Decentralized Finance in China:

Viable Solutions in Blockchain Infrastructures

by

# Yu Xu

An honors thesis submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Science

Business and Economics Honors Program

NYU Shanghai

May 2020

Professor Marti G. Subrahmanyam Professor Yannis Bakos

Professor Christina Wang

Professor Jens Leth Hougaard

Faculty Advisers Thesis Adviser

**Introduction**

In China, conducting financial activities with the prevalence of frauds and interference by third-party institutions brings about trust crises and mounts up transaction fees. Blockchain, a rising technology that claims to be capable of correcting these flaws has its trials in decentralized finance (De-Fi), which are then particularly meaningful to China. There are existing examples of De-Fi applications in which blockchain enhances trust by ensuring correct information with validation mechanisms, removes redundant intermediaries and pushes down transaction fees with smart contracts (e.g. smart derivatives, security token offering, decentralized crowd funding etc.). Section 1 of the paper provides a quick introduction to blockchain technology and its application in finance.

Section 2 of the paper examines a solution for financial institutions in China: Corda, a blockchain-inspired consortium ledger. Consortium blockchains only serve enterprises and are usually used to solve internal inefficiencies within institutions. With a restriction on accessibility and without the token feature of public blockchains, consortium blockchain projects avoid the complex logic in consensus mechanisms and token designs. As a benefit from fewer authorized users, consortium blockchains have greater scalability than public blockchains: e.g. Corda can reach TPS (transactions per second) around 1000-3000 while the public blockchain solution NEO in Section 3 has TPS around 33-1000. Besides better scalability, privacy in consortium ledgers outperforms that in public ones when confidentiality is an indispensable quality required in finance. In China, consortium blockchain projects are also less risky in compliance. The greater technical and legal feasibility with consortium blockchains encourages innovative financial institutions in China to land their De-Fi projects faster than startups focusing on public ones. We will look at a case using Corda to manage derivatives in Section 2.

Public blockchains are ones that serve the masses and have the most available use cases. Nevertheless, one of the biggest problems now with public blockchain in large-scale application is that transactions are time-consuming. NEO, the public blockchain representative in China that we will analyze in Section 3, can process around 33-1000 transactions per second, which is far from the average performance of centralized transaction systems in China: around 100,000-200,000 transactions per second.[[1]](#footnote-1) Current speed of NEO may be acceptable in use cases like VC funding where only accredited investors participate in the network from time to time. However, we need to find scaling solutions to support applications requiring high scalability like high frequency trading on a decentralized exchange. Another bad news is that there is an “impossible trinity”: public blockchains cannot achieve complete decentralization, high level of security and scalability at the same time. We need to find secure scaling solutions that attain a balance among the three extremes.

Most existing research regarding blockchain and finance suggests possible De-Fi applications without technical viability assessment on the underlying infrastructures.[[2]](#footnote-2) There are technical papers addressing specific cases like payments as well.[[3]](#footnote-3) But a newborn model simply based on lab experiments cannot guarantee success without a long-term history of real-life application. Because blockchain usually appears as a global network, most researchers fail to consider applications in a specific legal jurisdiction, neglecting the fact that countries have distinct attributes and one solution cannot fit all. These three gaps are where this research will fill in.

The paper aims at proposing infrastructure solutions to the underlying blockchain for decentralized finance in practice for both financial institutions and public finance in China. Two pioneering candidates are found to be promising: NEO, the Chinese version of Ethereum and the most prestigious public chain in China specializing in De-Fi, and Corda, a consortium ledger designed for financial institutions. Analysis of both projects and add-ons is performed on multiple aspects involving scalability, confidentiality, compliance feasibility, etc.

**Section 1 – Finance on Blockchain**

Literatures and empirical stories have witnessed flaws in Chinese financial market. Regarding stocks, according to “Bull In the China Market: The Gap Between Investor Expectations And Auditor Liability for Chinese Financial Statement Frauds”, the Chinese market pays much less attention to the quality of financial reporting than mature markets in developed countries like the US and has loose reporting standards. The auditing process in China is also less credible and transparent. The article also shows real examples on financial statement frauds leading to false valuation and unfair market behaviors.[[4]](#footnote-4) In terms of lending and borrowing, assorted centralized online platforms provide peer-to-peer lending services for small enterprises and individuals while claiming no responsibility when legal disputes occur. Frauds in lending platforms frequently happen in China.[[5]](#footnote-5) Financial derivatives are strictly managed by centralized financial institutions, causing inefficiency problems that we will address in Section 2. The lack of trust-worthy and transparent information is the root of the above issues. The good news is that the rising technology blockchain may serve as a cure for China.

A blockchain is a peer-to-peer network where non-trusting participants can securely transact with each other. It can be viewed as a public ledger consisting of a series of blocks, each containing some transactions. The ledger is collectively maintained by nodes, aka bookkeepers. Whenever a new transaction is initiated by a user paying the bookkeepers some transaction fees (usually lower than what is required by centralized organizations), it is broadcasted to all bookkeepers to be verified. After the bookkeepers reach consensus on the validity of the transaction after some mechanisms, the transaction will be added to a block if it is valid and the bookkeeper who accomplished the verification work will be rewarded by some incentive scheme with tokens. Each node keeps a copy of the whole ledger to maintain consistency and prevent cheating.

Use cases of blockchain are greatly expanded by smart contracts, which automatically execute some transactions or any types of operations when certain conditions are satisfied. It is more reliable than the traditional contracts signed by humans, as published code contains less chances of cheating or non-action. With smart contracts, we can then build functional applications on the blockchain just like the apps we use on our smart phone but using decentralized protocols. Applications on blockchains are called decentralized apps (Dapps). Financial Dapps are categorized as decentralized finance (De-Fi).

To replicate financial activities in the real world, a complete system of decentralized finance can be constructed as the following:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Product/Service** | Stable Coin | Issuance Platform | Decentralized Exchange (DEX) | Digital Payment | **Regulation Framework** |
| KYC/AML | Derivatives | Digital Wallet | Asset Management |
| De-Fund | Custody | De-Insurance | Credit & Lending Platform |
| **Protocol** | Digital Currency Protocol | Digital Asset Issuance Protocol | DEX Protocol | Wallet Protocol |
| Asset Management Protocol | Lending Protocol | Custody Protocol | De-Insurance Protocol |
| Fund Protocol |  |  |  |
| **Data** | Oracle | Privacy Protocol | Data Processing Protocol | Data Storage |
| **ID** | ID Protocol | ID Certification Parties | Name Service |  |
| **Infrastructure** | Consortium Blockchain/Public Blockchain/Cross Chain/Scaling Solutions (Settlement & Validation) | | | |

Table 1.1

The table conceives a systematic view of De-Fi, consisting of five layers. The fundamental layer is infrastructure responsible for transaction settlements and validation, which is the main focus of this paper. We propose potential solutions for businesses by evaluating the benefits, viability and risks of the infrastructure in the context of China by looking at two live cases using different types of blockchains: a consortium ledger used within enterprises in Section 2 and an open public ledger in Section 3.

**Section 2 – Consortium Solution: Corda**

This section discusses how the consortium ledger Corda can give impetus to De-Fi in China from the side of financial institutions. Compared to public blockchains like NEO, the subject studied in Section 3, consortium blockchains are much more legally feasible in China. As China banned all initial coin offerings on Sep 4th, 2017, when token/coin is a fundamental feature of public blockchains, consortium blockchains avoid the risk while preserving the function of a trust machine to the best degree.[[6]](#footnote-6)

In contrast with the bold attempts taken by blockchain startups of building brand new projects to deal with financial instruments, large financial institutions usually start their trials in existing credible projects so that they are backed by vendors who can be called to account whenever a technical risk occurs, as required by the Chinese government.[[7]](#footnote-7) Among assorted consortium ledgers facing enterprises, Corda stands out as being specially designed for financial activities.

To better fit with compliance requirements, Corda is more like a partially decentralized ledger than an authentic blockchain. There are three major differences between Corda and a blockchain that we will further examine later. First, for confidentiality, information on Corda is not fully visible by all users: there is no public ledger. Second, scalability of Corda is higher than a common blockchain because of its architecture design. Third, Corda is well attached with the real world because of its specific target applications while a generic blockchain is virtual.

A recent case of enterprise De-Fi in China is the SPLICE (structured products life cycle events management platform) project launched by UBS China using Corda as the infrastructure. Before blockchain and smart contracts come into play, the life cycle events management of structured products is done manually by officers from the Operations division in an investment bank. The process has been extremely inefficient as earnings on structured products are usually tied to some other assets, whose prices can change rapidly even over a short period of time. A heavy calculation on proceeds is thus required whenever a client inquiry comes in. Also, officers may not be able to track precisely every historical change or event so mistakes can happen from time to time and reexamination must be performed. Because of the huge costs and inefficiency generated by human work, the business has not been as lucrative as it ought to be. As a result, the management team specializing in financial technology saw potential in the consortium ledger Corda, which fulfills regulatory requirements in China, solves the inefficiency problem combined with external smart contracts and brings other benefits like operational transparency to clients. To be applied in business, the following assessments on Corda are performed.

1. UTXO model[[8]](#footnote-8)

Unlike the account-balance model of banks, Corda uses the UTXO model which directly tracks the original facts and states rather than the calculated balances. From the experience of Ethereum which also uses the account-balance model, logs on balances sometimes cannot be enough to explain how the states evolve. So, in accordance with Chinese financial institutions that are strict with original facts and documents, the model greatly prevents cheating in financial processes.

The UTXO model of Corda is more extended than that of Bitcoin. It not only traces value transfer, but also the status of non-value documents. All sorts of invoices, bills, certificates and orders in financial procedures can be captured.

Transaction in Corda is simply the process of a verified UTXO input becoming a valid UTXO output after going through corresponding business logics.

