulty of the proof-of-work.

In that example we only hashed the nonce. In the actual block chain the entire block along with the nonce will be included in this calculation. Since each peer decides which transactions to include in their block, each block will be different and this effectively means that the calculation time is randomized. One person could be lucky and happen to have transaction data that causes their hash to get the required number of leading zeros the first try. Another person could get unlucky and get a set of transaction data that would require years to get the required leading zeros. In practice this means that each time you run a hash of a block you have a set percentage chance to get the hash you require and finish the block.

Below is the listed probabilities for 1 to 5 leading zeros (in Hexadecimal).

Leading Zeros	Percent
0	6.25
00	0.390625
000	0.0244140625
0000	0.00152587890625
00000	0.000095367431640625

Now as time progresses computing power gets better, this is why many proof-of-work systems have a means of increasing the difficulty. For example, in bitcoin's proof of work system a calculation is ran every 2016 blocks. This calculation gets the average amount of time it took to create each block. Bitcoin has a target time of ten minutes for a block. If the average took less than ten minutes then the difficulty is increased and if the average took longer than 10 minutes it will decrease the difficulty. This target time is not the same in every block chain. In Ethereum the target average is only 12 seconds.

Though this demonstrates the basics of proof-of-work, but not every block-chain uses this method of proof-of-work, bitcoin's implementation of proof-of-work is actually more complicated in order to allow more fine grained control of difficulty and storage efficiency. PrimeCoin doesn't solve for hashes but instead solves for new prime numbers. The goal of Prime Coin being to help in research. FileCoin doesn't use proof-of-work at all but instead uses proof-of-storage.

This proof-of-work combined with the immutability of the blockchain means that in order for a person to fraudulently modify the data in the blockchain they would need to possess more processing power than at least half of the peers calculating block combined. It is a race to calculate those blocks and the legitimate chain has a big head start and an even bigger engine.

The proof-of-work also has a second purpose. That is the creation of the cryptocurrency. The cryptocurrency in a blockchain is in a closed system. It cannot be added from the outside, though it can be lost. Whenever a new block is created a reward of coins is given by the system to the person who managed to create the block. This is what people refer to as mining.

Mining incentivizes people to try and create these blocks, which makes it even more difficult to fraudulently alter a block because a criminal will have even more people to compete against. Also each transaction also contains the address of the previous transaction the coin was a part of. This means you can track a coin all the way back to the block that created it. This is used to verify the validity of a transaction preventing forgeries and also increases the difficulty of fraudulently modifying a block. If an

attacker was actually possess enough processing power to modify a block and recalculate the chain, there would be no point. It would be simply more profitable for them to just mine.

The Next Step: A better record

Creating a cryptography based currency is an incredible feat but that is not the end. The various solutions created in this technology can be applied in numerous other applications. One important thing to realize is that you can store more than just transaction data within a block. The blockchain is a database and can store any type of data you wish. From property deeds and medical records to digital media and applications. This combined with the immutability aspect solves many of the problems in recordkeeping.

Factom is a company that is offering a blockchain based mortgage solution called Harmony. Leveraging the immutability aspect of the blockchain, the solution offers a rapid and accurate way of storing and retrieving mortgage related data. Paystubs, appraisal value, title insurance, and many other items are required to be kept along with the timing of those documentation. Storing that data is no small feat as this data can continuously grow for years as more paperwork is added. Eventually the documents are stored away and can sometimes be audited for legal purposes years later. The industry spends millions dealing with these huge amounts of paper. A blockchain storage solution would simplify a lot of this process.

De Beers announced earlier this year that they have a plan to build a blockchain based diamond registry. This registry is intended to track diamonds from the mine to the consumer. One purpose of this registry is to help combat the sale and distribution of diamonds from combat zones. Some countries such as the United States do not allow the sale of diamonds from these areas due to the possibility of them being used to fund terrorism. The use of a blockchain database would streamline the process of proving the legality of these diamonds making it cheaper and faster.

IBM is also working in a collaboration with several major retailers in the development of a blockchain based system to track and monitor food distribution. This has many benefits such as assisting in tracking down contaminated food. This is incredibly important because often times food contamination kills and speed is of upmost importance when trying to remedy the problem. This could potential save thousands of lives every year. Another benefit is that even competitors can use the same system while maintaining privacy due to public/private key cryptography.

