VEE: Blockchain Database and Apps Platform

Sunny King
Kate Shan
Rob Zhang
Scott Nadal

Background

Occasionally we still remembered that, the birth of the Internet, a technology event arguably as important as the Industrial Revolution, or the advent of the Information Age, happened only less than 30 years ago. Walking on the streets of the Chinese cities, it seems that almost everyone is constantly checking on their phones, for something. That is how pervasive the Internet has become to our lives. Yet in the recent years we have already heard that Artificial Intelligence now leaves human champions in the dust in the Game of Go [Silver 2016], while making great strides in countless practical applications in our society. Not to mention, we were told, Quantum Supremacy, a messenger for the coming of the Quantum Age, is now looming in the immediate future. It almost seems that, the creation of Internet, just like computer, has already belonged to an ancient past. As a technologist, one cannot help but be amazed sometimes by the vast ingenuity of human civilization and the breakneck speed of progress it is capable of.

Satoshi Nakamoto's announcement of Bitcoin [Nakamoto 2008], in comparison, was not so eventful like those other breakthroughs. Instead, it's been constantly subject to tons of skepticism. It's a statement of how much ahead of his time Nakamoto was. Yet, his brilliant creation has gradually picked up steam. Now in 2017, as bitcoin price makes a run through US$10,000, blockchain technology is widely received as one of the major innovative technology field and disruptor. Yet, the impact of blockchain technology could still be underestimated. In fact, we would argue, that Bitcoin has introduced to the world what we have termed *strong private property*, which may have profound implications to the future of humanity. A brief review of the history of human civilization has always pinned private property as one of its foundational pillars. Thus, it could be argued that, the impact blockchain technology would bring to the world could be even greater than those of the Industrial Revolution. With these visions, we have termed this new economic era the *Virtual Economy Era*, and named the base currency in our platform *Vee*. To pay tribute to Satoshi Nakamoto's Bitcoin, we internally mark the calendar year 2009 as the year of V.E. 1.

One should however make a distinction between Bitcoin and Blockchain Technology. Bitcoin is the first practical solution to the Byzantine Generals Problem [Lamport 1982], a previously

difficult distributed consensus problem in computer science. Blockchain technology is derived from the underlying algorithm invented by Nakamoto to run Bitcoin network. It is now widely accepted to encompass all similar networks, public or private, which generally have some level of decentralization in their consensus model.

Bitcoin was designed as a virtual currency. What it stores in the Bitcoin blockchain is its ledger, the bookkeeping required to determine the ownership of the bitcoins. In 2011, Namecoin marked the first attempt at storing other type of information in such a blockchain. Later on there were many attempts at using the Bitcoin blockchain as a data store, for various application purposes. However, Bitcoin generally discourages such practice, as Bitcoin was not designed for such purposes thus making such utilization difficult and expensive.

However, blockchain technology in general, should be viewed as a distributed database system. This means, a significant portion of data in the world probably could be stored in such systems, just like the traditional relational databases and the fairly recent cloud databases. In this paper, we would explore this aspect of blockchain technology, demonstrating that blockchain could become a very competitive choice for future databases.

Introduction

Since Bitcoin was not designed for general data usage, attempting to use Bitcoin blockchain as a data store proved difficult and expensive. To discourage its blockchain from being used as data stores, Bitcoin protocol limited legitimate data usage to the scale of 100 bytes per transaction (this limit changed several times but the order of magnitude is essentially the same). Illegitimate ways of storing data via splitting data into smaller pieces exist, but of course it introduces complexity and overhead, thus proving difficult and expensive. This limitation of Bitcoin is put in place deliberately, as accommodation to data usage is a conflicting goal to system performance. It has been a prolonged battle and drama to raise Bitcoin's maximum block size limit, which reflects the inherent scalability limitation of the technology. To promote data usage, it would cause even more consumption of the limited storage resource, while reducing maximum transactional throughput of the system.

The scalability issues stem from the fact that, unlike previous distributed databases, Bitcoin is an extremely redundant system. Each full node of Bitcoin network has a complete dataset of the Bitcoin blockchain, and must also validate the blockchain in its entirety. The cost of such extreme level of redundancy is its impact to scalability that we observe.

Bitcoin initially tackled the redundancy level via a system called light-weight validation, which cleverly organized transactions in a Merkle tree data structure such that users can still follow the blockchain consensus in a decentralized fashion by only using light-weight nodes. This technique significantly lowered redundancy level of Bitcoin network. Currently, number of light-weight nodes far exceed number of full nodes in the Bitcoin network.