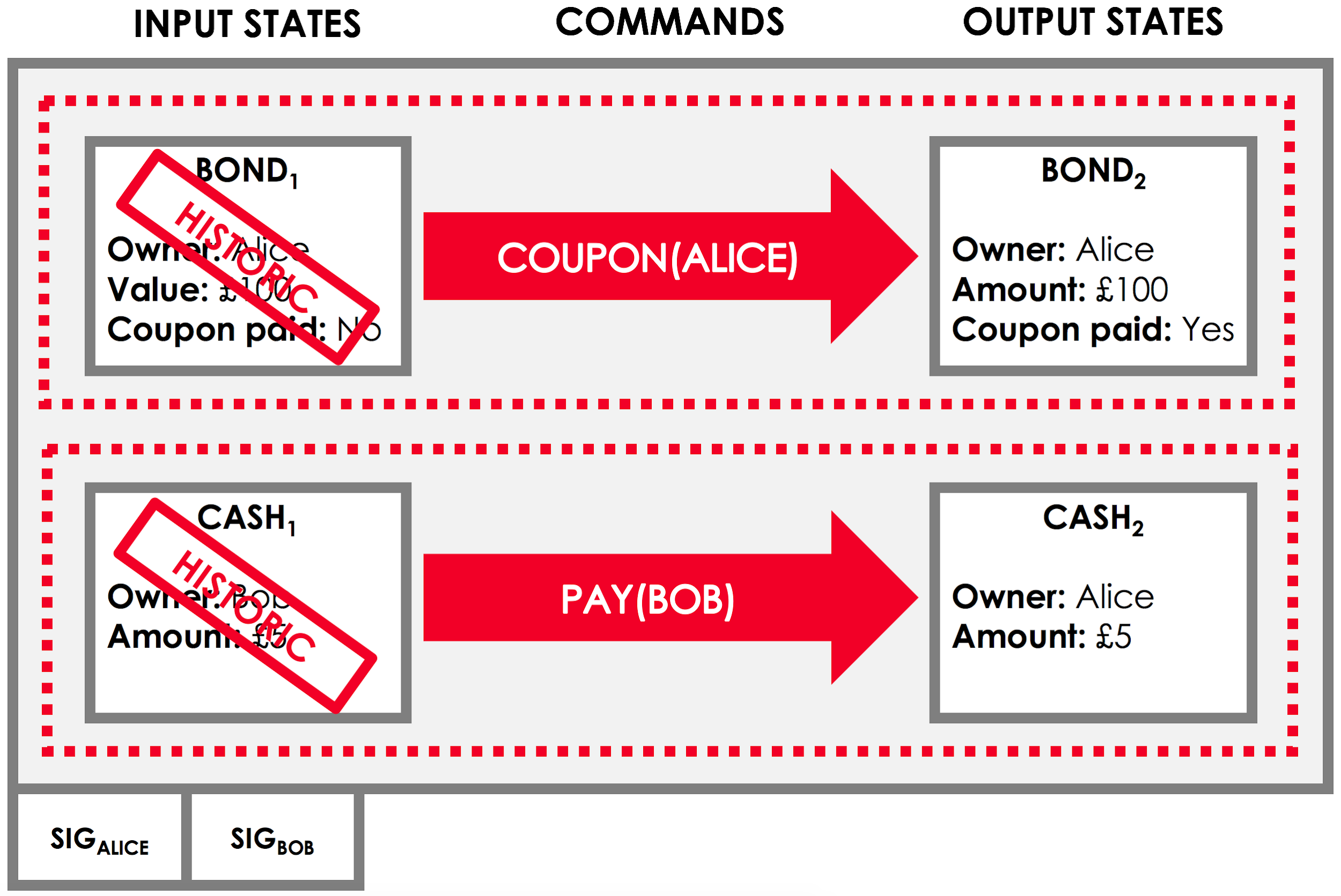
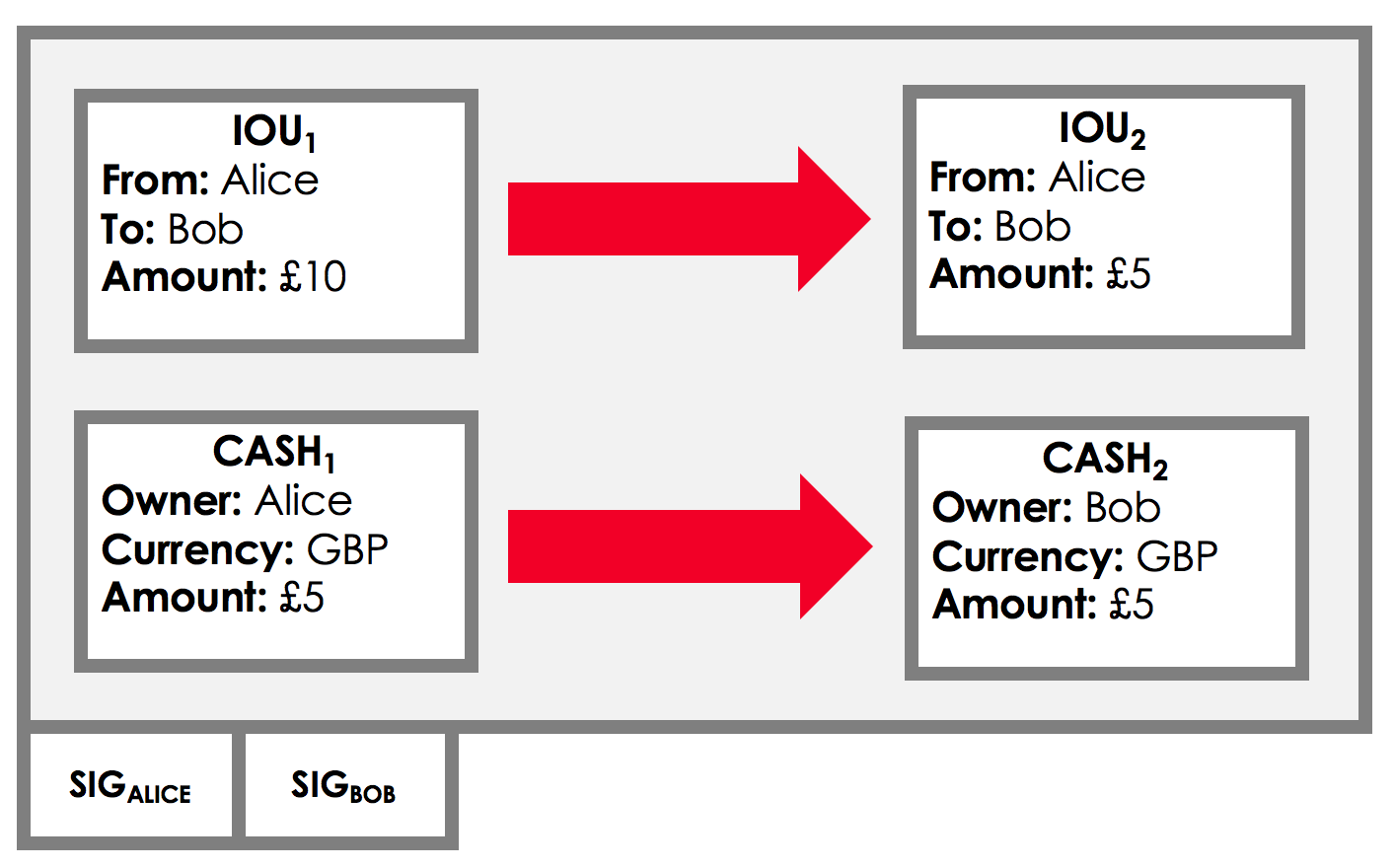
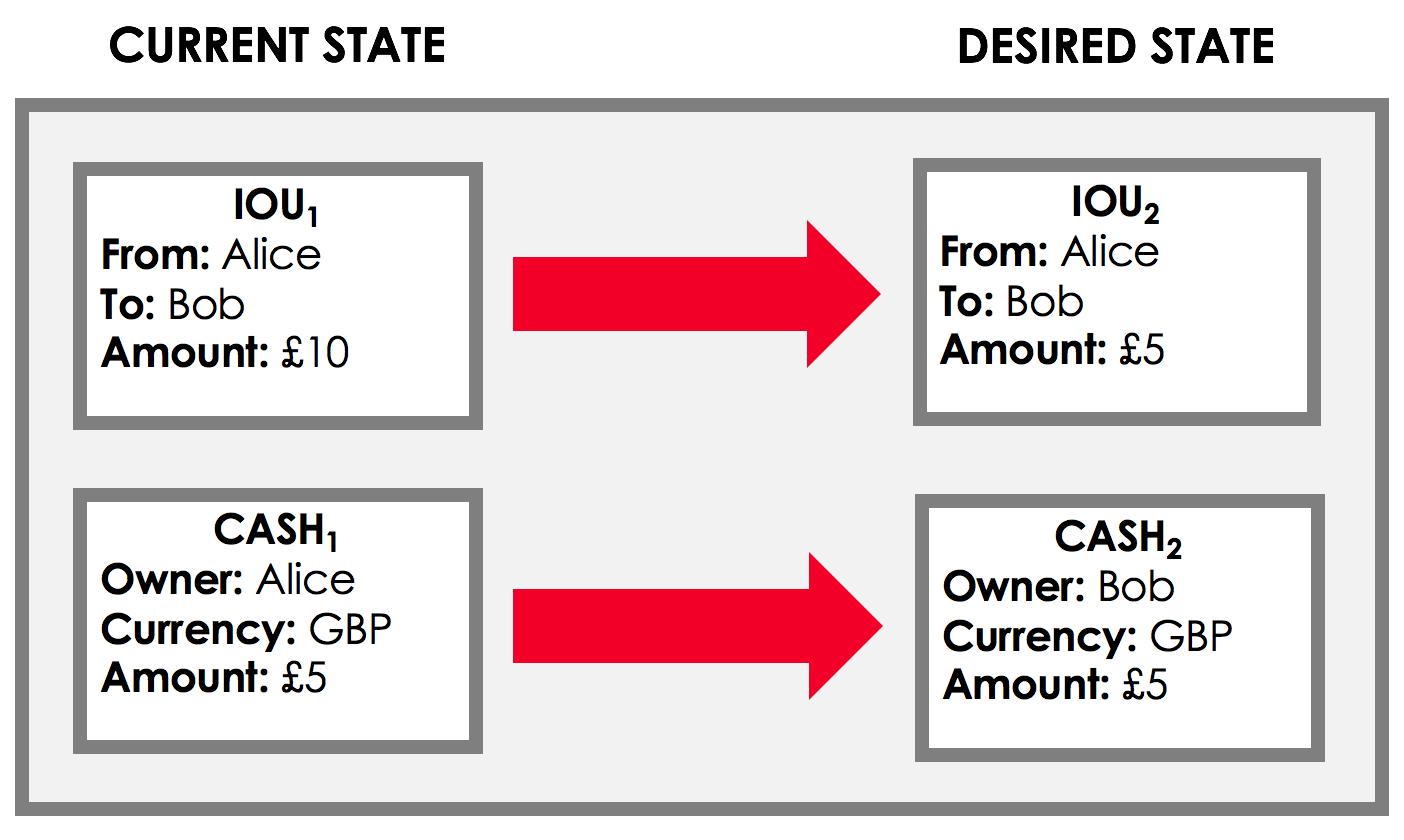


Figure 2.1 An Example of Transaction with UTXO Model in Corda[[9]](#footnote-9)

1. Graph (Directed Acyclic Graph, DAG)[[10]](#footnote-10)

Graph is the core of Corda with self-contained evidence. As there is no public ledger visible to everyone, graphs come into play instead. The evidence along the graph is organized as DAG, where the participants of a transaction can verify from the sink node by node up. In theory, to successfully verify a transaction, both direct and indirect evidence should be closely examined until the verifier reaches the original state.

Generic blockchains have a linear structure, where a specific timestamp can only be associated with one block. In contrast, DAG allows multiple chains to grow at their own pace as long as they are grown from previous nodes. DAG is also a key to high scalability in Corda as each node does not have to wait for consensus among all other nodes to proceed with new transactions. The throughput in Corda increases as well because DAG does not limit the amount of new data appended to the sink as routes work in parallel with each other. Another advantage of DAG is that transaction fees can be greatly reduced.[[11]](#footnote-11)

As the project is currently being tested with fewer than 10 clients, the scalability bottleneck has not appeared yet. Even though the number of clients grows, it can still be handled by the Enterprise version of Corda as the total number of clients of the investment bank is limited by Chinese regulatory departments with high entering bars on assets and social identities.

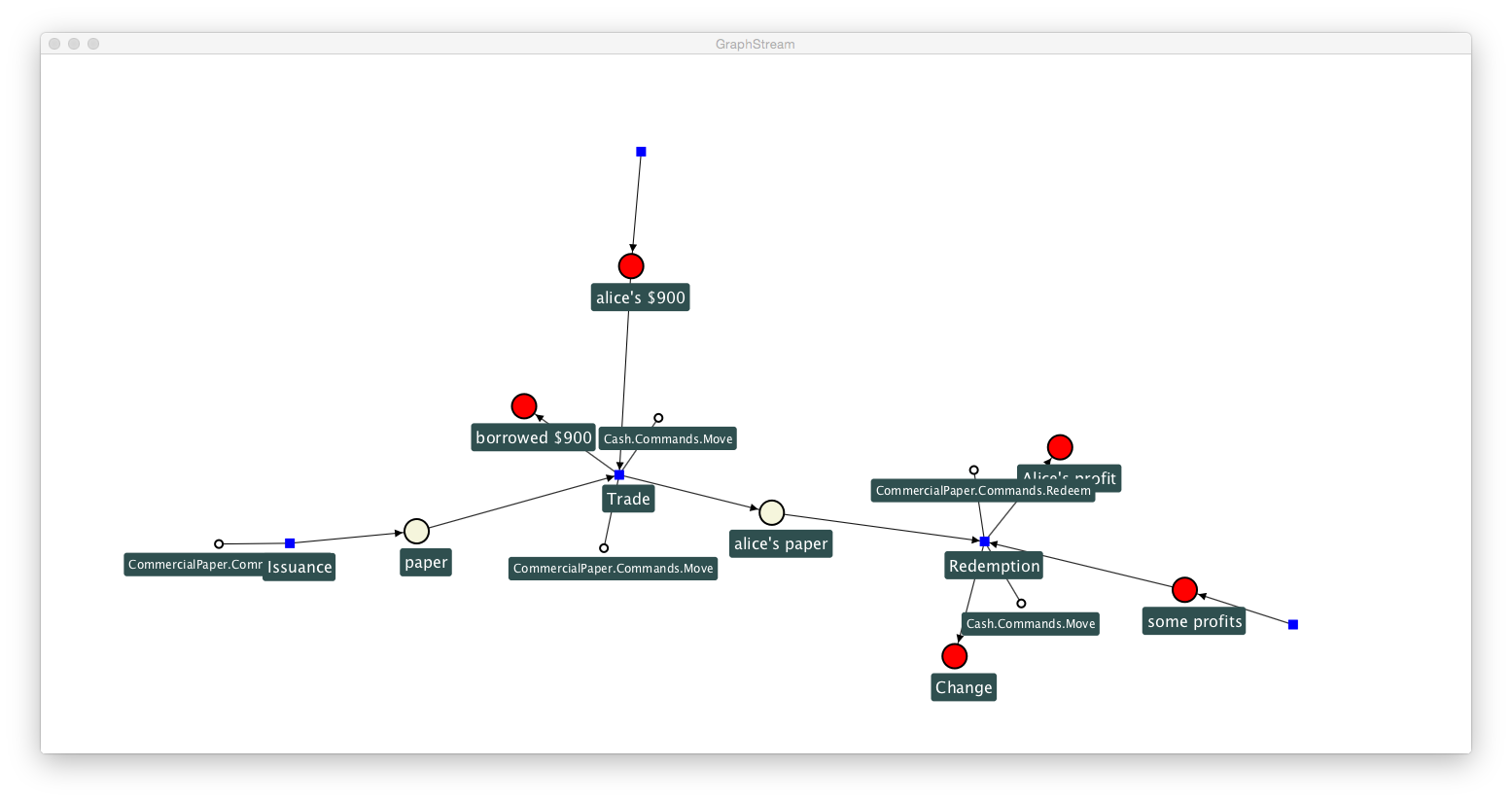


Figure 2.2 An Example of Graph in Corda[[12]](#footnote-12)