The blockchain presents a new way to keep records that are safer, more secure, and more efficient than ever before. This by itself is a very useful but static data is not the only thing that can be stored in a block. A block is also capable of storing code. This allows for the creation of distributed apps and a thing called a smart contract. Smart contracts are automated agreements between individuals that is implemented in a code form.

Currently smart contracts are being researched to automate and streamline the transfers of various goods and services. The United States postal service has been exploring the use of smart contracts in their money transfer services. They also believe that smart contracts could be used in wills

or escrow services as well as ownership transfer. A well coded smart contract could cut out a lot of intermediaries who are required in order to transfer these things. With smart contracts A blockchain can not only store data in a private, secure, distributed, and immutable way but it can also act upon that data with others and keep a record of those interactions on its own.

Smart contracts however do have their own problems. A poorly coded smart contract can be just as hackable as any other program. This is where the immutability works to your disadvantage. You cannot patch what you cannot change.

The Ethereum Project is a cryptocurrency/smart contract system that is currently very popular. It is open source and the basis for multiple other blockchain projects. But aside from smart contracts Ethereum is working towards becoming a distributed application platform.

Distributed applications are software that run on multiple machine and work together to accomplish a singular goal. The most famous distributed application was the Folding@Home project. In 2000 the project was started at Stanford. The projects goal was to simulate protein folding for research and development of new drugs. Since then it has expanded to allow research into various other diseases such as Alzheimer's and Huntington's disease. Currently the Project has less than 90,000 contributors. Despite these low numbers the project has been incredibly useful and has contributed to 158 research papers. This is an example of the power of distributed computing. But the blockchain and the cryptocurrencies can provide both a means and a motivation to push this even further.

In fall of 2017 FileCoin raised $257 Million in funding for their block chain based Crowd based Cloud storage system. FileCoin wants to create a distributed file sharing system that will allow users to share the extra unused space on their hard drive. But FileCoin isn't just creating a framework to allow the distributed storage, it's making a system that will make people want to share their extra space. FileCoin is a Blockchain that rewards users with tokens for donating storage on their local Pcs. Earlier in this paper we discusses how completing the proof-of-work rewards the miner for creating new blocks, FileCoin will not use proof-of-work to reward coins but instead will use the shared hard drive space to reward users. This creates a lot of new possibilities.

The proof-of-work in a cryptocurrency does not need to be solving a hash. For example, in PrimeCoin the proof-of-work is calculating and finding new prime numbers for mathematical research. This opens a new set of possibilities as this means that you can incentivize people to share their extra computing power for all kinds of scientific research. Projects like Folding@Home have already accomplished great things. But it suffers from a lack of users. Imagine what it could do if it had a few million users.

Summary

The blockchain presents a new solution to many challenges that we have been struggling with for a long time. The blockchain has immutable records that are widely distributed. The blockchain can store just about anything inside those records including instructions. Lastly the blockchain requires a proof-of-work or similar that can be anything you want it to be. The power is in the simplicity and

modular nature of this technology. It is however still a young technology and promises many new innovations.

Ref.

https://hackaday.com/2008/12/30/25c3-hackers-completely-break-ssl-using-200-ps3s/

https://hackernoon.com/learn-blockchains-by-building-one-117428612f46

https://www.coindesk.com/257-million-filecoin-breaks-time-record-ico-funding/

https://github.com/ethereum/wiki/wiki/Sharding-FAQ

https://www.investopedia.com/terms/i/initial-coin-offering-ico.asp

https://www.investinblockchain.com/10-most-successful-icos/

http://folding.stanford.edu/

http://folding.stanford.edu/papers-results/

https://www.technologyreview.com/s/610454/the-us-military-wants-ai-to-dream-up-weird-new-helicopters/

https://www.factom.com/products/harmony

https://siliconangle.com/blog/2018/01/17/de-beers-launches-blockchain-pilot-track-diamond-authenticity

https://siliconangle.com/blog/2017/08/22/major-distributors-retailers-sign-ibm-blockchain-enhance-food-safety/

https://www.uspsoig.gov/sites/default/files/document-library-files/2016/RARC-WP-16-001.pdf

https://www.lynda.com/Data-Science-tutorials/Blockchain-Basics/574704-2.html

https://www.lynda.com/Blockchain-tutorials/Blockchain-Beyond-Basics/636127-2.html

https://bitcoin.org/bitcoin.pdf