RealTract Whitepaper V2

Contents:

EXECUTIVE SUMMARY

This whitepaper describes how RealTract is going to disrupt and transform the blockchain industry and cryptocurrency market with the launch of the first practical smart contract in the world. RealTract is going to launch Smart Contract 2.0 on the Blockchain 4.0. RealTract aims to create a truly democratic and decentralized blockchain enabling common users to get the benefits from the blockchain technology and digital currencies.

There are many technical difficulties around blockchain core technologies, which need breakthroughs and out of box thinking. At present, the infrastructure to support development of blockchain applications is unstable and time consuming. There are a number of major problems related with the current blockchain and smart contracts such as Low transaction throughput, Energy costs, Difficult to use, Uncompetitive Applications, low Transaction speed, Transaction fees, Issues of interoperability and Platform lock-in, lack of robust security for smart contract 1.0, etc.

The current blockchain architecture and technologies along with application scenario are limited by performance, applicability and stability of the underlying chain. Therefore, there is an urgent need to study the underlying mechanism of Blockchain, and redesign or improve the various key technologies of blockchain to solve the various associated problems.

RealTract aims to solve the problems such as low applicability, transaction congestion, high commissions, long confirmation latency, weak resistance to quantum attacks, slow communication and transactions, incapability in crossing and merging chains, large space for storage and etc. RealTract would optimize and improve blockchain technology in all aspects including protocols and mechanisms, and become a genuine infrastructure of Blockchain 4.0.

1. INTRODUCTION

1.1 Blockchain

Blockchain is an open, transparent, and distributed ledger that can record the transactions between two parties or a group in an efficient, permanent and verifiable way. Blockchain is a continuously growing list of digital records, called blocks, which are linked and secured using the cryptography processes. Each block contains a hash pointer as a link to a previous block, transaction data and a timestamp. Blockchains are secure by design and inherently resistant to the modification of the data.

Blockchain formation is based on a lengthy and time consuming process. The main chain consists of the longest series of the blocks from the genesis block to the current block. The blocks existing outside of the main chain are orphan blocks and have little relevance or no value in the system.

For use as an effective distributed ledger, a blockchain is managed by a peer-to-peer network that collectively adheres to a specific protocol for validating the new blocks. Once recorded, the data stored in any given block cannot be changed retroactively without changing all the subsequent blocks, which requires collaboration of the network majority. Thus, any unilateral or unauthorized changes are next to impossible.

Blockchain is based on a design that prevents the owner of a currency token from committing a fraud by spending it twice. The first spend is recorded publically for all to see, so no one would accept a second spend.

Decentralized consensus is achieved with a blockchain. This makes blockchains potentially suitable for the recording of events, medical records, and other records management activities, such as identity management, transaction processing, documenting provenance, voting, or various other phenomena.

Blockchain is leading to a sea-change in the internet's evolution, from the internet of information to the internet of value with its capabilities to generate unprecedented opportunities to create value. To stay functional, it needs a lot of computing power and continuous innovation. In 2017, despite the cryptocurrency boom, there was a limited spread in the reach of the cryptocurrency and blockchain in terms of the common masses due to complex technical issues and time consuming processes.

1.2 Smart Contract 1.0

Smart contract basically refers to the self-executing computer program or protocol to digitally facilitate, verify, and enforce the performance of a contract automatically as per the rules defined in the contract without third parties under specified conditions. Smart contract has become very popular with the advent of decentralized blockchain network.

A smart contract is a special protocol intended to contribute, verify or implement the negotiation or performance of the contract. Smart contracts allow performing credible

transactions without third parties. These transactions are publically traceable and irreversible.

Smart contracts contain all the information about the contract terms and execute all envisaged actions automatically. The terms of the agreement between buyer and seller are directly written into the lines of code existing across a distributed and decentralized blockchain network.

At first, assets and contract terms are coded and put into the block of a Blockchain. This contract is distributed and copied multiple times between the nodes of the platform. After the trigger happens, the contract is performed in accordance with the contract terms. The program checks the implementation of the commitments automatically.

For creating a smart contract you need subject of the contract with the program having access to goods or services under contract to lock and unlock them automatically, contract terms in the form of an exact sequence of operations, participants initiate an agreement by signing the contract with their private keys, and deployment of the contract to the decentralized blockchain platform and distributed among the nodes of the platform.

There are numerous applications and usage of the smart contracts like logistics and supply chain, financial services, bank systems, insurance, real estate, IoT, voting results data in the encrypted and anonymous form in the blockchain eliminating manipulation possibilities, and so on. Blockchains and smart contracts can solve areas of business and general life that frustrate the users like the flawed banking system and dysfunctional global payments system.

NASDAQ is piloting a stock exchange off blockchain technology. Microsoft launched blockchain as a service last year. Smaller companies are building dozens of apps on blockchain. According to CoinDesk, a couple in Singapore recorded their prenup on a blockchain, specifying that “every 10 days, 100 minutes must be spent on a date night, that shopping sprees shall be limited to once per fortnight, and so on.

Smart contracts provide various benefits of blockchain technology such as:

- Automation: The various processes are automated through programming.
- Economy: The costs are reduced as intermediaries are eliminated.
- Security: The smart contract is well encrypted and distributed among network nodes, guaranteeing that it will not be lost or changed without proper authorization.
- Standardization: There is a very wide range of various types of smart contracts available at present and they have been standardized as well.
- Customization: You can choose a standard smart contract template and change it according to your needs.