1. Notary[[13]](#footnote-13)

Notary serves as a third-party possessing authority to sign documents, equivalently speaking, a node possessing its own private key. It is an automatic program in most cases, with all its operations being transparent and regulated. The notary is selected and trusted by participants of a transaction, which means that the documents signed by the notary are binding on all parties within a transaction. The notary is not just one node or one person but a group of people or nodes reaching consensus using the PBFT(Practical Byzantine Fault Tolerance) mechanism. In sum, the notary in Corda is more reliable than traditional third-parties.



Figure 2.3 The Back End Structure of Notary Service in Corda[[14]](#footnote-14)

1. Smart Contract[[15]](#footnote-15)

The smart contract in Corda is essentially a verification function to assess whether the input state and output state in the UTXO model are valid. It is different from smart contracts in traditional blockchain in that it is not a complete program containing a series of transactions and transferring values. In contrast, the smart contract in Corda is just a small verification program called by transactions when necessary. Such design has the advantage that it eliminates the need to assess whether the smart contract is turing complete, contains endless loop or wastes computing resources of the virtual machine. However, if the institution has to run more complex business logic on Corda, it will need to bring in an external smart contract language to build on Corda. And the connection between the two has not been tested yet.

An extraordinary design about the smart contract in Corda is that it combines program code with texts of laws. Whenever an attack or a dispute occurs, the texts of laws have more direct legal force than the code.

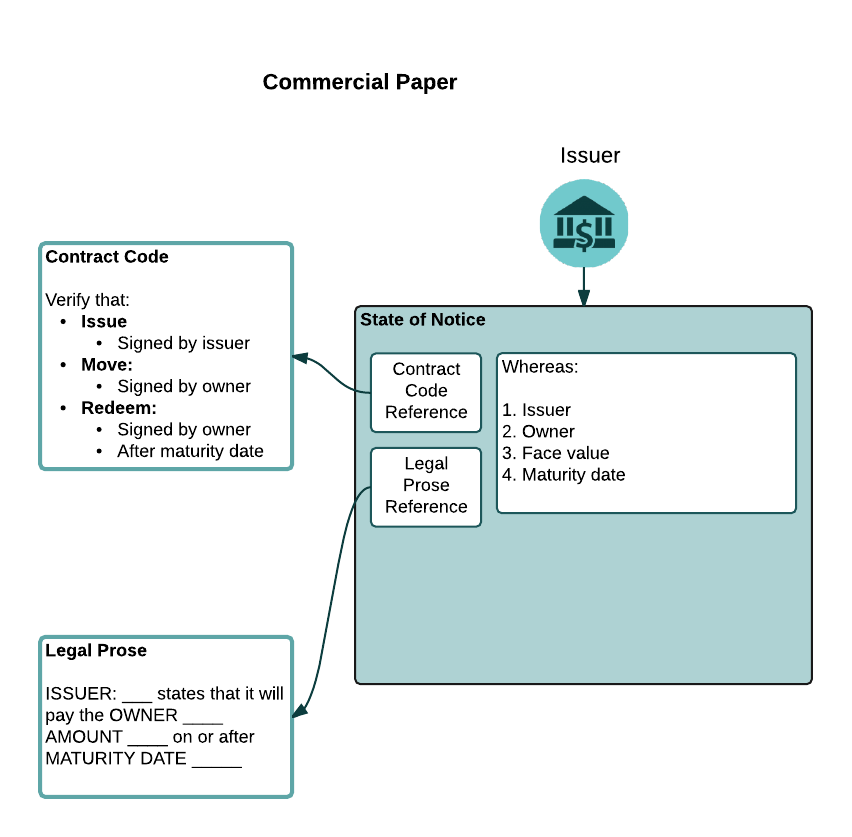


Figure 2.4 An Example of Smart Contract in Corda[[16]](#footnote-16)

1. Confidentiality[[17]](#footnote-17)

There are two major techniques in Corda to ensure confidentiality. The first is called blind signatures which tears off sensitive information. The technique hashes sensitive and insensitive texts distinctly. Then, it builds up a merkle tree in hierarchies and the tree is sufficient to verify insensitive messages after tearing off the sensitive ones. Once a dispute occurs, if the teared sensitive messages were faked, the merkle tree can be used to discern the authenticity of the information.

The other technique is called composite signatures using perceptron models. It entitles multiple parties from an assigned group to sign and sets a threshold for the weighted sum of signatures. Once the weighted sum exceeds the threshold, the composite signature becomes effective.

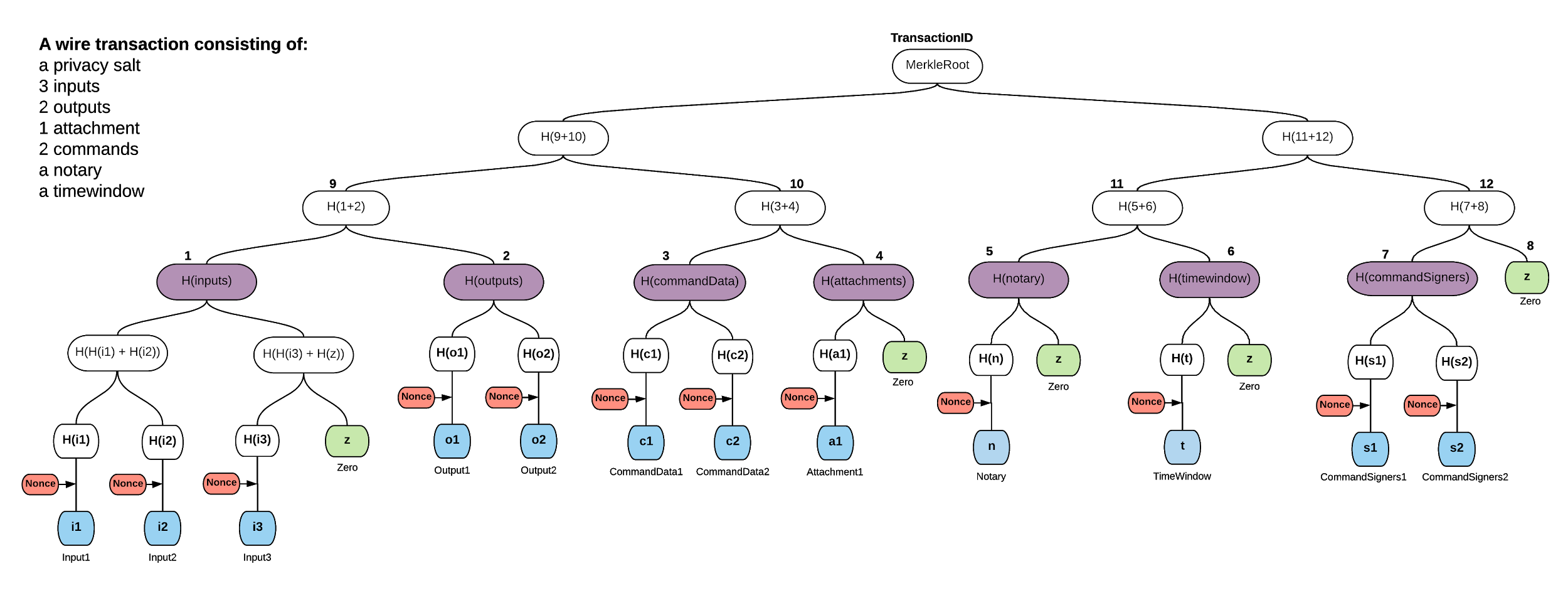


Figure 2.5 An Example of A Merkle Tree with Tear-off in Corda[[18]](#footnote-18)

**Section 3 – Public Solution: NEO & Scaling Add-ons**

In contrast with the consortium ledger in Section 2 which is specially designed for enterprise interest, this section studies infrastructure for De-Fi in China developed by startups that care about the masses and launch their products on public blockchains. NEO will be the main subject of discussion, as being the most long-lived and government-backed public blockchain in China.

With the company entity based in Shanghai, NEO has finished its ICO early in 2016 before the ban. The great upsurge in heterogeneous public-chain projects dramatically drops after the ban and NEO stands out as a fortunate survivor with sound technology, a reliable team and promising goals in De-Fi. It soon gains support from the government of Yangpu district, Shanghai, as the city is pushing forward innovative high-tech startups. To avoid further compliance issues, NEO brands itself as a non-for-profit community, sponsoring assorted decentralized applications in its ecosystem using the fund it raised during the ICO. De-Fi is one of the main fields that the team has been devoted to for the recent years.

In terms of detailed technology, NEO has the most suitable consensus mechanism for financial applications compared to those of other public blockchains. There are several prevailing consensus mechanisms we observe: proof-of-work (POW) used by Bitcoin and Ethereum, proof-of-stake (POS) which will be used by Ethereum in the near future for its scaling project Casper and delegated proof-of-stake (DPOS) used by EOS. A mechanism called Delegated Byzantine Fault Tolerance (DBFT) is used by NEO. It’s easy to see why DBFT is the most suitable algorithm for finance from comparisons among these mechanisms.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Intro to the mechanism** | **Pros** | **Cons** |
| **POW**[[19]](#footnote-19) | People who possess more computing power have more chances to become the next bookkeeper. | 1. Easy to implement. 2. Minimum information exchange among bookkeeping nodes. 3. Difficult to hack. | 1. Waste of energy. 2. Hard to cut down validation time of transactions. 3. Risk in bifurcation. 4. Poor finality. |
| **POS**[[20]](#footnote-20) | People who possess more stake in the network have better chances to become the next bookkeeper. | Similar to POW but more energy-saving. | 1. Selected bookkeepers may not be specialized. 2. Risk in bifurcation. 3. Poor finality. |
| **DPOS**[[21]](#footnote-21) | A modified POS. First, elect bookkeepers based on the stake they possess. Then, let the bookkeepers do their job in turns. | Similar to POS but selected bookkeepers are specialized. | The problem of poor finality is not solved either. |
| **DBFT**[[22]](#footnote-22) | First, elect bookkeepers based on the stake they possess. Then, the bookkeepers achieve consensus by an algorithm called Byzantine Fault Tolerance. | 1. Specialized bookkeepers. 2. High level of fault tolerance. 3. Bookkeeping is done collaboratively by several nodes. Every block has finality and there’s no chance of bifurcating. |  |

(Finality means how long a transaction can be “finally” confirmed.)

Table 3.1

To sum up, the core benefit of DBFT is to assure finality of the system, which is required in financial activities as users cannot afford to wait endlessly for their transactions to be processed. Furthermore, with the help of NEO’s smart contracts and token economy, users can tokenize their assets and securities from the real world, transfer and trade, register and issue new assets, finish clearing and delivery through the peer-to-peer network of NEO. It is worth mentioning that security tokens can also be issued on NEO in compliance with China Contract Law and China Corporate Law.