Blockstream later suggested that applications may be offloaded to sidechains [Back 2014]. In order to maintain a Bitcoin-centric world, a pegging system to Bitcoin was proposed in this scheme.

Ethereum proposed to tackle the redundancy level via sharding. Sharding is a distributed database technique to divide a large database into smaller 'shards', which is stored at different nodes. This system introduces risk of reduced availability of shards, due to reduction of redundancy. To counter this risk, Ethereum likely would require the existence of several highly available full nodes storing the whole of the blockchain.

A recent paper by Plasma [Poon 2017] proposes another solution to scalability.

VEE Platform: Rearchitecting Blockchain Technology

It is estimated that there are already tens of thousands of blockchain projects if not more happening in the world. Cost of developing and maintaining blockchain systems have become a significant barrier of entry to many who are considering such possibilities. It's time to have another look at the blockchain technology as a whole. If we could significantly lower the cost of blockchain technology while improving scalability, it would enable more innovative uses of blockchain and speed up the adoption of blockchain technology.

Blockchain as a Database

The main paradigm shift blockchain technology brings about is decentralization. One should look at blockchain databases from this new angle.

Generally speaking, traditional user accounts can be substituted by public-private keys and addresses in blockchains. Typically, traditional databases are subject to strong access control, almost all data is limited to authenticated accounts. Account creation is of centralized model in traditional databases. That is, a database administrator grants the user an account for access. With blockchains, key pairs are generated freely by anyone, without the need of centralized administration. Much of the data is then considered public access, unless the data is stored in encrypted form in blockchain. Even for private blockchains inside an organization's own LAN, unencrypted data in blockchain should still be viewed as public access, due to unavoidable breach into the LAN. Privacy is instead protected by the anonymity of the virtual identities. Paradoxically this may actually be a stronger privacy protection compared to centralized model where we constantly hear about loss of customer data due to hacking.

What if an application requires some form of centralized administration? This can be achieved with business logic inside client/node software. Privileged key pairs known as *administrators* can be built into the client software. Administrators can then choose to mark those key pairs in violation of service agreement as violators. Administrators can also mark specific data for censorship. The data of the violators or specific data that is inappropriate or illegal can then be disregarded by node software.

Note this type of central censorship is of weak form since violator data is still allowed to enter blockchain, it is just not recognized by the official node software.

What about applications that require customer identification? Typically such applications have account opening requirement to pass identity verification before the account is activated for use. This can be achieved as well inside client/node software to introduce a white list of public keys that have passed identity verification. Only data from this list of public keys are recognized by the software.

With the above concepts in mind, a significant portion of databases in use today may be suitable for migration to blockchain databases.

The platform considers elements in the database as objects. Examples of basic objects:

- Public key: the public part of a key pair generated by users
- Address: an abbreviate form of public key
- *Virtual Identity/Avatar*: long term use identity, as compared to public key, which can be temporary use
- *Organization*: an identity associated to and managed by multiple virtual identities/avatars
- *Fungible*: virtual asset/token of a fungible nature, such as currency, share etc.
- *Account*: a container of fungibles for an identity, like a bank account. Not to be confused with user account of traditional databases.

The followings are examples of basic relations:

- Ownership: relations between identities and objects
- Creation: relations between objects and identities who create them
- Issuance: relations between token issuer and fungible

The followings are basic user database operations:

- Create database
- Insert object
- Update object
- Delete object
- Create index
- Query by index key value

Objects, in the form of JSON objects, are pretty powerful data structures to represent structured data. Key-value pairs can be considered a simple example of objects. A key in a key-value pair should not be confused with public key of a virtual identity. This term is sometimes also referred to as name-value pair to avoid confusion. Key space or name space scope in the database can either be local to the user or global.

Under an ownership type of data model, the data object may be regarded as 'owned' by the identity who inserted it, meaning it can only be modified or deleted by this owner. For global namespace there is global namespace resolution problem. This can also be understood as global uniqueness constraint problem. When a user attempt to insert a key-value pair, an observer sees the key or name in the broadcast and then makes a competing insert of the same key or name, which may get confirmed into blockchain instead of the original insert. Namecoin introduced a protocol to deal with this issue. The idea goes like this:

- User sends a pre-insertion reservation transaction, where the key/name of the insertion is hidden via hashing. The protocol understands that the reservation transaction reserves the insertion of the given key for some period.
- Wait for the pre-insertion reservation transaction to confirm.
- Then broadcast the actual insertion transaction to the network. The insertion transaction should include a link/reference to the reservation transaction to pass protocol validation that the insertion and the reservation match each other.