1.3 The problem of current blockchain and smart contract 1.0

There are many technical difficulties around blockchain core technologies, which need breakthroughs and out of box thinking. At present, the infrastructure to support development of blockchain applications is unstable and time consuming. Thus many applications are not effective. It is imperative to make research and development on blockchain infrastructure, thus providing reliable support for various blockchain applications, as well as promoting implementation of blockchain applications in all kinds of

industries in order to make blockchain technology more effective in serving mankind. The major problems related with the current blockchain and smart contracts are as follows:

- Low transaction throughput: Performance is one of main challenges for current blockchain technology. The current blockchain platforms and applications have a low transaction throughput. Bitcoin is designed to handle only seven transactions per second, and Ethereum can only handle a few more.

As the current blockchains are simple concatenations of single data entities state changes; reconstructing the actual states of these entities implies a whole chain scan, which causes an even greater system slowdown and resources usage. This slowness is caused by the lack of horizontal scalability, i.e. the increase of computation capacity obtained by merely adding processors. The current blockchain safety mechanism is designed to prevent anyone from taking over the majority of the clusters by making it very expensive to achieve in terms of calculation power and/or cost. This causes the problems of scalability and low transaction throughput.

- Ability of extension: Upgrading blockchain networks poses serious challenges. All nodes on a blockchain network must validate the same blocks. It is therefore impossible for any subset of parties to freely upgrade to a new protocol without affecting the rest of the network; or permanently detaching from it by forking the chain. Simple chains of data are not flexible enough to fulfill emerging needs, in which complex data structures need to be organized. At the same time, those structures need to be validated and made immutable with blockchain-based techniques, increasing traceability and security.
- Energy costs, depending on the miners: The threshold for current blockchain has been rising, which is reflected not only as a higher requirement of technical performance, but also as increasingly fierce competition of computing power. Particularly in the early era of Bitcoin mining, the personal computer alone could easily mine the coin. Later, mining became more demanding of a computer's performance. This led to emergence of specialized mining computers. As a result, there is an increasing requirement of nodes. Most of which, are controlled by a select number of mining pools. This further isolates the blockchain from the masses. Moreover, the heavy energy consumption also has adverse impact on environment, which is a major cause of concern in today's world.
- Difficult to use: Today's blockchain applications are built for the tech people who know how to use them, rather than common users. Nearly all blockchain applications require users to either run a blockchain node or install a light node. It takes a long time for users to adapt to application. To attract large number of people, blockchain applications need to be as simple as today's Internet and mobile apps. Blockchain technology should be easy to understand and use for the consumer.
- Uncompetitive Applications: The current consumer applications must be able to handle tens of millions of active users daily. In addition, some applications will only become valuable when certain throughput is reached. The platform itself must be able to handle a large number of concurrent users. A fine experience demands reliable feedback within only seconds. Long latency frustrates users and makes applications built on blockchains less competitive with existing non-blockchain alternatives.
- Attack potential 51%: High volume crypto-assets platforms are constantly attacked by hackers who seek to bring the systems down, typically through DDoS attacks.

Fraudsters also try to break into accounts using social engineering to steal cryptocurrencies from users. Many high-volume platforms could not withstand these attacks and were forced to shut down. It is estimated that since 2011, at least three dozen major heists against cryptocurrency exchanges occurred. Close to 1 million BTCs were stolen.

- Transaction speed: Bitcoin, based on first-generation blockchains and Proof of Work algorithm for transaction validation, and the second generation Ethereum based on the smart-contracts-enabled blockchains, are extremely low energy efficiency with low block validation speed and transactions per block. The third-generation and the fourth generation blockchain solutions, using techniques like Proof-of-Stake validation algorithm, off-chain routing, graph-chains, and complete or partial centralization, are still not able to solve the issues of scalability, speed, and energy consumption.
- Transaction fee: The traditional systems involve high transaction fees resulting from high costs due to low transaction speed and low throughput. Even simple applications can slow down the Ethereum based platform and increase transaction fees dramatically.

The high cost of using blockchain technology is a major barrier to mass adoption. It also limits developers who need the flexibility to build free services. Just like today's Internet and mobile Apps, there is no need to pay for every operation during blockchain transaction. Similar to the Internet, blockchain technology should be able to support free applications. Making blockchain free to use is key to its widespread adoption. A free platform will also empower developers and businesses to create valuable new services they can monetize, rather than having users pay fees to use the blockchain network.

Issues of interoperability and Platform lock-in: The current blockchains have critical platform lock-in problems. Developers have to decide which blockchain to develop, and then implement platform-specific code, which makes it very difficult to switch an application to another Blockchain. Developers don't want to be locked into working with a certain blockchain technology. They need freedom to evaluate, use, and switch between options. Some applications may even need to run on multiple platforms to provide best user experience.

- The security of smart contract 1.0 (Weakness): The security measures stop at data level as they don't ensure user safety, making it impossible to recover lost or stolen coins and tokens even if they are located on the chain, or to block malicious accounts.
- The practical application of smart contract 1.0.

The current smart contracts now, typically the Ethereum smart contract, is called smart contract 1.0. Although it is most commonly used in today's ICOs and has realized its enormous potential, but its applicability in the real world is too low: Smart contract 1.0 works on the basis of the principle of one step operation.

But this creates practical difficulties and awkward situations. Imagine a practical application to the situation:

"*You buy a house. When the seller has received the money, you own the house and find the serious damage in the concrete frame, the walls have heavy cracks, and the seller just*

covered your eyes when you came to see the house only 1-2 times. If you know that, would you be ready to buy that house?"

o Actually, **A** makes the contract,

o **B** transfers the money to **A;** but still needs the conditions to ensure that product is correct and fulfills **B**'s expectations.

· Too many third parties use smart contract 1.0 on the Ethereum platform to provide real services that render the reliability and performance of the user perturbed with unverified information.

Smart contracts in the present form have some deficiencies like

· Slow Speed: The processes related to smart contracts are complex and lead to slow execution.

· High implementation costs.

· Complex and lengthy programming/coding.

· Lack of arbitration and conflict resolution in case of dispute/problems.