However, though the features of NEO make it intrinsically suitable for financial use cases, other aspects commonly found in public-chains can hinder it from mass application. The biggest problem now with public-chains is that on-chain transactions are costly, in terms of time. Bitcoin and Ethereum - global representatives for public blockchains - have low speed (TPS, stands for transactions per second). Bitcoin has only 7 and Ethereum has 15. The TPS of NEO is much better because of its efficient consensus algorithm, around 33-1000. However, centralized transaction systems now being commercially used have extraordinary performance on speed - e.g. Taobao on 11/11 has more than 200,000TPS overnight. We can see reasons for the low scalability from a comparison between two types of network, taking POW as an example.

For a traditional public blockchain using POW consensus mechanism, to confirm a transaction, the steps are listed as below:

1. A wallet node broadcasts a transaction to the network
2. A mining node includes the transaction into a block, solving some math problems using its computing power (POW) and then broadcasts the block to the network
3. Other mining nodes see the block, validate the work done by the miner, add the block to their own chain and confirm the transaction
4. To prevent bifurcating, need to wait for several nodes’ confirmation

For a centralized network to confirm a transaction:

1. A wallet node initializes a transaction
2. The server receives the transaction and confirms it

Even if NEO uses another consensus mechanism which is more speedy than POW, it still requires some validation work similar to the aforementioned. We can see an apparent discrepancy in time used to approve a transaction. A way to diminish such discrepancy is to apply scaling solutions on public blockchains, which enhances throughput as well as speed. Generally speaking, there are two directions to go towards: layer1 (do something on chain) and layer2 (do something off chain). The research below is then done for NEO to expand its capability to process transactions faster. Most experiments have currently been done on Ethereum because of its wider application, from which NEO can draw lessons as the two blockchains share similar design.

Layer1:

1. Vertical scaling: Aiming at expanding the size of each block on chain and requiring each node in the ecosystem to do more work. The major problem with vertical scaling is that as it requires a larger workload on each node, the majority of people cannot accomplish the computing task because their computers are not powerful enough. Vertical scaling may reduce effective nodes in an ecosystem and strengthen the idea of centralized mining. An example: BCH.
2. Horizontal scaling: To process more transactions through adding more nodes into the network. The major problem is that it cannot really bring up the speed as every transaction in a blockchain still needs to go through every node in the system.
3. Consensus mechanism modification: Some oversea public chains try to sacrifice security and decentralization for higher speed, like EOS. EOS only has 21 effective nodes so it can process hundreds of transactions per second. However, this modification might distort the original intention of a public blockchain.
4. Sharding: Sharding is a term derived from a database system. It means horizontally cutting a large database into smaller and faster partitions (i.e. shards) that are easier to manage. Currently, sharding is the most promising solution in layer1 set. There are existing examples that can work in theory, e.g. OmniLedger, Ethereum Casper. We will see in later paragraphs.

Overall, layer1 expansion mainly uses the method of base-scaling, which is not a good idea as very few projects are proved to succeed. In contrast, the idea of layer2 solutions is more promising. By building a protocol on the blockchain, through which we can do some costly computational work off chain at a faster rate and go back to the chain when validation is needed. The best example may be TrueBit, a project that we can have security, decentralization and speed assured at the same time. Major cases of layer2 projects include sub-chains(Loom, Plasma), state channels(Lightning Network, Liquidity Network, Trinity Network) and protocols(TrueBit).

Layer1:

1. Sharding
2. Ethereum Casper[[23]](#footnote-23)

Casper is a mechanism that is used to address security issues of sharding for Ethereum. To accomplish sharding on Ethereum, its POW consensus mechanism needs to be modified to POS as miners (the bookkeepers) can hack the network with much less computing power when the huge ledger is partitioned. As for NEO, the consensus mechanism may not be a problem as the specialized bookkeepers are already elected based on the stake they hold and do not leverage on their computing power.

Here is how Caper will work to ensure on-chain security after sharding:

1. A bookkeeper has to deposit some fraction of ether as earnest money. When he discovers a new block, he places bets to validate the block.
2. If the block is successfully added to the blockchain, the bookkeeper will get rewards that are proportional to his bets.
3. If the bookkeeper tries to be a malefactor, all the deposits will be confiscated.
4. OmniLedger[[24]](#footnote-24)

OmniLedger is the first project that can compete with the scalability performance of the centralized transaction system Visa. The goal of OmniLedger is to attain a balance in the “impossible trinity” - ensuring sufficient scalability while keeping necessary degrees of security and decentralization.

It uses a bias-resistant randomness protocol to select large shards that are statistically representative as bookkeepers. Regarding transactions involving multiple shards, OmniLedger introduces an efficient cross-shard commit protocol to ensure security and correctness. Meanwhile, to optimize performance, OmniLedger executes concurrent transactions within shards. By state blocks with collective signature and a “trust-but-verify” method to validate penny antes with low latency, the size of ledger is significantly cut down.

OmniLedger uses RoundHound and Verifiable Random Function (VRF) protocols to randomly distribute validators to different shards. The consensus mechanism among shards is PBFT (Practical Byzantine Fault Tolerance), which can tolerate less than ⅓ nodes being malefactors. In lab experiments, TPS of OmniLedger increases linearly with the increase of shards: 5850 TPS with 16 shards, 13000TPS with 25 shards.

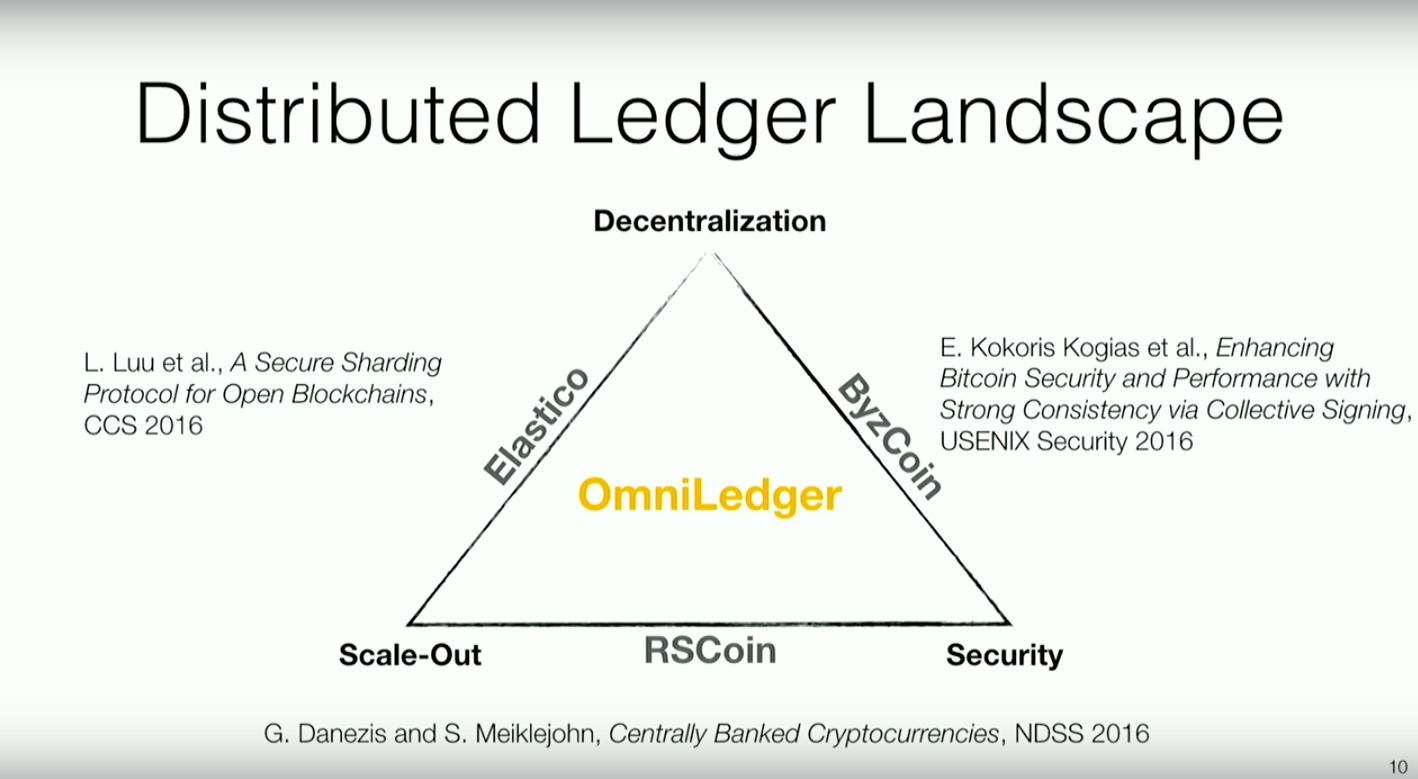


Figure 3.1 The Design of OmniLedger to Balance the Impossible Trinity[[25]](#footnote-25)

Layer2:

1. State Channel
   1. Lightning Network (for Bitcoin)[[26]](#footnote-26) / Raiden Network (for Ethereum)[[27]](#footnote-27)

State channel is a layer2 technique to execute transactions or update states, such as changes within a smart contract. The concept of state channel was first raised by Jeff Coleman in 2015. It has already been realized for Bitcoin as Lightning Network and for Ethereum as Raiden Network. Though the channel operates off chain, the reliability and completeness of events happening in the channel can be assured: if any dispute, there are chances to trace back to the blockchain and verify the truth.

Lightning/Raiden Network is convenient to use for two parties whose amount of transactions is less than their on-chain deposits and trade frequently with each other. It is especially helpful when the users have to update their states continually. Also, it ensures confidentiality very well as everything just happens within the channel but not broadcast to the whole blockchain. It has good finality as the state channel can be closed immediately once the two parties reach consensus on a state update.

However, a state channel has limitations as well. It provides poor liquidity as users must lock some funds on-chain to open a channel.

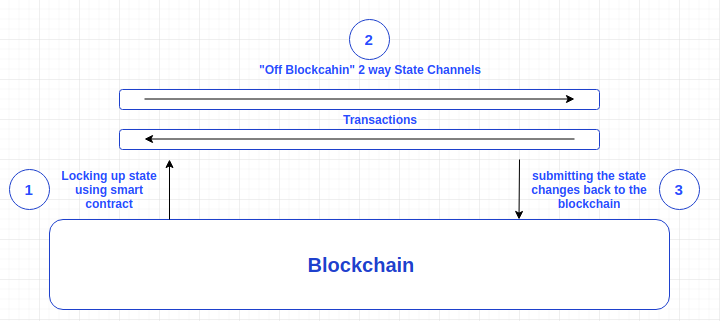


Figure 3.2 An Example of How Status Channel Works[[28]](#footnote-28)

* 1. Trinity Network (for NEO)[[29]](#footnote-29)

Trinity Network is the NEO version of Lightning/Raiden Network. The only difference is that Trinity Network only works for NEP-5 tokens within NEO network, while Lightning/Raiden Network works for ERC-20 tokens.

* 1. Liquidity Network[[30]](#footnote-30)

Liquidity Network solves drawbacks from traditional state channels:

Liquidity Network is free and allows multiple parties to transact. State channels can only be established between two parties and each channel requires deposits. If the number of users is large, there will be a huge amount of funds locked thus leading to poor asset liquidity. In contrast, users do not have to lock any deposits on Liquidity Network and transactions are not limited to only two parties.

Opening a new payment channel on Liquidity Network does not require a new smart contract, thus avoiding running repeated contracts on the root blockchain. It is also not necessary for users to establish a smart contract on the blockchain when they can directly deposit tokens from their off-chain wallets on Liquidity.

The technology behind Liquidity Network is called Revive, which was first raised by its founder in 2017. It is a much simpler routing algorithm based on “hub” network structure.The security of transactions and assets is assured by the combination of on-chain smart contract and off-chain signature packages. However, as the technology Revive does not belong to standard state channels, the rebalancing process within channels is not transparent enough and requires regulations. Also, Revive can only be applied in network topology with cyclic structure.

