[[1]](#footnote-1)

Blockchain Analysis (April 2019)

Yumei Bennett, *Graduate Student, Southern Methodist University*, Daniel Byrne, *Graduate Student, Southern Methodist University*, Benjamin Wilke, *Graduate Student, Southern Methodist University*, Brian Coari, *Graduate Student, Southern Methodist University*

*Abstract*— Blockchain technology is at the heart of bitcoin and is increasingly being developed for use in a broad range of financial applications. The goal of this project is to provide a survey and tutorial on Blockchain technology. The project and writeup will include an evaluation of an experimental Blockchain database.

*Index Terms*— Computer network management, Cooperative communication, Cryptographic protocols, Data transfer, Encryption, Distributed databases, Document handling, Fault tolerant control, Identity management systems, Information exchange, Internet, Network function virtualization, Network security, Peer-to-peer computing

# INTRODUCTION

B

lockchain is, at its heart, simply an immutable transaction log, often referred to as a [digital ledger][1], which employs cryptographic hashing techniques in a distributed fashion to create a one-way transaction register which memorializes the characteristics of an interactions between parties and secures those actions with completed proofs of work and a distributed, decentralized, and public log. It was originally designed as a means to facilitate commerce on the Internet without the need for a trusted intermediary such as a bank overseeing and injecting itself into transactions. Blockchain, due to its versatility and security, has been leveraged into a whole host of other applications for which trust, privacy, and security are necessities.

A Blockchain transaction starts with an owner of a digital coin wishing to spend that coin on something else. That owner's equity stake in digital coins is recorded as a transaction in a block secured with his public key, a hash, and the private key of the person he bought his equity stake from. When the owner wants to spend his coin, he transfers the coin by digitally signing a hash of the previous transaction with his private key and the public key of the next owner and adding these to the end of the coin. The payee can then verify the signatures to verify the chain of ownership. These one-way cryptographic hashes are computationally impractical to reverse, thus this interaction secures the ownership chain; however, it doesn't provide a means to prevent a bad actor from duplicating (double spending) or reversing a transaction.

The traditional means of ensuring the authenticity of a one way transactions is to use a trusted third party such as a bank or some governing body overseeing the transaction. While this works, it relies on the goodness of the intermediary and a measure of blind faith. Unfortunately, the intermediary institution might have other loyalties, rogue employees, or an entitlement mentality. Thus, this uncertainty manifests itself as an increase in the cost of performing a transaction. It also and has an inherent and an unfortunate but accepted level of grift and risk.

To get around the need for a central authority, the author(s) of Bitcoin, the first Blockchain technology, employ a proof of work concept in which nodes calculate a difficult secure hash from the previous block's hash, a nonce, and a collection of recent transactions. Every new transaction added to a chain of other transactions each of which are inextricably linked to the block that was added before it in the list. To modify a past block, an attacker would have to redo the proof-of-work of the block and all blocks after it and then catch up with and surpass the work of the honest nodes.[17]

This paper will explore the distributed ledger technology known as Blockchain. In this paper we will cover what Blockchain is, the theories and technologies behind Blockchain, its strengths and shortcomings, as well as its current and possible future uses. We will explain the different types of [Blockchains][3] (public, private, and federated), and the pros and cons of each. This paper will conclude with a description of an emerging application of Blockchain, [BigchainDB][16], which is a NoSQL database augmented with the core strengths of Blockchain, its decentralized nature and its immutability.

# Related Work

## Early Cryptography

Until about the 1970s, cryptography was mainly the purview of state actors. However, in the 70s, two publications brought cryptography into the public eye the US Government's DES, the Data Encryption Standard, and [Whitfield Diffie & Martin Hellmanand's publication on public key cryptography][6].