· Human involvement as the code is written by people leading to mistakes like DAO developers's mistakes in the code caused huge losses to the users and the company.

2. VISION AND SOLUTION OF REALTRACT

Admittedly, blockchain in general and smart contract in particular is a big step forward for the mankind. But go first does not mean being all good; seeing the weaknesses of the current blockchain platforms in terms of scalability, speed and performance. RealTract is born as a thorough solution to the disadvantages of the existing blockchain and the current smart contract, adding and improving technologies to help put blockchain and smart contract into real life applications effectively and efficiently.

RealTract would optimize and improve blockchain technology in all aspects including protocols and mechanisms, and become a genuine infrastructure of Blockchain 4.0. Also, RealTract would provide a platform for developing various DApps (distributed Apps), as well as feasible solutions to create Smart Contracts 2.0. RealTract focuses on core technology of blockchain infrastructure and platform. Our goal is to build an infrastructure conquering current key technical problems and supporting all domain applications in terms of ecological view.

RealTract aims to implement a real practical and powerful support mechanism for blockchain, and provide the infrastructure for all kinds of blockchain based applications, and an underlying development platform for all kinds of DApps and practical and feasible solutions for constructing the global blockchain of future.

The RealTract uses the following elements to solve the current problems of blockchain industry:

2.1 Blockchain 4.0

Blockchain 4.0 is based on the basic principle of DAG protocol that we upgraded and called as Infinity Block Graphs (IBG). Many nodes of the elements in the RET Chain through the Proof-Of-Result algorithm helps maximize transaction processing capabilities, unlimited scalability and a high-reliability consensus mechanism.

2.2 Smart Contract 2.0:

Nowadays, smart contract is on its way to become the leading technology which could simplify the interaction between parties but intensify the efficiencies of a contract; and thus change the way we sign contracts. Smart contract could guarantee the transparency and justice of a contract between parties; also many people believe that it could transform certain industries in the near future.

However, there is still a major problem of smart contract, which according to the majority of technology experts, could reduce the applicability of smart contracts in real life even if its potential is enormous. The current concept of smart contracts, which considers the rules and conditions of a smart contract is unchangeable, might bring litigations to both parties if any of the rules and conditions need to be changed.

Most of the current smart contracts are pursuing the perfection of performance and automation because most of the developers believe that the necessary laws and rules cannot compare to the importance of perfect code and fast performance. Unfortunately, it becomes the disadvantage of smart contracts, and makes it difficult to apply to practice, especially in other industries.

The smart contracts are on the radar of the top global firms, MNCs, and constancies that are researching their role and giving them increasing importance keeping in view their future potential for the industry, business, and governments.

Therefore, RealTract has invented the advanced version of smart contract called Smart Contract 2.0 (Practical Smart Contract), which is a significant improvement over the Ethereum smart contract 1.0. In addition to maintaining the excellent operating mechanism of the older version, smart contract 2.0 adds conditions and algorithms to completely protect the interests of the parties when participating in the transaction

3. Technical Overview

The goal of RealTract is to create a trustless and decentralized system in which transactions are similar to real-world transactions. RealTract accomplishes it by designing its network as a multi-layer blockchain with PoR consensus algorithm to allow transactions to be linked with additional information on-chain. Users, developers, node operators,

organizations, enterprises, crypto-exchanges, partners, and other blockchains & cryptos can take part in the development of RealTract.

3.1 RET Chain

3.1.1 Infinity Block Graphs (IBG)

3.1.1.1 Introduction

Infinity Block Graphs are similar to DAGs. A DAG is a finite directed graph with no directed cycles. It consists of finitely many vertices and edges, with each edge directed from one vertex to another. The key structure which makes DAGs work is a Tangle. The Tangle is a particular kind of directed graph, which holds transactions. Each transaction is represented as a vertex in the graph. When a new transaction joins the tangle, it chooses two previous transactions to approve, adding two new edges to the graph.

Infinity Block Graphs also work on a similar concept with multiple nodes and directed connections between them.

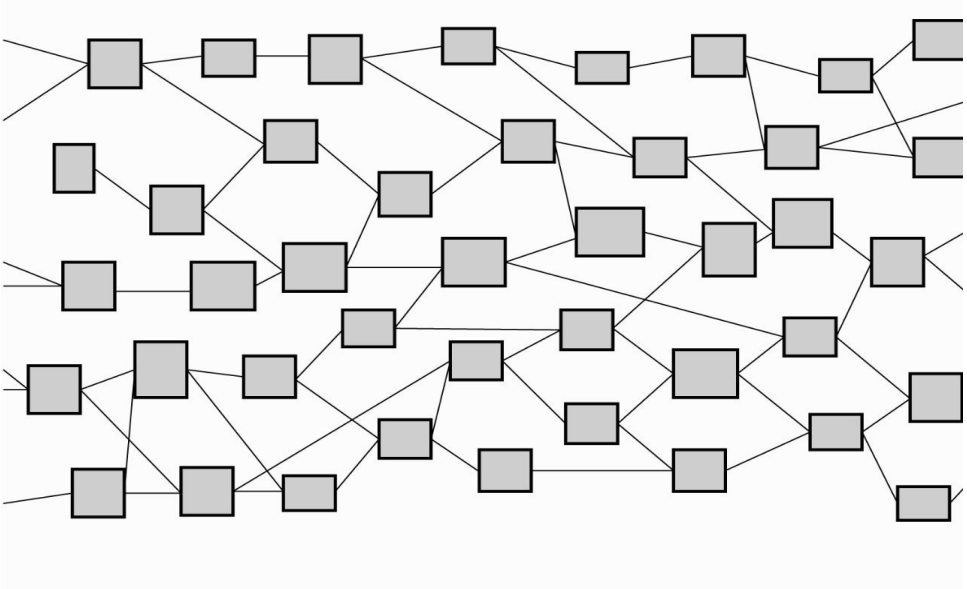


Figure 1: DAGs protocol

3.1.1.2 Definition

A round of consensus takes the resulting hash block of a previous round and adds it as a regular transaction to the transaction pool. The filling of the transaction pool is isomorphic to an unfold operation. Once this checkpoint block is filled with the previous round plus new transactions, it is hashed. This is isomorphic to a fold operation.