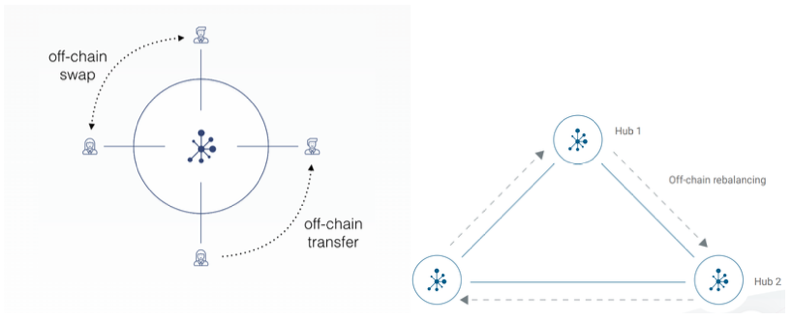
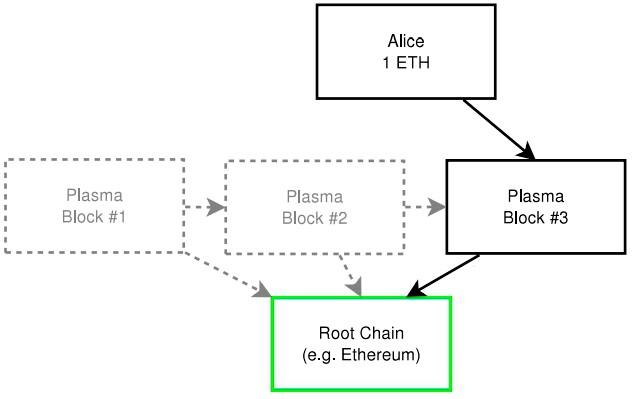


Figure 3.3 The Hub Structure of Liquidity Network[[31]](#footnote-31)

1. Sub-chains:
   1. Plasma[[32]](#footnote-32)

Plasma is an off-chain scaling solution. Plasma achieves higher scalability by connecting to the root blockchain, which can also derive its own sub-chains recursively. Complex computations and applications can then be operated on sub-chains, keeping minimum interaction with the root chain. Consequently, Plasma sub-chains can be much more speedy and less expensive as we do not need to replicate every operation on the root chain.

A significant difference between Plasma and state channels is that when parties of a channel reach consensus, assets can be withdrawn from the channel instantly. When Alice and Bob simultaneously close the channel, they can withdraw their money immediately as long as they reach consensus on the final status. However, it is impossible on Plasma sub-chains: users must wait for a questioning period before they can withdraw their assets. State channels are even faster and cheaper than sub-chains, which means that we can easily establish state channels on sub-chains to further optimize the performance. For example, a penny ante between two users in an application.



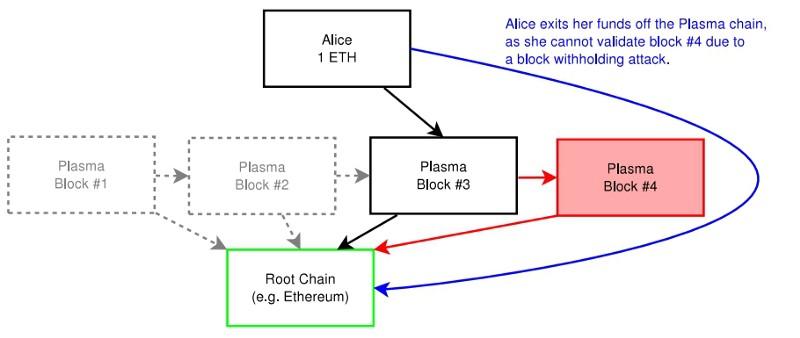


Figure 3.4 An Example of Plasma on Ethereum[[33]](#footnote-33)

1. Celer Network[[34]](#footnote-34)

Celer Network is inspired by the hierarchical abstraction of the traditional Internet. It breaks the complicated network into simple layers, each functions differently. Such architecture reduces the complexity of system design, development and maintenance. It includes the following:

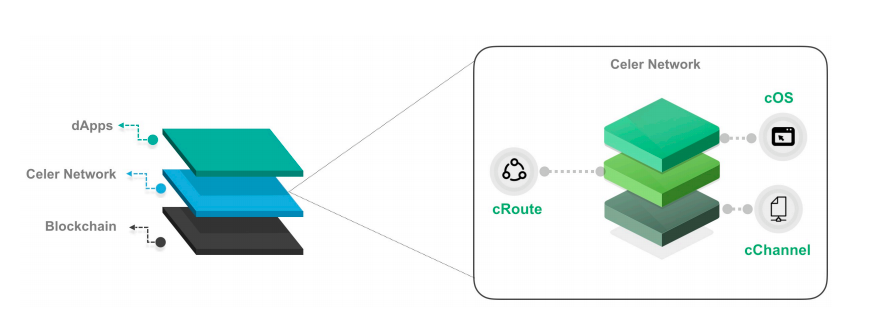


Figure 3.5 The Design of Celer Network[[35]](#footnote-35)

cChannel connects with the main blockchain, which can be seen as a state channel and sub-chain toolkit. It relies on DAG (direct acyclic graphs) to reach rapid status turnover, expanding liquidity to the maximum extent.

cRoute is a scalable and efficient routing algorithm. Most other routing algorithms use Dijkstra's shortest path first algorithm. When a large amount of assets move around, the network topology can be so unstable that the algorithm may not converge. Celer Network uses gradient back pressure routing algorithm, computing with gradients. It takes the balance of off-chain network channels into consideration, stabilizing the network topology even with a large number of transactions going on. Compared to Revive used by Liquidity Network, the rebalancing process in Celer Network is much more transparent as ensured by their distributed balanced routing (DBR). Currently, Celer Network has been applied to 77 nodes and 254 state channels. cRoute raises the network throughput 15 times larger than the state-of-the-art solutions.

cOS provides developers with SDK, helping them avoid the complicated logics regarding the interaction between on and off chain. It offers a common design setting and deals with off-chain states, operations, storage, tracing, and disputes. With the friendly cOS, developers can focus on decentralized applications.

With the root blockchain ensures trust, confidentiality and decentralization, the TPS of running De-Fi applications on Celer Network can reach billions. For terminal users, connecting to Celer Network is just like surfing on the Internet, where they can interact with applications like Google and Facebook.

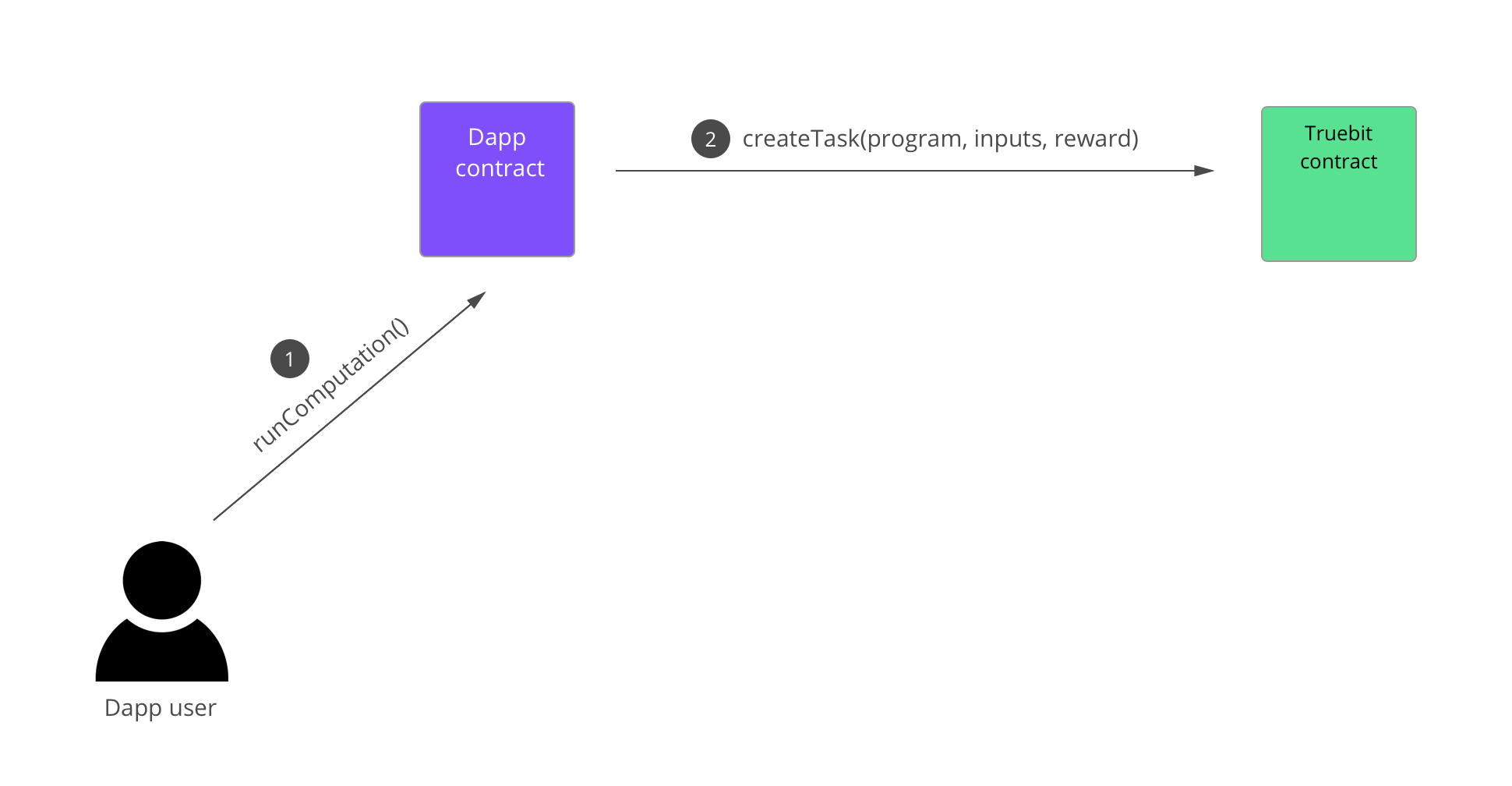
A fun fact about Celer Network is that it has a unique token economy model. It assigns Network Liquidity Bakers (NLB) as a role that provides overall liquidity. Through a mining like approach, Celer provides token CELR as an incentive scheme to NLB. Off-chain Service Provider (OSP) offers state channels, requiring transaction fees based on auctions. Off-chain state monitoring is done by Service Guard (SG) which maintains the state after the user is offline. Users have to deposit CELR as an incentive scheme to SG.