## DES

The Data Encryption Standard is a symmetric-key, block cipher algorithm for the electronic data encryption first developed at IBM, and then submitted as a candidate in a invitational study to the National Bureau of Standards, now the national Institute of Standards and Technology, following the agency's invitation to help define a government wide standard encryption algorithm for the protection of sensitive, but unclassified electronic government data.[7] IBM's initial submission was modified to use S-blocks and to decrease the number of bits used for the key as suggested by the National Security Agency before final ratification of the standard. The final design was a symmetric key, 56-bit, block encryption algorithm using S-block, or substitution blocks.[8]

Symmetric-key or shared-key algorithms like DES are cryptographic algorithms the use the same or very mildly transformed keys for both encryption decryption of data. The keys enable the possibility of a secure communication link between 2 or more parties. The requirement of this type of secure channel is that all parties to the communication need to know the secret keys. This fact represents the main drawback to symmetric key algorithms as the number of communicating parties increases it becomes exceedingly more complex to engineer a secure means of transferring the secret keys before the encrypted communication using the keys can begin.

Block ciphers, like DES, are cryptographic algorithms which deterministically encrypt fixed length groups of bits. In a typical block cipher implementation data is split into equal size blocks which are then transformed with a secret key seeding a mathematical transformation. However, since each block is deterministic and unwavering, block ciphers are susceptible to an exploit called differential cryptanalysis in which output blocks are scanned for evidence of non-random behavior caused by some repeating value in the plaintext, and that information is used to reverse engineer the encryption keys.

Substitution blocks, at the time a new feature in cryptography are a cryptographic algorithm in which an input of n bits are transformed into an output of m bits, and where n and m are not necessarily alike. DES incorporated the first widely used implementation of substitution blocks in its 6x4 s-blocks in which 6 input bits were transformed to 4 output bits. S-blocks serve to obfuscate the deterministic regularities arising from simple block ciphers thus making them more resistant to differential cryptanalysis.

## ### Early DES Opposition

During the DES public comment period, parts of the proposed standard were deemed classified, and thus could not be vetted by researchers in the field. Also, it was made known that the NIST enlisted the help of the NSA in modifying the initial IBM proposal, and that made people a little bit wary.

Due to the partially classified nature of the publication, the involvement of the NSA, and the introduction of heretofore unknown feature in the design called S-boxes, the initial specification was immediately met with wide skepticism in the academic realm. Whitefield Diffie and Martin Hellman, prominent cryptography researchers at Stanford University, panned the standard in an "Exhaustive Cryptanalysis of the NBS Data Encryption Standard" [11]. Primarily their concern was that the 56-bit key proposed by the standard was too short to be secure.

There was also public skepticism that the classified implementation of the S-boxes, now known as substitution boxes, in the design represented some sort of backdoor introduced by the NSA to allow it to decrypt secure communications. It wasn't until years later when S-boxes were rediscovered, that it was determined that the inclusion of S-boxes in the DES standard made it more secure against differential cryptanalysis.

However, the NSA did push for a reduction of the key size from the originally proposed 64bits to 48bits. IBM pushed back and the IBM and the NSA eventually compromised on a 56-bit key. However, this weakened the standard so much, according to Diffie and Hellman, that state actors and other organizations with deep pockets could theoretically build a computer to crack DES encryption through brute force. In Diffie and Hellman's early public shaming of DES they made this point in their opening introduction. [4]

Fittingly, in 1998, the Electronic Frontier Foundation built a $200,000 machine, well within the financial means of companies and especially governments, that cracked DES in a few days. The project was headed by John Gilmore, a founder of the Cypherpunk movement and a cryptographic academic and activist. The goal of the project was to demonstrate that DES was insecure, and that the US government had been telling deliberate lies about the security of DES for some time. [5]

## Public Key Cryptography

Diffie and Hellman and went on to advocate for an asymmetric encryption algorithm with greater complexity of the 56-bit DES and the ability to agree upon a shared key over an unsecure channel. This process became known as public key encryption and they went on to eventually patent this concept with Ralph C. Merkle in 1977. The Diffie-Hellman Key exchange protocol is what is can become to be named. [11,12]

Interestingly though, the concept was actually conceptualized and secretly described in a now unclassified paper by James H. Ellis, Clifford Cocks, and Malcolm J. Williamson of GCHQ, the British signals intelligence agency, 7 years earlier in 1969. [13]

The private key must remain secure, but the public key can be widely disseminated without fear that the encrypted data will be compromised. The security of this cryptographic system is hinged on the sheer complexity and time it would take to factor the product of the two very large prime keys with current technology.