A IBG $i : A \rightarrow C$ can be defined in terms of its separate anamorphic and catamorphic parts. Take the definition of a IBG

$i = [(c, \oplus),(g, p)]$ (1)

The anamorphic part can be defined in terms of a unary function $g : A \rightarrow BA$ defining the list of elements in B via repeated application or unfolding, and a predicate $p : A \rightarrow$ Boolean providing the terminating condition.

The catamorphic part can be defined as a combination of an initial value $c \in C$ for the fold and a binary operator $\oplus : B \times C \rightarrow C$ used to perform a fold.

In our case the gossiping of messages is our anamorphism and the hashing of those messages plus the result of the previous block is our catamorphism.

If an operator is defined as a cryptographic hash function, $\oplus$ = CHash and $g\ n = (n, n - 1)$ and $p\ n$ = False (as our chain has no termination condition) for the nth iteration, RETChain is defined categorically as

RETChain = [(GenesisBlock, $\oplus$),(g, p)] (2) where GenesisBlock is our starting element, namely the genesis block used when we deploy RETChain.

3.1.1.3 Components

Blocks

The Block data structure includes the following:

• Stored Data: A Block can contain multiple data packages. There are several types of data package, depending on functions such as transaction, smart contract, history information, reputation management, compensation etc.

• Signature: The signature of the user who created the Block is included and the user is identified through an account or address.

• One or more hash values of the previous block: This is included to provide links between Blocks.

Like other Blockchain technologies, where the new block verifies all previous blocks (including the transactions inside them), all new Blocks will verify only their parent blocks. A new block will be connected to its parent block through hash and all hashes will be derived from parent blocks, so that it is impossible to modify or delete the previous blocks. When an block is connected, another node will build a new block on top of that block.

Score Table

The Score Table is a data structure that could save the connection data of specific blocks. The data structure includes the following:

- Trusted Score (TS)
- Connectivity: Contains information about the connection with other Comma

Comma

A Comma is a Block that contains a Score Table and can see the supra-majority of blocks created in the path of previous blocks. The Block that can connect a supra-majority among

blocks will be appointed as the Comma. Based on their TS, each Comma will have voting right for Dot and for the consensus of other blocks.

Dot

Dot is a set of special blocks which run validation processes and host the current chain state and hence seed the chain state/history to nodes on the network. It is appointed based on the information in Comma and constitutes the Main Chain.

Main Chain

The Main Chain consists of Dot and related blocks. The Main Chain intends to be used for the validation of blocks and to maintain the entire network structure.

3.1.1.4 Structure

The IBG technology intends to achieve high performance and secure data storage. All blocks can be created asynchronously from nodes and each of these blocks consists of a set of transactions (payment, remittance, smart contract, story, reputation, rewards). The new block is connected to the parent block, which is the most recent previous block, and the node is intended to generate the block at a high speed through the PoR.

The PoR can be represented by a graph where all blocks are connected. There exists a most trusted chain that could be connected through set blocks and it is called the Main Chain.

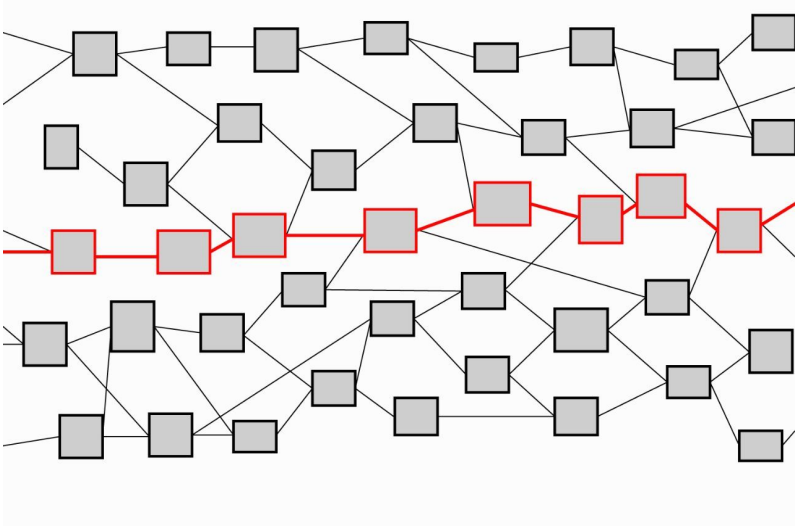


Figure 2: IBG Mainchain

The Main Chain is a set of blocks that can validate blocks created over a period of time. The IBG can effectively solve various problems such as double-spending issues or malicious attacks by intentionally generating incorrect blocks while maintaining the Main Chain. The Main Chain has an influence on the ordering between blocks that occur asynchronously. The Main Chain helps blocks that occurred earlier to have a priority in the sequence. At the heart of the Main Chain is Dot