1. Off-chain Computation
   1. TrueBit[[36]](#footnote-36)[[37]](#footnote-37)

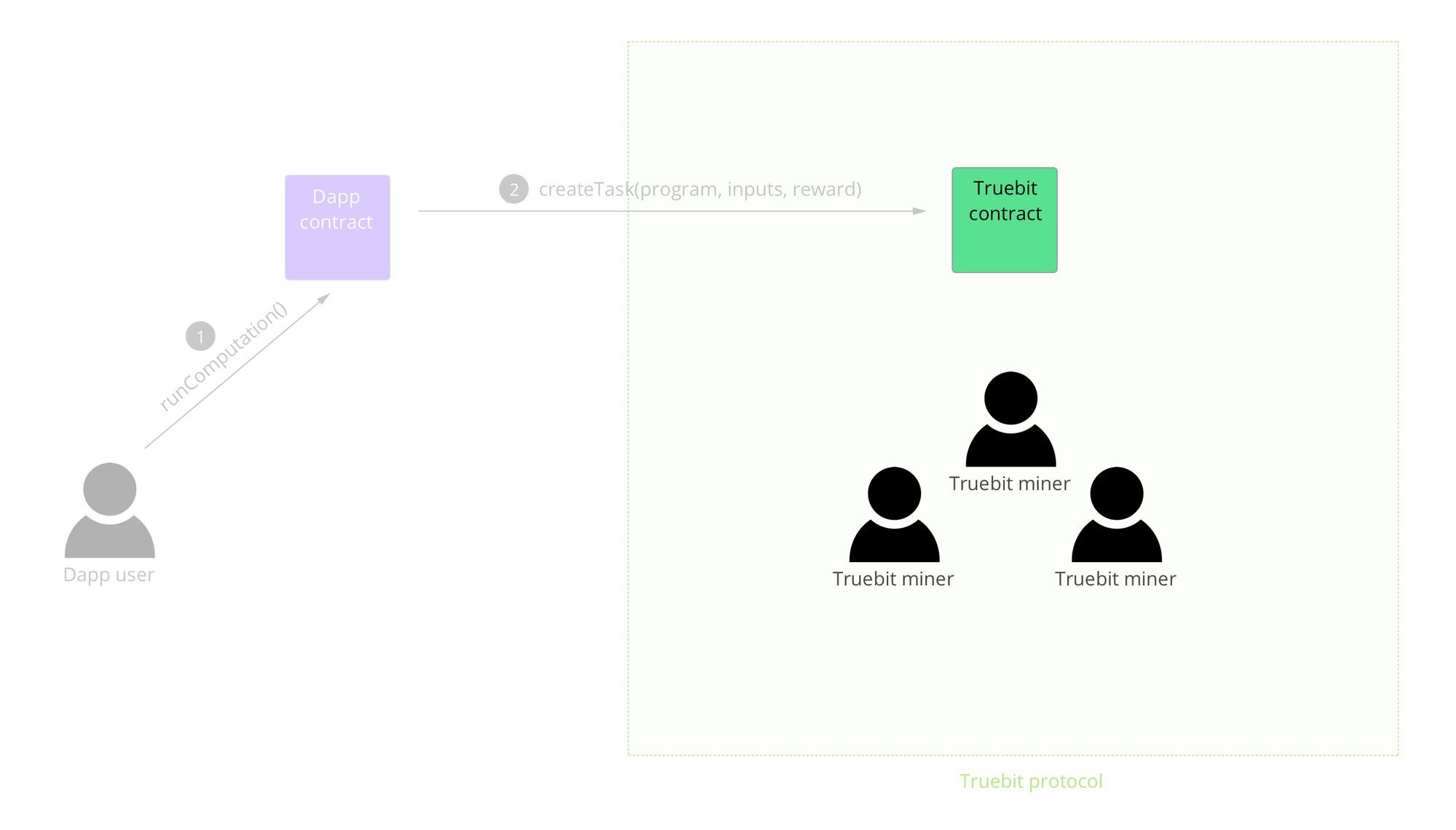
When state channels and sub-chains aim at raising throughput, Truebit helps a public blockchain solve complex computations off-chain. In a public blockchain, there is a limitation on how many operations a block can contain. So Truebit transfers some of the operations requiring huge computing power off-chain.

When a De-Fi application wants to carry out a complicated computation that is limited by the blockchain, it goes to Truebit Protocol rather than the root blockchain.

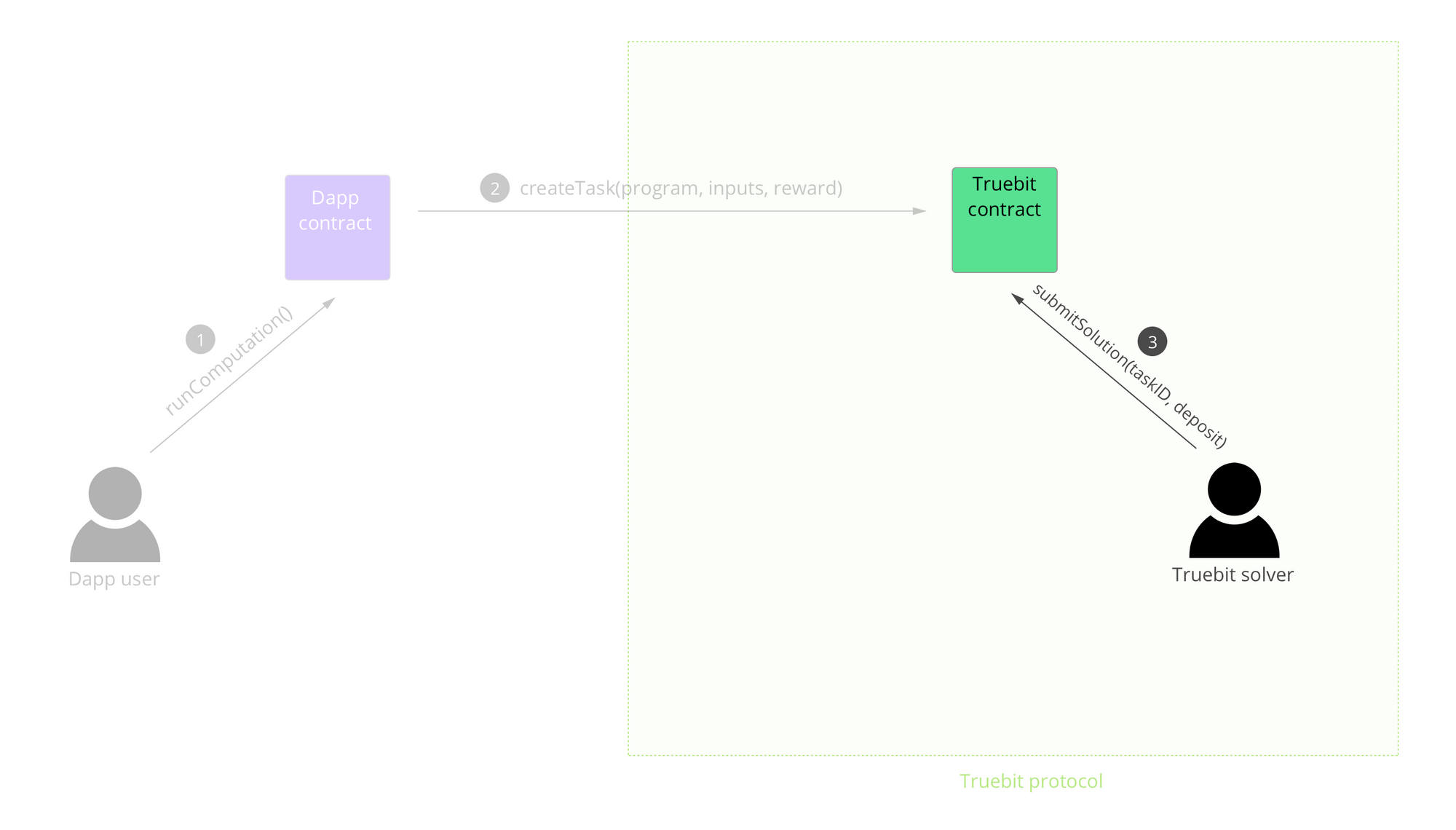
The API to interact with Truebit Protocol is simply a function called “createTask” in Truebit smart contracts. By running WebAssembly virtual machines, De-Fi applications can directly send their programs’ bytecode and inputs. As being task givers, users have to attach rewards to their tasks as well. Everyone is able to join Truebit for free and gets rewards for solving computational tasks. The miners in Truebit will monitor events on Truebit contracts. Once a task is given, a Truebit miner will download the code, use his local WebAssembly VM to compute the result and submit it to the contract. The solver (miner) has to deposit as well to ensure his responsibility for his results. Then, a Truebit Verifier (Challenger) can submit his deposits and challenge the results within a certain period of time.