Since the squatter does not know what the key or name is at the time when the reservation transaction is broadcasted, it wouldn't be able to get in before the actual owner, unless a reorganization of the blockchain happens after the insertion transaction is broadcasted.

Although, this still cannot prevent squatters to guess what other people want and claim them in advance, like what happens in the domain name system.

Ownership can be transferred. During the transfer transaction the owner of the object is modified.

By default, only owners can modify or delete an object. However other models which allow more flexibility are also considered. For example, a documentation or wiki application may not require ownership of each data record. Once the object is inserted, everyone is free to modify or delete them. Another possibility would be a relationship of a white list of identities who are allowed to modify the object.

Advanced Database Features

The platform also plans to introduce advanced database query features. An object-relational query language such as those similar to MongoDB, is more flexible than the traditional relational query model, also known as SQL. Google's MapReduce also presents a new form of data processing.

Database Migration

As a database, migration features are considered important. As database scales, it would be more cost effective to migrate it to a separate blockchain of its own, so the blockchain fees can

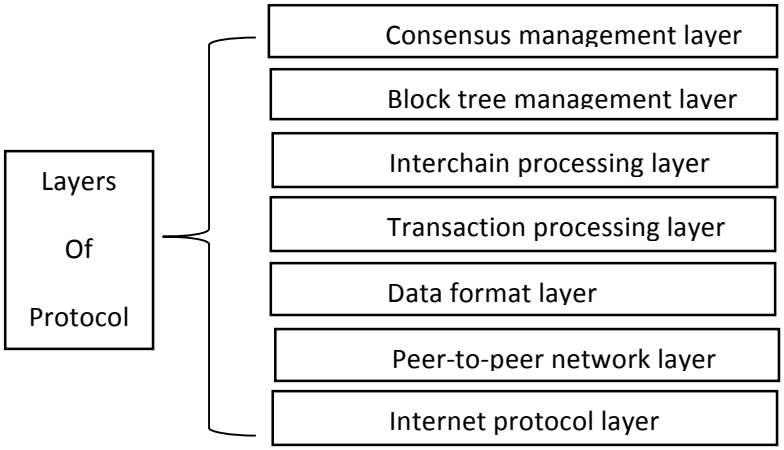
be lowered specific to the application itself. The platform plans to provide migration tools to move database from one blockchain to another.

Modularity Goals

Modularity is an important design goal to lower system complexity and reduce future development and maintenance cost, not only for the platform itself but also for the individual blockchains running applications in the ecosystem.

Layers of protocol:

- Consensus management layer
- Block tree management layer
- Interchain processing layer
- Transaction processing layer
- Data format layer
- Peer-to-peer network layer
- Internet protocol layer



System components:

- Pluggable consensus models
- Pluggable business logic container
- Database management component
- Database operation component
- Database query component
- Shared peer-to-peer networking

- Full node with blockchain processing
- Smartphone based light-weight cold wallet
- Smartphone based light-weight hot wallet
- Browser based wallet

Consensus Systems

The original Bitcoin proof-of-work consensus is now known as the *Nakamoto Consensus*. Nakamoto Consensus is The Breakthrough where it all began. For more than 8 years in running, Bitcoin's system certainly proves its reliability.

Primecoin [King 2013] introduced prime number based proof-of-work consensus, which we refer to as *Primecoin Consensus*. Primecoin Consensus was the first and still the only consensus system to generate interesting byproducts with mining, while achieving network consensus at the same time with predictable level of security. Primecoin has been running for over 4 years reliably.

Proof-of-stake Consensus was first introduced by Peercoin [King 2012]. The main difference is that instead of allocating weight based on computing resource consumption as in the case of Nakamoto Consensus or Primecoin Consensus, proof-of-stake consensus systems allocate weight relative to the amount of coin holdings participating in the consensus activity, also known as block minting. This algorithm decouples the consensus security level from system energy consumption level and eliminates the requirement of energy consumption in order to reach consensus, thereby resolving the energy consumption concerns over Nakamoto Consensus, while reducing overall system operating cost in the process.

Proof-of-stake Consensus is a major breakthrough as it significantly lowers the cost of Blockchain technology and barrier of entry, thus enabling a vastly diversified blockchain ecosystem. At some point in the future we believe the number of operating blockchains could exceed the total human population of the world. Proof-of-stake Consensus is the enabling technology for blockchain technology to reach such a massive scale.

Several major networks have been running proof-of-stake consensus systems for several years, with some variations between each other. Proof-of-stake Consensus has proven its track record. The platform plans to implement at least the above three different innovative consensus systems which have proven themselves.

There exists other consensus systems, the platform will evaluate their reliability as the project moves forward. A consensus algorithm with innovative ideas and proven reliability will be a candidate for implementation in the platform.