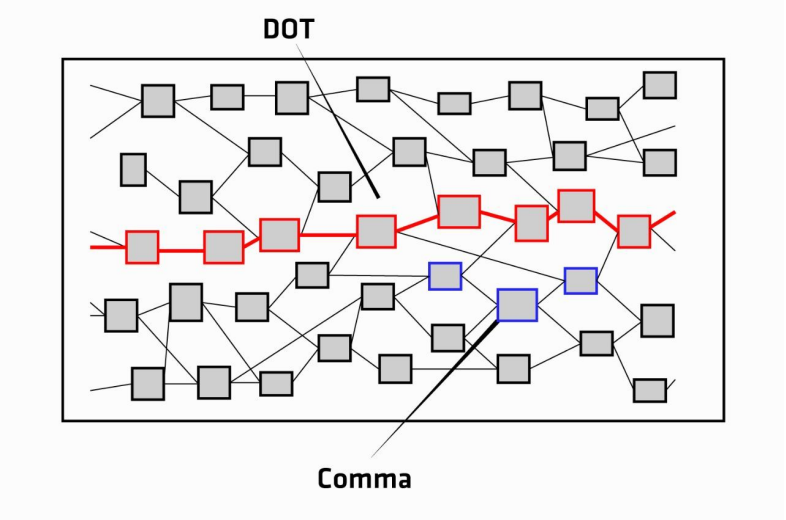


Figure 3: Dot and Comma of IBG

In the RET Chain, there exists a Comma which has a supra-majority (more than two thirds) connection with a set of blocks. In a random distribution of blocks the Comma is located in a certain location at a minimum deviation from the supra-majority. Each Comma has a Score Table that stores its connection information with another Comma set. By using the Score Table which is the connecting information between Comma and the trusted score of each one, the Dot is designated. During the process of designating an Dot through the information of the Score Table, consensus of the block in between the Comma set is met. Such consensus will implement aBFT

3.1.2 Proof-of-Result

Fault tolerance in RET Chain can be improved with a reputation system for selecting delegates (Dot). We therefore present Proof-of-Result; a distributed consensus method that incorporates a node's historical participation into delegate selection. Definitively, a Trusted Score (ST) is a unit representing an imitable idea or behavior that can be spread. Thus in our case, it represents benevolent behavior across RET Chain that is rewarded and should be imitated to improve a nodes overall reputation within the system. Proof of Result is meritocracy compared to Proof of Stake which is a plutocracy.

A ST in our sense is a feature vector corresponding to each block; in the simplest case it is a matrix of float values used as input to a deterministic machine learning algorithm. We follow in the footsteps of REGRET, which uses an ontology of features to describe a node's reputation within in a network; technically, this is a tensor product of feature spaces. A corresponding deterministic machine learning model, which uses this ontology (ST) as it's feature space, is used to determine the node's reputation score and transitively the probability of said node being chosen to participate in consensus.

A ST is the basis of the reputation score and the reputation score gives the probability of being selected for consensus.

Probabilistic delegate selection has been studied extensively within the context of improving BFT in asynchronous consensus mechanisms GURU. Specifically in their work on GURU, A. Biryukov et al. have shown that the typical fault tolerance of 1/3 malicious nodes in

byzantine consensus can be improved to up to (and in some cases over) 1/2. We incorporate their validation framework for our delegate selection in PoR.

There is a body of work related to Extended Trust Chain that solves the problem of sybil resistant reputation modeling. Specifically the P. Otte"Sybil-resistant trust mechanisms in distributed systems" uses the NetFlow algorithm to prevent sybil attacks and a modification of PageRank, namely Temporal PageRank, to update reputation state. IBG will first replicate Temporal PageRank and then improve where possible during performance analysis.

Our use of reputation based scoring for delegate selection with Proof-of-Result, in conjunction with each node's role as an individual account, allows our permissionless network the benefits of a permissioned system by enforcing transparency which incentivizes good behavior. All transaction and consensus history, for every block, is publicly notarized. This allows us a kind of trust that is neglected in existing technologies which enforces good behavior with transaction fees and inegalitarian consensus mechanisms.

3.1.2.1 Proof-Of-Result Explained

We have discussed three bodies of work that solve issues with collusion in distributed consensus. These are all components to our delegate selection as follows. We will follow the approach used in REGRET to develop an ontology for a ST, which will become the feature space for our reputation score. We will adapt Temporal PageRank to update reputation state. We incorporate GURU's validation framework for our delegate selection method. Further details will be added during the development process. Building a reputation ontology, training a deterministic model for reputation score and implementing a sampling algorithm for delegate selection are key components of our development.

We now detail how this score is used for delegate selection. Each feature is representative of a block to the network (has it been notable faulty during consensus, how much memory can it commit to consensus and so on and so forth). A ST can be transformed into a numeric value by passing it into a function, a to-be-trained model as described above, which is how we will quantify a ST's reputation. In order to provide a mechanism that promotes new blocks to improve their ST, a probability distribution of ST scores is drawn over buckets of fixed size; the distribution outlines the probability that a ST of a score in a particular bucket gets selected. From a computational perspective, this will probably be implemented as a clustering algorithm that re-clusters blocks as a subroutine. This is meant to maximize throughput by relaxing the number of facilitators yet maintaining the same level of fault tolerance and confirmation times. Reputable Commas will be given preference, but new Commas will also have an opportunity to participate and build their reputation on the network. Logs of performance will be notarized on chain, so Commas that provide faulty resources or perform in an adversarial way will have their reputation reduced while high performing and trustworthy Commas will have theirs increased. Our idea for delegate selection in the inductive case is as follows:

1) Consensus is performed.

2) The hash block is fed to a deterministic algorithm that outputs updated ST scores for each Comma. This is performed at the top consensus tier (Dot) which is responsible for choosing facilitators.

3) This constant is used to shuffle our distribution (move STs between buckets based on new data of their performance).

4) The previous block hash (result of last consensus) is used to sort the contents of each bucket and the top N from each bucket (according to a probability distribution) are selected for this round. The ST of a consensus participant who acts faulty or provably malicious (as verifiable within logs) will be docked in the next consensus participant election.