## Cipher Punks

Blockchain technology grew out of the tenants of a group of high tech privacy enthusiasts calling themselves Cypherpunks. The term Cypherpunk is a compound word combining root words \*cypher\* or \*cipher\*, a coded message, and \*punk\*, in this instance referencing a member of a rebellious counterculture group advocating the widespread use of cryptography as a means to social and political change.

The Cypherpunk movement traces its roots to the early days of public key cryptography. With the initial academic uproar surrounding the distrust of the NSA's involvement in the manipulation of DES, and the secrecy in which the implementation of the algorithm was shrouded in, there began the seeds of dissent and the onus for the beginnings of the Cypherpunk movement. The eerily prescient early focus of the small, mostly online group was on discussing individual privacy in a digital world, government monitoring, and central authority control of information, all issues which are in national conversation today.

The Cypherpunk group started out in the as in informal meetup between tech minded individuals with scofflaw tendencies founded by Eric Hughes, Tim May, and John Gilmore. These three together with 20 of their friends began meeting regularly in the offices of John Gilmore's company, Cygnus, a stalwart of the burgeoning open source community, in the early 1990s.

Their meetup shortly morphed into a mailing list to broaden its appeal and to attract likeminded individuals from around the world.

One of their primary concern was that of governments and large powerful corporate entities capturing information, as John Gilmore described the problem in a speech at the first ACM conference on Computers in March 1991,

"In most of Europe, phone companies don't record the phone numbers when you call, and they don't show up on your bill. They only tick off the charges on a meter. Now, I was told that this is partly because the Nazis used the call records that they used to have, to track and identify the opposition after taking over those countries in World War II. They don't keep those records anymore." [9]

The mailing list's topics and tenor favored privacy in communications, self-revelation, financial privacy, anonymity, and pseudonyms. It also derided and actively opposed government data collection, forced self-revelation, and censorship. It is notable that an early and very active, member of the Cypherpunk mailing list was Julian Assange of Wikileaks. [15]

Privacy in communication was a primary concern of the movement, but equally important to the movement were discussions around financial privacy, as Eric Hughes, one of the founders, puts it in his “Cypherpunk Manifesto”,

"When I purchase a magazine at a store and hand cash to the clerk, there is no need to know who I am. When I ask my electronic mail provider to send and receive messages, my provider need not know to whom I am speaking or what I am saying or what others are saying to me; my provider only need know how to get the message there and how much I owe them in fees. When my identity is revealed by the underlying mechanism of the transaction, I have no privacy. I cannot here selectively reveal myself; I must always reveal myself." [10]

## Bitcoin

From this primordial stew of technologist, university researchers, cryptographers, a love of privacy in communication and in financial transactions, and a general mistrust of central authorities an anonymous, short (9 pages), but groundbreaking paper, was mailed to the Cypherpunk mailing list in 2008[4] by Satoshi Nakamoto, fittingly an alias, describing an electronic currency, Bitcoin, based upon cryptographic principals akin to asymmetric encryption, one-way functions, but in reverse. While Bitcoin was not the first digital currency created, it was the first to solve the problem of double spending and the first to do it without the need for a central authority regulating the transaction.