A Task is Created



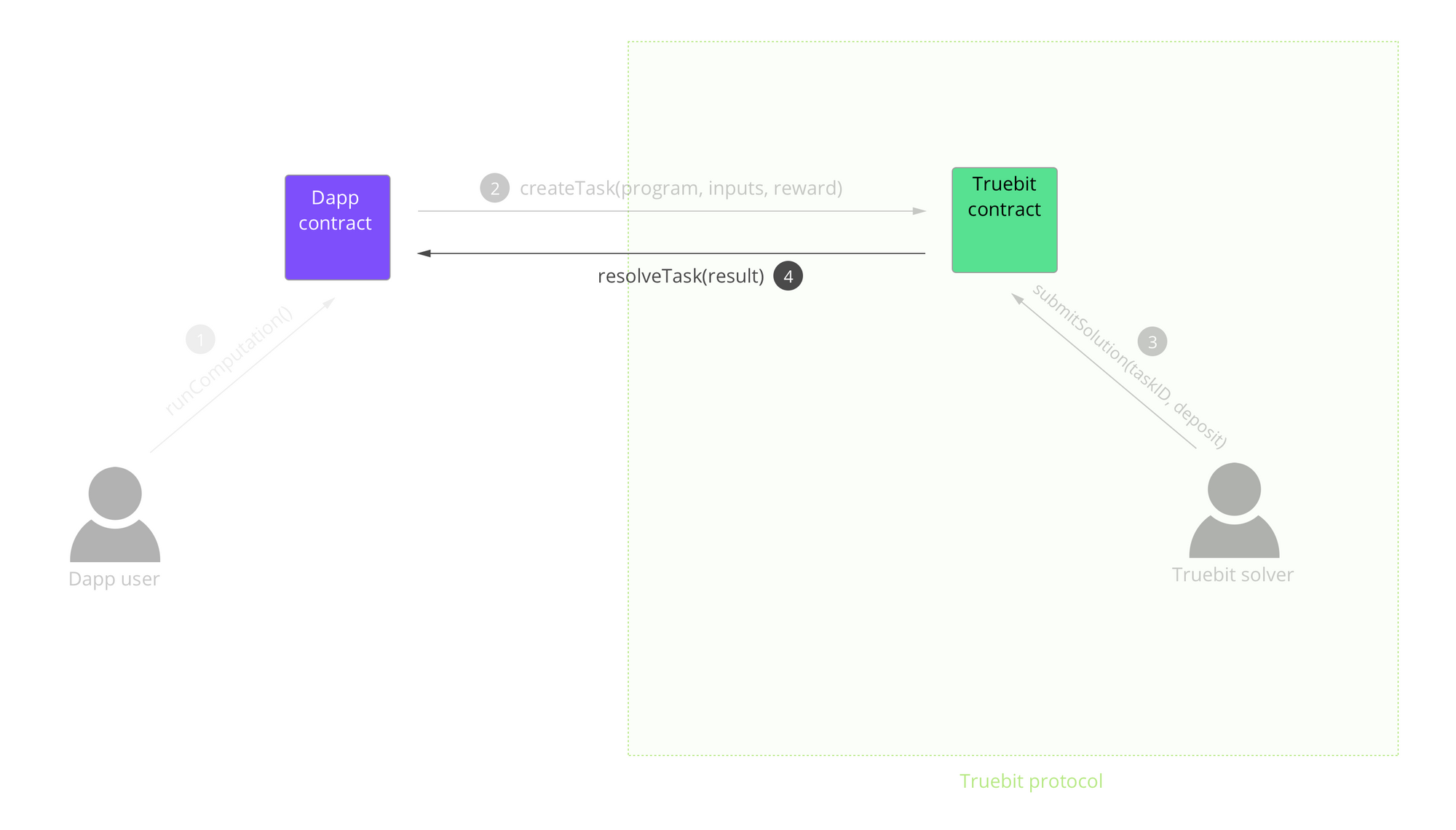
Miners/Solvers Monitor the Tasks



A Solver Submits an Answer

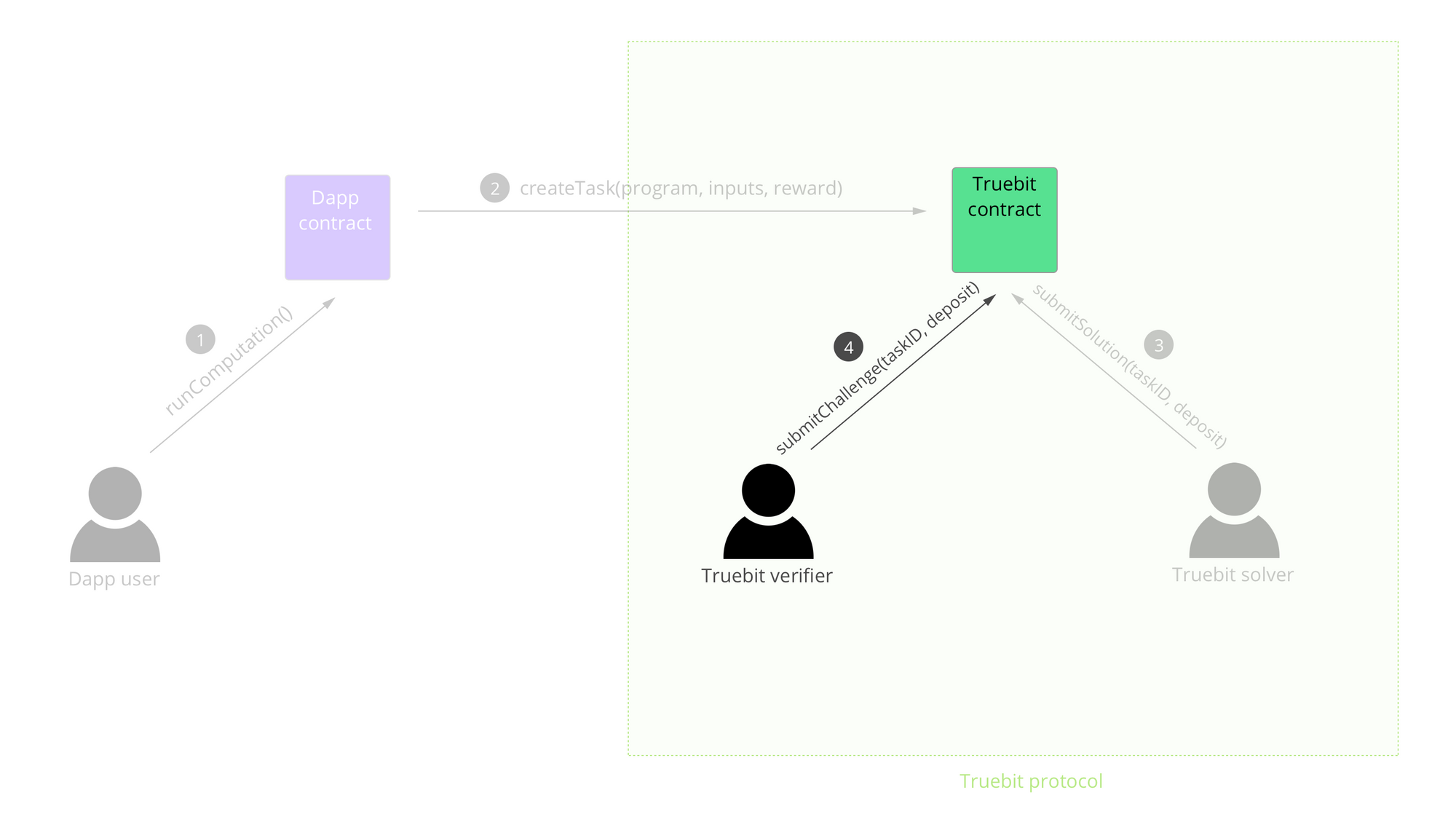
Case I: No challenger

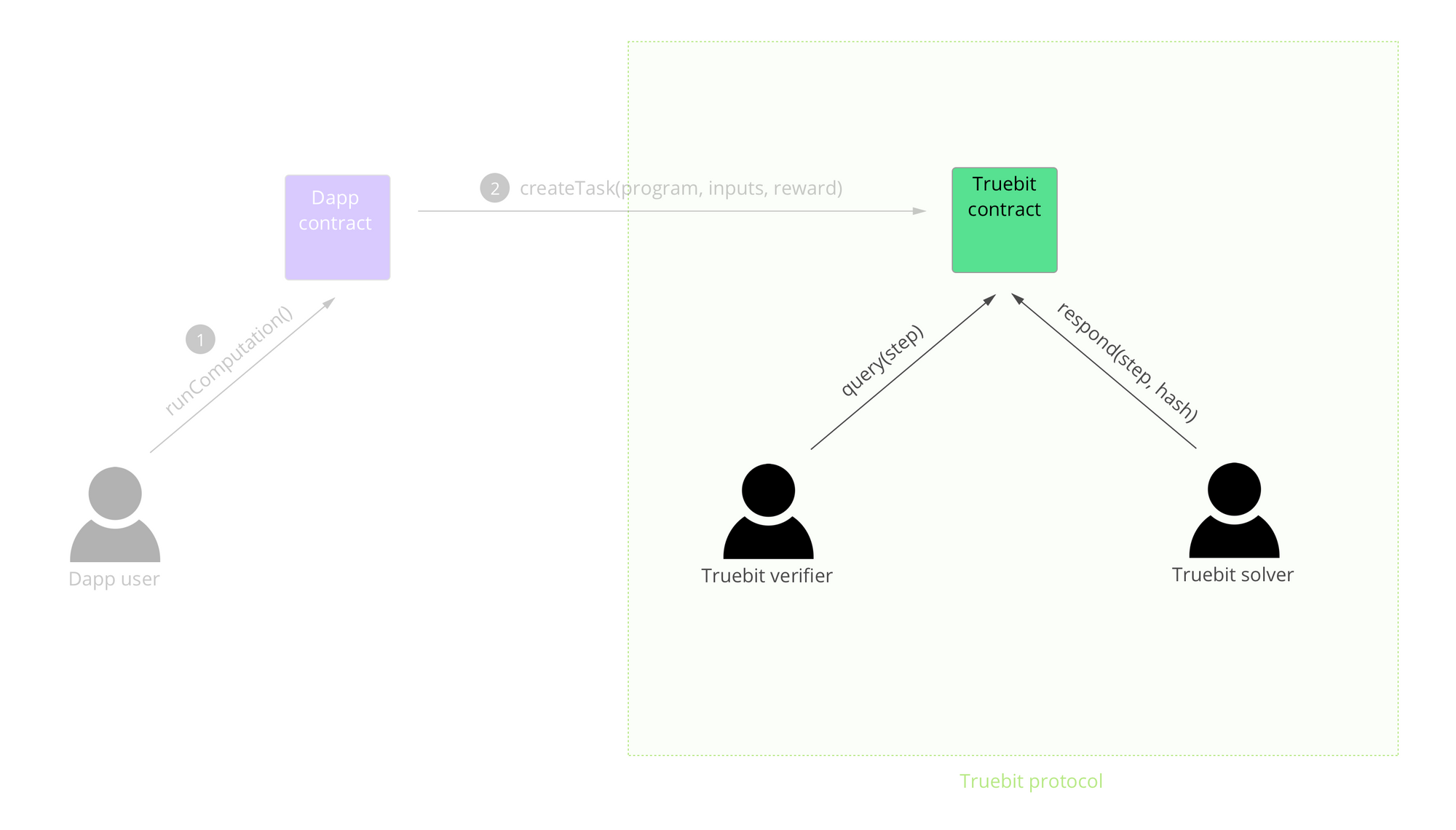
Consequence: Solver’s answer will be directly sent back to the application.



Case II: There are challengers.

Consequence: The challenger submits his deposits and challenges the current answer given by the solver. Now, the contract locks both the rewards given by the task giver and the deposits submitted by the solver and the challenger. The verification game starts to tell who is correct.

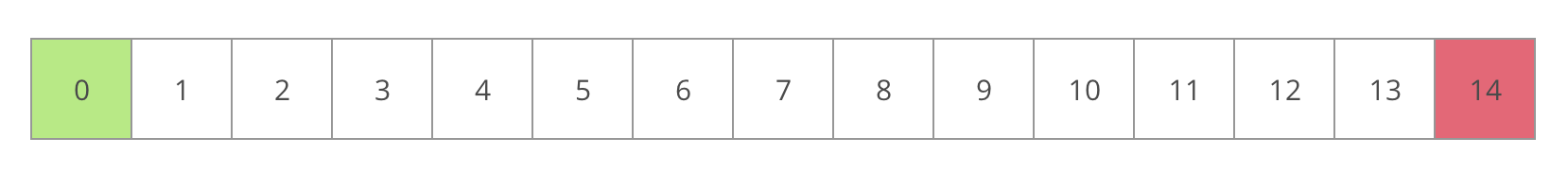




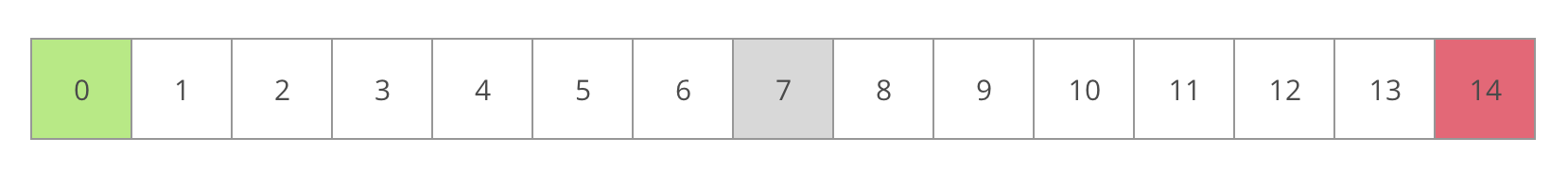
The Verification Game:

The game uses a merkle tree as data structure. On every node of the turing instruction set, the solver and the challenger plays the game with binary search. When the merkle of memory and state among computation steps cannot increase any more, the game can tell who is cheating. The contract of the game runs on the blockchain, requiring merkle hash at a certain point of time then updates a state to check whether the derived merkle hash corresponds with the required one.

When the solver gives a conclusion and the verifier challenges it, the initial state between the two entities is 0. They will start a local VM to run the task respectively. Assume that we see different results after the first round to the end of the program (after running 14 instructions). This means that the solver and the verifier diverge on state 14.

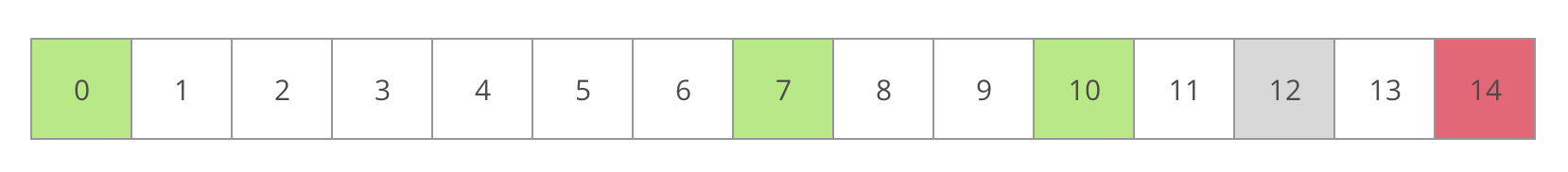


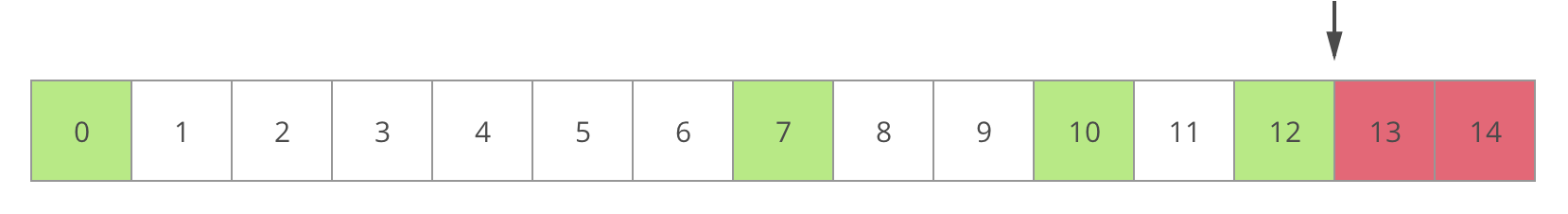
Then, the verifier checks the midpoint of the program at state 7.



The solver can use the WebAssembly VM from Truebit to compute his own hash value as a merkle tree root and submit it to the smart contract. Then, the verifier computes his own merkle tree root as well and compares it with that from the solver’s. If two roots are the same, it suggests that the dispute occurs in the second half of the instructions. Otherwise, the dispute is in the first half.

For example, we now assume that the roots are the same at state 7. The verifier then submits a query asking the solver to compute the merkle tree root again at the midpoint of the second half (state 10) and compares as before. As this keeps on going, the solver will be forced to find where the dispute occurs.





Now, the computational work is narrowed down to the point of dispute, which becomes small enough to be solved on the blockchain. If the blockchain returns a different result as what is proposed by the solver, the solver’s deposits will be confisticated. All but the last step is computed off-chain, which is a much more reliable approach than the BFT algorithms requiring at least ⅔ people behaving honestly. In such a consensus mechanism where no dispute can occur, system security is assured as long as there is one honest verifier.

The jackpot mechanism:

Money in the jackpot comes from the commissions given by task givers. The fraction of commissions put into the jackpot is called taxes. Verifiers (challengers) are rewarded by payout from the jackpot. When a solver makes a mistake without a forced error given by the system, his/her deposits will go to the jackpot. A reasonable design of the fraction of taxes can guarantee profits for verifiers who protect the system security when not harming task givers’ initiative.