As Comma reputation begins to hold more value, it replaces the need for a transaction fee. Stored Data, Transactions may look to reputable Commas for service hosting, which would be more profitable than earning transaction fees for performing consensus. Instead of increasing prices when latency is high, Commas with low reputation scores will be throttled; their transactions processed with lower priority. A Comma in this case would provision resources (participate in consensus), thus increasing throughput for themselves and solving bandwidth issues for the network.

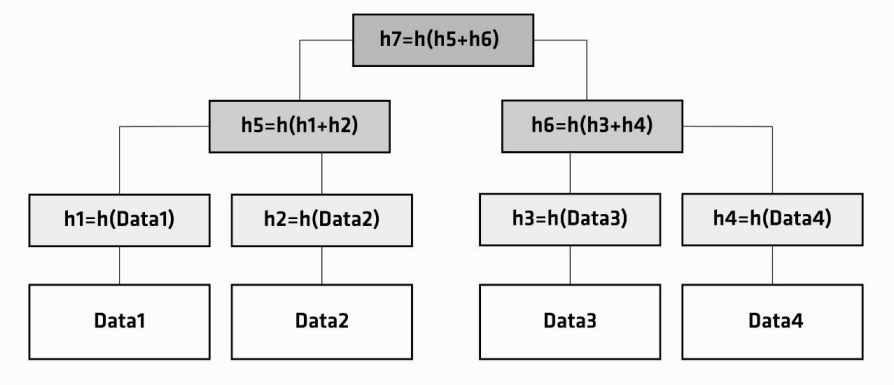
4.2 SMART CONTRACT 2.0

Smartcontract 2.0 was created with **Verschemelized Intellectual Structure (VIS)** technology on RETChain, which are specialized for smaller transaction sizes, more privacy and larger smart contracts. Smart contracts are now implemented by adding time and multi-conditional order into them.

4.2.1 Verschemelized Intellectual Structure (VIS)

Components of VIS include VIS Tower, Floor and Data.

VIS Tower is a way of describing a program by splitting it into its individual parts, which can make it easier to analyze and optimize. To generate an VIS Tower, you connect each function to its dependencies until all of the dependencies have been mapped out. Here's an VIS Tower for the example encumbrance described above:



In the other hand, VIS Tower allow you to verify that an individual element is a member of a set without the whole set being present.

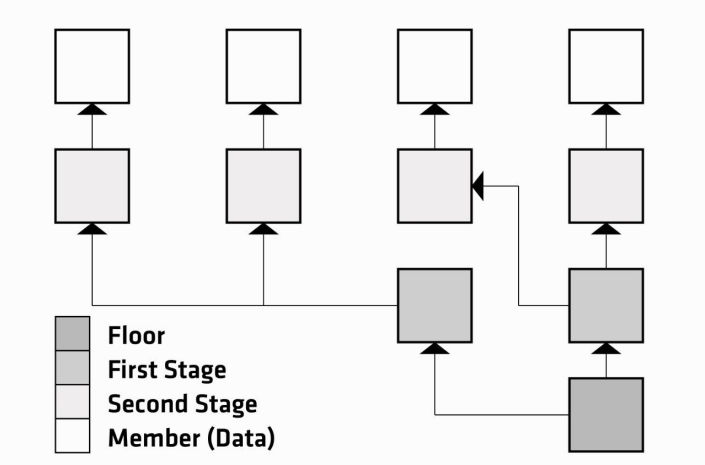


Figure 4: VIS Tower structure

To generate a VIS Tower, each member is individually hashed, producing a short unique identifier for that member. Each of those identifiers is then paired with another identifier and hashed again, producing another short unique identifier for that pair. This step is repeated until only one identifier remains, called the Floor, which uniquely identifies the whole set in just a few bytes of data.

To verify that a particular member is part of the set, someone with the whole set provides you with just the identifiers you need in order to connect that particular member to the Floor of the whole set. This proof that the member belongs to the set is called a VIS proof.

In short, the technique behind VIS allows us to split a program into its individual parts, and VIS Tower allow us to verify the individual parts belong to a complete program without the entire program being present. This is the basis of VIS, which allows spenders to replace the unused parts of encumbrances with a merkle proof—reducing transaction size, increasing privacy, and making larger smart contracts possible.

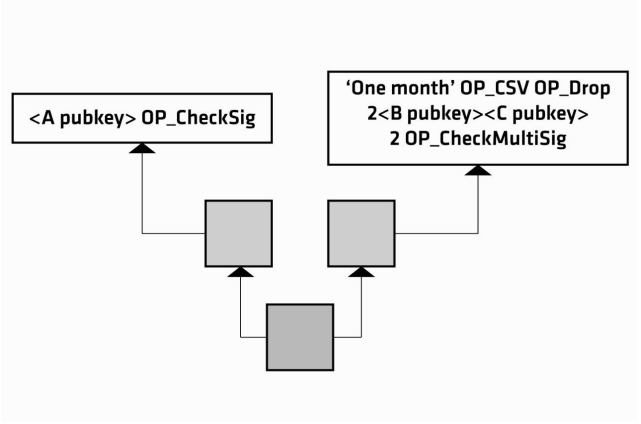
4.2.2 An example of Verschemelized Intellectual Structure (VIS)

Let's take our example encumbrance from above and split it into separate sub-scripts for each of the two possible outcomes we allow:

1. **A** can spend his RETs at any time (below left)