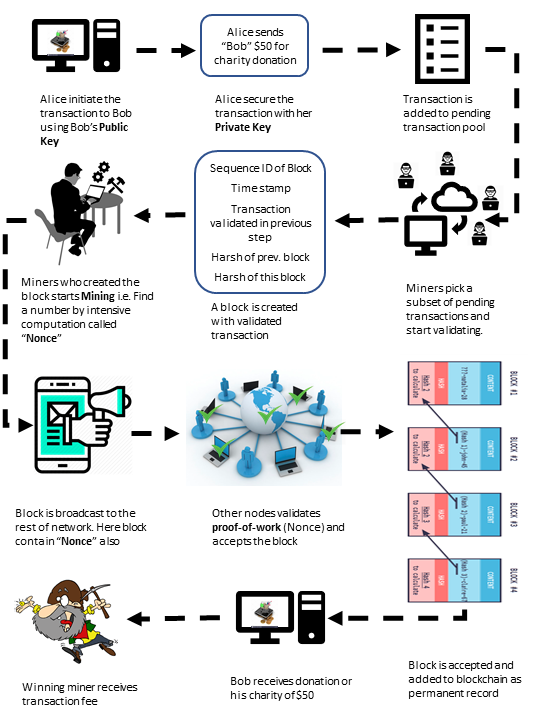
# Blockchain Technology

Blockchain technology is probably the best invention since the internet itself. It allows value exchange without the need for trust or for a central authority. For the first time, blockchain enable a way for one Internet user to transfer a unique piece of digital property to another Internet user, such that the transfer is guaranteed to be safe and secure, everyone knows that the transfer has taken place, and nobody can challenge the legitimacy of the transfer. The consequences of this breakthrough are hard to overstate.

Blockchain is a distributed database or a Ledger (as used in Cryptocurrencies) containing all the records linking the history to the very initial record, that can be used in various applications like Currency, Supply Chain, Identity Management, Assets Tracking etc. Distributed because it is not authored or controlled by a centralized entity like a Financial Institution (Banks) or Government, rather, each participating node has a copy of the most updated Ledger to prevent any tampering of data in the Ledger.

The Ledger is divided into Blocks of equal sizes and each block is connected to its previous block using the Cryptographic Hash of the Data in Previous Block, so if someone tries to change the value of a record in any Block, all the Blocks ahead of it are invalidated, thus providing an indication that the data has been tampered.

Here are the steps involved:



1. On a Blockchain, the transactions are happening between different nodes on the network, all the time. In our example, Alice initiate a transaction to Bob.

2. Alice’s wallet encrypt the content with Bob’s public key and signs the transaction with her private key.

3. The transaction will not be added to the block database (ledger) right away, it is added to pending transaction pool.

4. Special nodes, called Miners, pick up several transactions (size varies depending on application, up to 1MB in size for Bitcoin network).

5. Miners first validate all the transactions to see that all the transactions are fair and valid and does not contain any fraudulent transaction with problems like enough resources/fund in Alice wallet, duplicate transaction etc. (Validation rules are application specific.) These validation rules are called transactional consensus.

6. Once the transactions are validated, the miners create a Block of the transactions he picked. This Block contains a sequential ID, the Transactions picked by the miner, SHA256 Hash of all the values and Hash of the Previous Block.

7. The Blockchain Consensus requires the miners to resolve a computational-intense hard mathematical problem to sign the generated block before broadcasting it to the network.

8. The other miners validate the solution of the problem and if validated accepts the block.

9. Once over 50% of the Miners on the blockchain network accepts the block, the block is added to the ledger (block database) and distributed to all nodes on the network.

10. Bob finally sees $50 in his charity donation wallet.

11. The winning miner who created the accepted block is awarded with transaction fee.

# Blockchain Applications

In the year 2017, Blockchain became the second most popular search word on Gartner's website, and Distributed Ledger Technology (DLT) will continue to gain utmost significance across many industries.

Blockchain will help address several modern-day security concerns, including issues with contracts, identity, and fraud management. Blockchain-based lists will allow online retailers and financial organizations to conveniently vet their customers and fight against fraudulent activities.