Mainchain-Sidechain Model

The platform introduces its own model for *mainchain* and *sidechain*.

A blockchain **S** is called a sidechain of another blockchain **M**, the mainchain, if **S** satisfies:

- Awareness: full nodes of **S** are also full nodes of **M** and process the entire blockchain of **M**
- Synchronization: **S** observes *abstract clock synchronization* to **M**

Abstract clock synchronization deals with ordering of blocks between the two blockchains. Imagine the blockchain as an abstract clock, whereof each block in the chain is a clock tick. It is called abstract as it has nothing to do with the local timestamps written into the blocks. Timestamps are local values that cannot determine the correct ordering of events globally. Instead, block number inside blockchain can determine a global time sequence. Observers can safely say events in a previous block always happen before events in a later block regardless their timestamps.

When a sidechain block is generated, it links to the latest mainchain block as its *mainchain parent*. Multiple consecutive sidechain blocks are allowed to share the same mainchain block as their mainchain parent. This mainchain-sidechain parent-child relationship must also be order-preserving.

This model of mainchain-sidechain allows us to develop a proprietary communication method between the two blockchains. Unlike Blockstream's proposal, our model does not require pegging on sidechains, thus giving sidechain projects much more freedom to innovate.

Cloud Features

The platform plans to provide toolsets to sets up blockchain for applications. Blockchain template preparation allows user to choose from different protocol parameters and pluggable components such as consensus model.

Once template and options are selected, the platform provides toolset to deploy a new blockchain for the application, possibly even before a specific business logic needed for the application is developed.

Smart Contracts

Smart contracts [Szabo 1996] allow parties to create binding agreements without a third trusted party. Bitcoin used a simple scripting system when validating a transaction. But this scripting system is quite limited and for fear of potential issues Bitcoin restricted its use among standard transactions. Later, Ethereum [Buterin 2014] redesigned a new smart contract system with a new Turing-complete programming language known as Solidity. This is a significant step forward for blockchain technology allowing autonomous and decentralized contracts to be realized for many application scenarios.

EOS recently proposed to implement another smart contract system utilizing WebAssembly, also known as wasm. Wasm is an emerging Web standard for low level in-browser client side scripting. Wasm is typically developed via C or C++ and compiled to Wasm.

The platform plans to support compatible implementations of Ethereum and EOS style smart contracts. Virtual machines will be implemented in a modular fashion so applications can choose to enable a preferred style of smart contracts. As more competing smart contract systems are developed by the industry they would also be evaluated and considered.

Scalability

A great deal of effort has been spent on the scalability limitations on a single blockchain. While some of them may be notable, we believe ultimately the future of scalability lies in the ecosystem of unlimited number of blockchains. As mentioned before, we envision a world with possibly billions of blockchains operating at the same time. The platform allows applications to be run in separate blockchains if necessary, achieving complete scalability isolation to other application systems in the same ecosystem.

Usability

Usability has long been a bottleneck for the general acceptance of cryptocurrencies. The platform plans to develop both browser based wallet and mobile light-weight wallet for smartphones with modern user experience and high security in mind. Cold wallet should be easily used by everyone allowing users to safeguard their virtual assets with peace of mind, free from the threats from the dark corners of the Web.

Conclusion

VEE platform is aimed to significantly lower cost of blockchain technology and massively increase the competitiveness of blockchain as a database platform compared to traditional database systems. It is our vision that the future of blockchain is not only in a few billion dollar blockchains, but also in billions of blockchains as well to bring a new economic era to the world.

References

[Back 2014] Enabling Blockchain Innovations with Pegged Sidechains, https://blockstream.com/sidechains.pdf

[Buterin 2014] Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform, http://www.the-blockchain.com/docs/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf

[King 2012] PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake,
https://peercoin.net/assets/paper/peercoin-paper.pdf

[King 2013] Primecoin: Cryptocurrency with Prime Number Proof-of-Work,
http://primecoin.io/bin/primecoin-paper.pdf

[Lamport 1982] The Byzantine Generals Problem,
http://lamport.azurewebsites.net/pubs/byz.pdf

[Nakamoto 2008] Bitcoin: A Peer-to-Peer Electronic Cash System,
https://bitcoin.org/bitcoin.pdf

[Poon 2017] Plasma: Scalable Autonomous Smart Contracts,
https://plasma.io/plasma.pdf

[Silver 2016] Mastering the game of Go with deep neural networks and tree search,
https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf

[Szabo 1996] Smart Contracts: Building Blocks for Digital Markets,
http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html