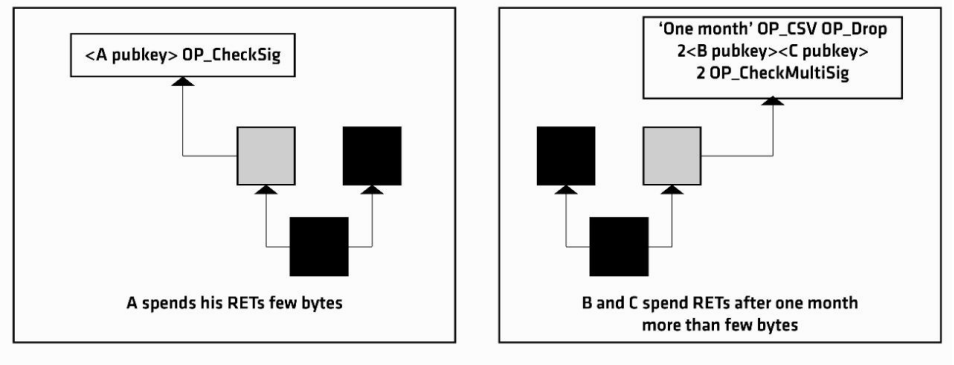
2. Or, after one month pass without **A**'s RETs being spent, **B** and **C** can agree where to spend **A**'s RETs (below right)

Let's create a VIS Tower based on these two independent sub-scripts:



The Floor for this Tower uniquely identifies A's complete encumbrance in just few bytes of data. A then uses a substitute encumbrance that says that a spender must provide a VIS proof connecting the Floor to one of the sub-scripts and that the sub-script must return True.

The VIS proof with sub-script could be visualized like either of the examples below depending on which sub-script we wanted to use:



4.2.3 Benefit of Verschemelized Intellectual Structure (VIS)

Verschemelized Intellectual Structure (VIS) improves the flexibility of RealTract smart contracts, increases privacy, and helps a great deal in scalability.

Since many complex types of conditions can be expressed using VIS to lock up RETs, one can easily have multisig contracts like 1 of 1000 or 20 of 200 types easily.

Also, since VIS requires you to include only that condition in the transaction that you are using instead of the whole set has data benefits. This way your transaction becomes much smaller while simultaneously giving you the capability to engage with complex unlocking conditions.

Lastly, with VIS you get the additional privacy because one can only guess by looking at a VIS transaction that there are conditions involved, but one would never come to know what those conditions were or are.

5. APPLICATION

RealTract is driven by the research and development capabilities. The mission of the project is to build the cost-effective block chain platform and smart contract creation facility at the lowest possible operational expenses. RealTract will share the benefits with millions of the potential and existing cryptocurrency and blockchain users.

Blockchain now has not just established itself as a technology that cryptocurrency is just a subset of; it has also established itself as the solution of the two problems that all businesses face, i.e., Security and Lack of Transparency. Blockchain has many use cases that show how this technology holds the potential to change the world.

There are many applications of RealTract and Smart Contract 2.0 (practical smart contract):

- Retailers: Reduction card fee, payments, delivery contract, warranty, etc.

It can help in a simple transfer, i.e., a sender transfers money to a recipient such as a grocery store.

- Financial service: lending, distributed ledger manager

Big banks, investors and other financial institutions have invested millions of dollars in blockchain, hoping it could make transactions faster, easier and more secure.

It can help in fraud reduction. By bringing all the information on a distributed ledger with a timestamp and batches of specific transactions with a link to another block, the blockchain technology will make it impossible for the hackers to breaks into the system without the timestamp of the breach getting highlighted.

It is estimated that banks spend somewhere around $60 million up to $500 million per year in their 'Know Your Customer' project. These practices are followed to lower the money laundering instances and to keep terrorist out of the banking ecosystem. If the KYC process is brought on Blockchain, the verification time and associated cost will get lowered by manifold.

RealTract can enable reoccurring Payments from a lump sum such as:

o A sender transfers money to escrow

o Every month, $100 is transferred to the recipient like mutual fund from escrow.

- Logistics

IBM has been partnering with leading companies in various industries, including Danish transport company Maersk, to create blockchain-based products that can streamline complex international dealings across sectors.

With the help of blockchain, an employee of a food chain, or even a customer, can grabs a packet of mangoes, types in the identifying number on the package and the entire journey

appears before his eyes -- when they were picked, sent to be washed, sliced, passed through Customs and Border Protection, and when they hit shelves.

It takes roughly two seconds for all of this information to appear. In the event of an E. coli or salmonella outbreak, the difference between two seconds and nearly a week is not only lifesaving but can save a company millions of dollars. Plus, the ability to quickly obtain these specific, secure records could help executives keep tabs on the flow of goods and prevent fraud.

By identifying the production processes and components and then storing the information on Blockchain, business can monitor their supply chain process from the raw material stage to the end delivery stage. For example, Walmart uses blockchain to enable their employees to scan the goods in store's app and then track them from the harvesting stage to the time it reaches the store floor. On the other hand, Makers use of the technology to monitor the cargo ships.

- Government Administration & Public Welfare

The government can utilize blockchain and Smart Contract 2.0 for various usage including even digital identities for refugees who lack official documents. Imagine no longer having a social security card, but a digital identity that couldn't be hacked. We wouldn't have to worry about data breaches like the recent one at Equifax.

Smart Contract 2.0 can cover agreement to a prearranged set of tort laws. These tort laws would be defined by contracts between private arbitration and enforcement agencies, while customers would have a choice of jurisdictions in this system of free-market governments. If these privately practiced law organizations bear ultimate responsibility for the criminal activities of their customers, or need to insure lack of defection or future payments on the part of customers, they may in turn ask for liens against their customers, either in with contractual terms allowing arrest of customers under certain conditions.

We can extend the concept of smart contracts to property. Smart property might be created by embedding smart contracts in physical objects. These embedded protocols would automatically give control of the keys for operating the property to the party who rightfully owns that property, based on the terms of the contract. For example, a car might be rendered inoperable unless the proper challenge-response protocol is completed with its rightful owner, preventing theft. This will help the government administration including police and public authorities.

- Healthcare

The fact that blockchain comes with an immutable architecture makes it possible to store the EHR data in a way that is safeguarded from any or all instances of hacks and breaches.