The following are top 20 real-world blockchain applications across the industry according to CoinSwitch desk.

***Cybersecurity:***

*Guardtime* - a keyless signature which is currently used to secure the health records of one million Estonian citizens.

*REMME* - a decentralized authentication system which aims to replace logins and passwords with SSL certificates stored on a blockchain.

***Healthcare:***

*Gem* - working with the Centre for Disease Control(CDC) to put disease outbreak data onto a blockchain.

*Simply Vital Health* - ConnectingCare which tracks the health progress of patients, and Health Nexus, which aims to provide decentralized blockchain patient records.

***Banking and Financial:***

*Barclays* - Barclays has launched a number of technological initiatives which involves tracking financial transactions, combating fraud and compliance.

*Augur* - Allows the creation of blockchain-based predictions markets for the trading of derivatives and other financial instruments in a decentralized ecosystem.

***Manufacturing:***

*Provenance* - provide a blockchain-based provenance record of transparency within supply chains.

*TransActivgrid* - allowing members to locally produce and sell energy, with the goal of reducing costs involved in energy distribution.

***Government:***

*Dubai* - 30 government departments formed a committee dedicated to investigating opportunities across health records, shipping, business registration and preventing the spread of conflict diamonds.

*South Korea* - Samsung is creating blockchain solutions for the South Korean government which will be put into use in public safety and transport applications.

***Charity:***

*Bitgive* - aims at providing greater transparency to charity donations and reduce operation overhead.

***Retail:***

*OpenBazaar* - a decentralized market where no middleman is involved in trading of goods and services.

*Loyyal* - universal blockchain-based framework which aims at allowing customers to combine trade and loyalty reward points in newer ways so that retailers can offer more sophisticated loyalty packages.

***Real Estate:***

*Ubiquity* - a blockchain-driven system for tracking the complicated legal process which creates friction and expenses in real estate transfer.

***Transportation and Tourism:***

*Arcade City* - An application which aims at beating Uber at their own game by moving ride sharing and car hiring onto the blockchain.

*Webjet* - an online travel portal, which is developing a blockchain solution to allow stock of empty hotel rooms to be efficiently tracked and traded.

***Media:***

*Kodak* - developing a blockchain system for tracking intellectual property rights and payments to photographers.

*Ujo Music* - record and track royalties for musicians and allows them to create a record of ownership of their work.

***Smart Contracts:***

*Slock* - an Ethereum based internet-of-things (IoT) platform, which is using this application so that people can rent anything from bicycles to apartments by unlocking a smart lock after both parties agreed on the terms of the contract.

# Blockchain Database

Blockchain databases are different from traditional centralized databases. They are distributed in nature. For a blockchain database, each participant maintains, calculates and updates new entries into the database. Blockchain databases consist of several decentralized nodes. Each node participates in administration: all nodes verify new additions to the blockchain. For an addition to be made to the blockchain, the majorities of nodes must reach consensus. This consensus mechanism guarantees the security of the network, making it difficult to tamper with.

Blockchain databases are able to keep information that is relevant now, but also all the information that has come before. Blockchain technology can create databases that have histories of themselves. They grow like ever-expanding archives of their own history while also providing a real-time portrait.

The choice of database for blockchain platform or application is somewhat confusing. Some consider blockchain technology is another category of database like relational and document storage. The current blockchain application or platform are built on top of semi-custom database.

For example, Ethereum is using Leveldb, Leveldb was originally designed by Google to be a disk backed memory database. As long as most of it can fit into memory, it performs very well, but as more and more disk access is required, performance can degrade. Facebook's RocksDB tries to address some of these issues with multi-threaded background computation, but it’s still a fairly new thing and being in C makes portability hard. Parity is using RocksDB as it's database. The reason for choosing RocksDB is performance. Faster than LevelDB in their tests. Lately though they have hit some performance issues and a lot of corruption issues with RocksDB and are developing a proprietary DB to better suit Parity, the ParityDB.