Summary of main steps in TrueBit:

1. A task giver publishes a task (a computational work to be done) and its associated rewards.
2. A solver is defined in a lottery, who has to prepare a correct answer and a wrong one, too.
   1. If the system starts the “forced error” mechanism, the solver has to show his wrong answer.
   2. Otherwise, he shows the correct answer.
3. Verifiers (challengers) checks the answer from the solver, they will get payout from jackpot if:
   1. They successfully find out that the answer is indeed wrong.
   2. They find a forced error started by the system.
4. If no verifier (challenger) claims that the answer is a mistake, the system will accept the answer. Otherwise, the answer accepted by the system depends on the result of “The Verification Game”.

**Conclusion & Future Works**

As the Corda project is being deployed with an external smart contract language, we will witness if such connection will work. In 2020, by applying multiple scaling solutions, the transactions per second in NEO is expected to reach 100,000, which can compete with centralized payment systems. Going back to the De-Fi world (Table1.1) conceived in Section 1, layers between the top and the bottom are beyond the scope of this paper and require future work, as they are all vital to the health of a De-Fi system. The layer above infrastructure is identity, standing for a real-name system. To be put into mass application, data privacy, security and storage protocols are essential as well. Upon these underlying architectures, we can then build assorted protocols to fulfill different use cases in De-Fi. What we see in products & services are the “final presentation” facing users.

However, the key to landing such a system in China should be a mature regulation framework going side by side with technology. The regulatory departments still have a long way to go before we can see a compliance system filling the gaps between dreams and realities.

References

[1] General Information on NEO Transaction Tracker. <https://neotracker.io/browse/tx/1>

[2] General Information on Corda Enterprise Documents. “Sizing and Performance”. <https://docs.corda.r3.com/sizing-and-performance.html>

[3] Varma, Jayanth Rama. (2019). “Blockchain in Finance”. VIKALPA, Volume(44), Page Number(1-11). <https://doi.org/10.1177/0256090919839897>

[4] Zhong, Lin., & Wu, Qianhong., & Xie Jan., & Guan, Zhenyu., & Qin, Bo (2019). “A secure large-scale instant payment system based on blockchain”. Elsevier, Volume(84), Page Number(349-364). <https://doi.org/10.1016/j.cose.2019.04.007>

[5] Barber, Phillip (2013). “Bull In the China Market: The Gap Between Investor Expectations And Auditor Liability for Chinese Financial Statement Frauds”. Duke Journal of Comparative & International Law, Volume(24:349), Page Number(350-353). <https://scholarship.law.duke.edu/djcil/vol24/iss2/3>

[6] Liu, Yujing. “China crackdown on fraudulent P2P platforms results in 62 overseas arrests, US$1.5 billion of assets frozen”. South China Morning Post. 18 Feb, 2019. Web. <https://www.scmp.com/business/china-business/article/2186608/china-crackdown-fraudulent-p2p-platforms-results-62-overseas>

[7] “China bans initial coin offerings calling them “illegal fundraising””. BBC News. 5 Sep, 2017. Web. <https://www.bbc.com/news/business-41157249>

[8] General Information on China Banking Regulatory Commission - Guide to strengthen risk prevention and control. <http://www.cbrc.gov.cn/chinese/home/docDOC_ReadView/E2D221D7C7BB463A85D8D3F8059947AE.html>

[9] Mike Hearn, Richard Gendal Brown (2019). “Corda: A distributed ledger”. Section 6.1. <https://docs.corda.net/_static/corda-technical-whitepaper.pdf>

[10] Transactions in Corda. Digital Image. Corda Docs. 20 Mar, 2020. <https://docs.corda.net/key-concepts-transactions.html>

[11] Mike Hearn, Richard Gendal Brown (2019). “Corda: A distributed ledger”. Section 4.2. <https://docs.corda.net/_static/corda-technical-whitepaper.pdf>

[12] Mike Hearn, Richard Gendal Brown (2019). “Corda: A distributed ledger”. Section 10. <https://docs.corda.net/_static/corda-technical-whitepaper.pdf>

[13] Graph in Corda. Digital Image. Corda Docs. 20 Mar,2020. <https://docs.corda.net/releases/release-M1.0/visualiser.html>

[14] Mike Hearn, Richard Gendal Brown (2019). “Corda: A distributed ledger”. Section 7. <https://docs.corda.net/_static/corda-technical-whitepaper.pdf>

[15] Smart Contracts in Corda. Digital Image. Corda Docs. 20 Mar, 2020. <https://docs.corda.r3.com/running-a-notary-cluster/introduction.html>

[16] Mike Hearn, Richard Gendal Brown (2019). “Corda: A distributed ledger”. Section 6.4-6.5. <https://docs.corda.net/_static/corda-technical-whitepaper.pdf>

[17] Notaries in Corda. Digital Image. Corda Docs. 20 Mar, 2020. <https://docs.corda.net/releases/release-M10.1/tutorial-contract.html>

[18] Mike Hearn, Richard Gendal Brown (2019). “Corda: A distributed ledger”. Section 6.7-6.8. <https://docs.corda.net/_static/corda-technical-whitepaper.pdf>

[19] Tear-offs in Corda. Digital Image. Corda Docs. 20 Mar, 2020. <https://docs.corda.net/key-concepts-tearoffs.html>

[20] Satoshi Nakamoto (2008). “Bitcoin: A peer-to-peer Electronic Cash System”. Section 4. <https://bitcoin.org/bitcoin.pdf>

[21] Vitalik Buterin (2019). Proof of Stake in Ethereum. Github Repository. <https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ>

[22] Greg Lee (2018). “EOS.IO Technical Whitepaper v2”. Delegated Proof of Stake in EOS. Github Repository. <https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md#consensus-algorithm-bft-dpos>

[23] General Information on NEO Whitepaper. Delegated Byzantine Fault Tolerance in NEO. <https://docs.neo.org/docs/en-us/basic/technology/dbft.html>

[24] Vitalik Buterin., & Virgil Griffith (2017). “Casper the Friendly Finality Gadget”. Github Repository. <https://github.com/ethereum/research/blob/master/papers/casper-basics/casper_basics.pdf>

[25] Eleftherios Kokoris-Kogias., & Philipp Jovanovic., & Linus Gasser,. & Nicolas Gailly,. & Ewa Syta,. & Bryan Ford (2017). “OmniLedger: A Secure, Scale-Out, Decentralized Ledger via Sharding”. IACR. <https://eprint.iacr.org/2017/406.pdf>

[26] Solutions to the Impossible Trinity in OmniLedger. Digital Image. 20 Mar 2020. <https://medium.com/primalbase/omniledger-a-secure-scale-out-decentralised-ledger-via-sharding-7a71adb9ec3b>

[27] Joseph Poon., & Thaddeus Dryja (2016). “The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments”. <https://lightning.network/lightning-network-paper.pdf>

[28] Heiko Hees., & Oliver Bunting (2017). “Raiden Network”. Picolo Research. <https://picoloresearch.com/upload/attachment/98636.pdf>

[29] Image of state channels on Ethereum. Digital Image. EthHub. 20 Mar 2020. <https://docs.ethhub.io/ethereum-roadmap/layer-2-scaling/state-channels/>

[30] General information on “Trinity White Paper An Off-chain Scaling Solution for Neo”. <https://static2.coinpaprika.com/storage/cdn/whitepapers/140779.pdf>

[31] General information on “Liquidity Network: Blockchain Payments for Everyone”. <https://icosbull.com/whitepapers/2786/Liquidity_Network_whitepaper.pdf>

[32] Image of the Hub in Liquidity Network. Digital Image. Whitepaper of Liquidity Network. 20 Mar 2020. <https://icosbull.com/whitepapers/2786/Liquidity_Network_whitepaper.pdf>

[33] Joseph Poon., & Vitalik Buterin (2017). “Plasma: Scalable Autonomous Smart Contracts”. <https://www.plasma.io/plasma.pdf>

[34] Image of Plasma on Ethereum. Digital Image. EthHub. 20 Mar 2020. <https://docs.ethhub.io/ethereum-roadmap/layer-2-scaling/plasma/>

[35] ScaleSphere Foundation Ltd (2018). “Celer Network: Bring Internet Scale to Every Blockchain”. <https://www.celer.network/doc/CelerNetwork-Whitepaper.pdf>

[36] Image of Celer Network. Digital Image. Whitepaper of Celer Network. 20 Mar 2020. <https://www.celer.network/doc/CelerNetwork-Whitepaper.pdf>

[37] Jason Teutsch., & Christian Reitwießner (2017). “A scalable verification solution for blockchains”. <https://people.cs.uchicago.edu/~teutsch/papers/truebit.pdf>

[38] Sina Habibian. Digital images and general Information on Truebit. “Truebit: the marketplace for verifiable computation

”. <https://sinahab.com/2018/03/truebit-the-marketplace-for-verifiable-computation/>

1. As calculated from NEO transaction tracker [↑](#footnote-ref-1)
2. Varma, Jayanth Rama. (2019). “Blockchain in Finance” [↑](#footnote-ref-2)
3. Zhong, Lin., & Wu, Qianhong., & Xie Jan., & Guan, Zhenyu., & Qin, Bo (2019). “A secure large-scale instant payment system based on blockchain” [↑](#footnote-ref-3)
4. Barber, Phillip (2013). “Bull In the China Market: The Gap Between Investor Expectations And Auditor Liability for Chinese Financial Statement Frauds” [↑](#footnote-ref-4)
5. General Information from News on China Cracking Down P2P Platforms [↑](#footnote-ref-5)
6. General information from news on China bans ICOs calling them “illegal fundraising” [↑](#footnote-ref-6)
7. China Banking Regulatory Commission - Guide to strengthen risk prevention and control [↑](#footnote-ref-7)
8. Section 6.1 Transaction Structure of Corda Technical Whitepaper [↑](#footnote-ref-8)
9. Image Source from Corda Docs [↑](#footnote-ref-9)
10. Section 4.2 Data visibility and dependency resolution of Corda Technical Whitepaper [↑](#footnote-ref-10)
11. Section 10 Scalability of Corda Technical Whitepaper [↑](#footnote-ref-11)
12. Image Source from Corda Docs [↑](#footnote-ref-12)
13. Section 7 Notaries and Consensus of Corda Technical Whitepaper [↑](#footnote-ref-13)
14. Image Source from Corda Docs [↑](#footnote-ref-14)
15. Section 6.4-6.5 Attachments and Contract Bytecodes, Contract Constraints of Corda Technical Whitepaper [↑](#footnote-ref-15)
16. Image Source from Corda Docs [↑](#footnote-ref-16)
17. Section 6.7-6.8 Dispute Resolutions, Oracles and Tear-offs of Corda Technical Whitepaper [↑](#footnote-ref-17)
18. Image Source from Corda Docs [↑](#footnote-ref-18)
19. Section 4 Proof of Work of Bitcoin Whitepaper [↑](#footnote-ref-19)
20. Original Document on Proof of Stake in Ethereum [↑](#footnote-ref-20)
21. Original Document on Delegated Proof of Stake in EOS [↑](#footnote-ref-21)
22. Original Document Delegated Byzantine Fault Tolerance in NEO [↑](#footnote-ref-22)
23. General information from the paper for Casper [↑](#footnote-ref-23)
24. General information from the published paper for OmniLedger [↑](#footnote-ref-24)
25. Image Source from Medium [↑](#footnote-ref-25)
26. General information from paper for Lightning Network [↑](#footnote-ref-26)
27. General information from Raiden Network [↑](#footnote-ref-27)
28. Image Source from EthHub [↑](#footnote-ref-28)
29. General information from Trinity Network Whitepaper [↑](#footnote-ref-29)
30. General information from Liquidity Network Whitepaper [↑](#footnote-ref-30)
31. Image Source from Liquidity Network Whitepaper [↑](#footnote-ref-31)
32. General information from the paper for Plasma [↑](#footnote-ref-32)
33. Image Source from EthHub [↑](#footnote-ref-33)
34. General information from Celer Network Whitepaper [↑](#footnote-ref-34)
35. Image Source from Celer Network Whitepaper [↑](#footnote-ref-35)
36. General information from TrueBit Whitepaper [↑](#footnote-ref-36)
37. Images and general information from SinaHub [↑](#footnote-ref-37)