- 4th Industrial Revolution

Blockchain is contributing in a big way towards the 4th industrial revolution. Many big industrial corporations are deriving benefits from blockchain technology in the product development and manufacturing processes.

Porche, the leading automobile maker has already introduced blockchain in its cars. There are a number of benefits that the brand accepted going blockchain brought for it, e.g., secure access of vehicle, fast data transfer and better security, autonomous driving, etc.

Coca-Cola, The beverage leader, along with the US State Department is developing a blockchain ledger which is designed to remove the state of forced labor from across the globe. Using the technology, they will develop a secure registry for the workers which would help with fighting forced labor market, globally.

IBM recently revealed its chip which they called the world's smallest computer that would help brands use blockchain in verification of authenticity of the products in a supply chain. IBM also uses blockchain to deliver distributed ledger services to over 400 different clients around the world including government, banking, logistics, and healthcare.

- Internet of Things (IoT)

Our easy to use, superfast, and low cost smart contracts 2.0 on the Blockchain 4.0 would help in creating micro-contracts that would enable the users to execute tiny transactions cheaply and securely. Such micro-contracts offer immense opportunities for the various mobile and AI devices and could fuel the internet-of-things.

The IoT sensors get to exchange data on the platform instead of a third party. Also, since the devices are addressable with Blockchain, businesses get an access to the usage history of the connected devices, which comes in handy at time of troubleshooting.

- Online jobs

Blockchain is proving very useful and companies are tapping the potential of the blockchain for their HR and job related processes. Blockchain can help in the online job searches, finding the right job for the right candidate, verification of the records and credentials of the prospective employees, etc. The companies can also keep track of their present and future employees with the help of blockchain.

Smart Contract 2.0 can help in the various online jobs and employee related aspects such as job agreements, fulfillment of the job related conditions, payments, employee benefits etc.

6. Business Model

Article for deployment plan of RealBusiness can be found here:

https://medium.com/@realtractofficial/realtracts-plan-for-realbusiness-b3915443b057

7. Road Map

Being a project established without crowdfunding, RealTract only engaged with the community through our first Airdrop program. Our goal is to develop a complete ecosystem for RealTract and other business plans from the previous project to raise fund for the technological development costs of this project.

- Q1 2018:
    - The RealTract vision defined
    - Whitepaper released
- Q2 2018:
    - Build website
    - Prepare to debut
- Q3 2018:
    - Airdrop preparation
    - Developing Smart Contract
    - Exchange Listing
- Q4 2018:
    - Developing community
    - Developing partnership
- Q1 2019:
    - Release RealBusiness
    - Support establishing our ecosystem
    - Burn token
- Q2 2019:
    - Whitepaper Update
    - Develop DApp and grow communities around business-related applications.
- Q3 2019:
    - Repair for Alpha testnet
    - Testnet lauching
- Q4 2019:
    - Release Mainnet
- 2020:
    - Establish RealTract global technology and service company
    - Expand business activities and business partnerships
    - Invest and support potential start-up programs.

8. References

https://en.wikipedia.org/wiki/Byzantine_fault_tolerance

https://en.wikipedia.org/wiki/Directed_acyclic_graph

"Blockchain". https://en.wikipedia.org/wiki/Blockchain

"Blockchain Technology: Preparing for Change" (PDF). Accenture. https://www.accenture.com/cn-en/~/media/Accenture/next-gen/top-ten-challenges/challenge4/pdfs/Accenture-2016-Top-10-Challenges-04-Blockchain-Technology.pdf

"Contract - Bitcoin Wiki". en.bitcoin.it. https://en.bitcoin.it/wiki/Contract

"Ethereum Whitepaper". github. https://github.com/ethereum/wiki/wiki/White-Paper

"How Do Ethereum Smart Contracts Work? - CoinDesk". CoinDesk. https://www.coindesk.com/information/ethereum-smart-contracts-work/

"Smart contract". https://en.wikipedia.org/wiki/Smart_contract.

"Smart Money: Blockchains Are the Future of the Internet". Newsweek. http://www.newsweek.com/entrepreneursmoneybusinessprofitinternetbitcointransactionblockchaindata-603100.

"What are Smart Contracts" (PDF). Chainfrog. http://www.chainfrog.com/wp-content/uploads/2017/08/smart-contracts.pdf.

Wikipedia. List of data breaches. https://en.wikipedia.org/w/index.php?title=List_of_data_breaches&oldid=811326932.

Leading the pack in blockchain banking: Trailblazers set the pace. https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=GBP03467USEN.

Chu Y, Ream J, Schatsky D. Contracts Get Smarter With Blockchains. The Wall Street Journal; 2017. http://deloitte.wsj.com/cio/2017/03/09/contracts-get-smarter-with-blockchains/.

Butts J. Forget Bitcoin, The blockchain Revolution Is Coming. NASDAQ; 2017. http://www.nasdaq.com/article/forget-bitcoin-the-blockchain-revolution-is-coming-cm862377.

Barlyn S. AIG teams with IBM to use blockchain for 'smart' insurance policy. Thomson Reuters; 2017. https://www.reuters.com/article/us-aig-blockchain-insurance/aig-teams-with-ibm-to-use-blockchain-for-smart-insurance-policy-idUSKBN1953CD.

48. Szabo N. 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.

J. Sabater, REGRET: A reputation model for gregarious societies. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.8.7554rep=rep1type=pdf

Guru: Universal Reputation Module for Distributed Consensus Protocols, A. Biryukov et al.

https://eprint.iacr.org/2017/671.pdf

P. Otte, ”Sybil-resistant trust mechanisms in distributed systems”

https://repository.tudelft.nl/islandora/object/uuid:17adc7bd-5c82-4ad5-b1c8-%20a8b85b23db1f/datastream/OBJ/z