So, what are the characteristics of a database that is suited for blockchain technology application development?

***Distributed:***

It is designed to be distributed and synchronized across networks, which makes it ideal for multi-organizational business networks. It also encourages organizations to come out from behind their firewalls and share data.

***Permanent Record:***

All confirmed transactions are stored as permanent record, you can’t just do whatever you want to the data. The types of transactions one can carry out are agreed between participants in advance and stored in the blockchain as “smart contracts,” which helps give confidence that everyone is playing by the rules.

***Majority Rules:***

Before one can execute a transaction, there must be agreement between all relevant parties that the transaction is valid. For example, if you’re registering the sale of a cow, that cow must belong to you or you won’t get agreement. This process is known as “consensus” and it helps keep inaccurate or potentially fraudulent transactions out of the database.

***Mutual Trust:***

Immutability of the data. Once you have agreed on a transaction and recorded it, it can never be changed. You can subsequently record another transaction about that asset to change its state, but you can never hide the original transaction. This gives the idea of provenance of assets, which means that for any asset you can tell where it is, where it’s been and what has happened throughout its life without any central authority.

Taken together, these four characteristics give organizations a high degree of trust in the data and the business network. That level of trust makes blockchain important for the next generation of business applications.

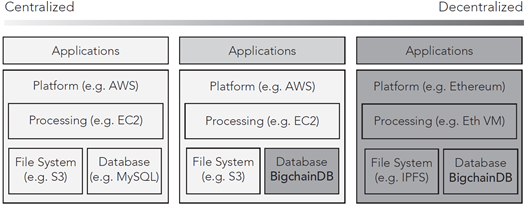
Next, we will explore one of the rising popularity database, BigChain DB who is adapting itself towards supporting blockchain applications.

# BigChain Database

February 10, 2016 — ascribe GmbH has announced the release of BigchainDB, a scalable blockchain database that can process one million writes per second and their first licensed user, Everledger.

BigchainDB achieves one million writes per second with sub-second latency and petabytes of capacity in a federated blockchain database. BigchainDB allows developers and enterprises to deploy blockchain applications and platforms at scale, formerly a limiting factor in getting sandbox proof-of-concepts into production systems. The throughput capacity of BigchainDB exceeds PayPal by 10,000 times, Visa by 20 times and the peak volume of Nasdaq by 10 times.

BigchainDB was a direct result of seeing scalability as a fundamental barrier to wider adoption of blockchain. BigchainDB is complementary to decentralized storage, processing and communication building blocks. It can be used side by side with higher-level decentralized computing platforms and applications, and protocols for identity, financial assets, intellectual property and sidechains. BigchainDB fills a gap in the decentralized stack.



As a blockchain database, BigchainDB is complementary to other decentralized systems, such as decentralized le storage (e.g. IPFS), decentralized data exchange protocols (e.g. Ocean Protocol), smart-contract blockchains (e.g. Ethereum or Hyperledger Fabric), and decentralized processing (e.g. TrueBit). BigchainDB works with centralized computing systems. From a base context of a centralized cloud computing ecosystem (left), BigchainDB can be added as another database to gain some decentralization benefits (middle). It also fits into a fully-decentralized technology stack (right).

BigchainDB is software that has blockchain properties (e.g. decentralization, immutability, owner-controlled assets) and database properties (e.g. high transaction rate, low latency, indexing & querying of structured data). BigchainDB version 2.0 makes significant improvements over previous versions. In particular, it is now Byzantine fault tolerant (BFT), so up to a third of the nodes can fail in any way, and the system will continue to agree on how to proceed. BigchainDB 2.0 is also production-ready for many use cases.



BigchainDB is designed for use cases requiring high-throughput, lower latency and more storage capacity such as streamlining clearing and settlements in the financial sector, tracking supply chains such as the tracking of diamonds, spare parts or consumer goods, or tracking records of provenance for physical or digital assets.

Diagram below illustrates the main components in a four-node Big-chainDB 2.0 network, and how they communicate with each other.



BigchainDB 2.0 uses Tendermint for all networking and consensus. Each node has its own local MongoDB database, and all communication between nodes is done using Tendermint protocols.

***Full Decentralization and Byzantine Fault Tolerance***

Any node can fail and the rest of the network will continue to operate. In fact, up to one third of the nodes can fail1 in arbitrary ways, and the rest of the network will continue to

work, i.e. the non-faulty nodes will agree on how to proceed.

***Immutability***

BigchainDB uses several strategies to achieve practical immutability. The simplest one is that there are no BigchainDB-provided APIs to change or erase stored data.

Another strategy is that every node has a full copy of all the data in a standalone MongoDB database (i.e. there is no global MongoDB database). Even if one node gets corrupted or destroyed, the other nodes won't be affected and will still have a copy of all the data.

Another strategy is that all transactions are cryptographically signed. After a transaction is stored, changing its contents will change the signature, which

can be detected (unless the public key is also changed, but that should also be detectable because every block of transactions is signed by the a node, and the public keys of all the nodes are all known).

***Owner-Controlled Assets***

Like most blockchains, BigchainDB has a concept of owner-controlled assets. Only the owner (or owners) of an asset can transfer that asset. (The owners are the holders of a particular set of private keys.) Not even a node operator can transfer an asset.

In most blockchains, there's only one built-in asset (e.g. Bitcoin or Ether), but BigchainDB allows external users to create as many assets as they need. However, it's worth noting that a user can't create assets that appear to be created by someone else.

***High Transaction Rate***

Despite its strong guarantees, Tendermint provides exceptional performance. In benchmarks of 64 nodes distributed across 7 data-centers on 5 continents, on commodity cloud instances, Tendermint consensus can process thousands of transactions per second, with

commit latencies on the order of one to two seconds. Notably, performance of well over a thousand transactions per second is maintained even in harsh adversarial conditions, with validators crashing or broadcasting maliciously crafted votes.

***Low Latency and Fast Finality***

Tendermint-based networks (such as BigchainDB networks) take only a few seconds (or less) for a transaction to be included in a new committed block. Once that happens, there's no way it can be reverted or considered defunct in the future, because Tendermint doesn't do forking.

***Indexing & Querying Structured Data***

Each node in a BigchainDB 2.0 network has its own local MongoDB database. That means that each node operator has access to the full power of MongoDB for indexing and querying the stored data (transactions, assets, metadata and

blocks, all of which are JSON strings). Each node operator is free to decide how much of that power they expose to external users. One node operator might decide to index geospatial data and offer optimized geospatial queries via a REST API, whereas another node operator might decide to offer a GraphQL API.

By default, BigchainDB 2.0 creates some MongoDB indexes and the Big-chainDB HTTP API includes some endpoints for doing basic queries.

***Sybil Tolerance***

Some blockchain networks (such as Bitcoin) allow anyone to add their node to the network. That brings the concern that someone could add so many nodes that they effectively control the network: a Sybil attack. Bitcoin makes Sybil attacks unlikely by making them prohibitively expensive. In a BigchainDB network, the governing organization behind the network controls the member list, so Sybil attacks are not an issue.

# BigChain DB Demo

# Blockchain outlook

Several analysts expect blockchain will first prove its value in supply chain management, where Cisco and IBM are both working on applying the technology.

Still, industry experts project on average that the technology will take 5.9 years to gain widespread adoption, according to a Cowen survey.

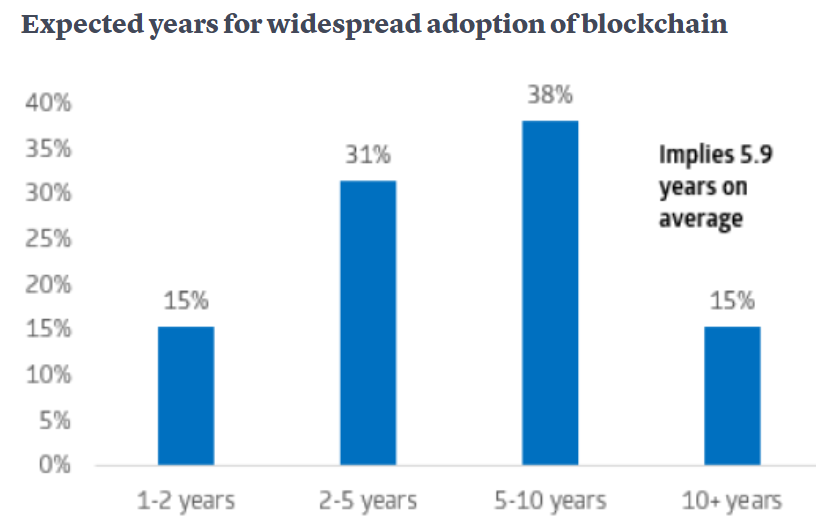
Many blockchain enthusiasts also say the technology will work best not as a new system within private enterprise, but in its pure form — an open-source, publicly accessible platform powered by a global network of users.

When it comes to proving the tangible benefits of the blockchain technology that many expect to change the world, it is still very early days and the earliest viable applications will likely be in supply chain management, rather than financial services, executives say.

In fact, Cisco stopped researching blockchain applications in financial services after 18 months because it will take a while for the many players in the complex markets to get up to speed, Anoop Nannra, head of the company's blockchain initiative, told CNBC in May. Instead, for the last year or so he said the team has been investigating blockchain applications in supply chain management.

Even still, their research indicates it will be roughly 10 years before blockchain really becomes prevalent in supply chain management. For financial services, Nannra predicts somewhere from five to 25 years.

"It's anyone's guess," Nannra said. "When you're trying to drive a mindset shift it's almost a generational thing."



<https://www.cnbc.com/2018/06/04/for-all-the-hype-blockchain-applications-are-still-years-even-decades-away.html>

<https://radiostud.io/eight-blockchain-platforms-comparison/>

<https://www.quora.com/What-kind-of-databases-are-used-by-Blockchain-platforms>

But if you are developing a blockchain-based application yourself, you'll probably need to provide support and maintenance for it. Research shows that the cost of support and maintenance for software is at least as great as the cost of developing it. This explains why software vendors charge so much for product support and maintenance. A good rule of thumb: If you develop an application that uses blockchains yourself, expect to pay at least as much throughout the application lifecycle as you do during development.

References

1. Judd Bagley (2018 , Sep . ). What is Blockchain Technology? A Step-by-Step Guide For Beginners. Blockgeeks. [Online]. Available: <https://blockgeeks.com/guides/what-is-blockchain-technology/>
2. Collin Thompson (2016 Oct.). How does the Blockchain Work? (Part 1). The Blockchain Review. [Online]. Available: <https://medium.com/blockchain-review/how-does-the-blockchain-work-for-dummies-explained-simply-9f94d386e093>
3. Collin Thompson (2016 Oct.). Blockchains & Distributed Ledger Technologies. BlockchainHub. [Online]. Available: <https://blockchainhub.net/blockchains-and-distributed-ledger-technologies-in-general/>

1. “This work was submitted as part of the MSDS 7349 Section 404 “Data and Network Security” course at Southern Methodist University as part of the Spring 2019 semester.

   Yumei Bennett (e-mail: yumei.bennett@oracle.com), Daniel Byrne (byrned@mail.smu.edu), Benjamin Wilke (e-mail: bwilke@mail.smu.edu), and Brian Coari (e-mail: bcoari@mail.smu.edu) are graduate students in the Data Science program at Southern Methodist University, PO Box 750100, Dallas, TX 75275 [↑](#footnote-ref-1)