Contents

Chapter 1

An Overview of Blockchain Interoperability

Vasileios Koukoutsas, Te Tan

Blockchain technology has experienced a fast development over past years driving its usage in many application areas beyond FinTech (Financial Technology). Different areas has different requirements and thus, different blockchain usage scenarios are being developed. Cross-chain interoperability becomes the key to answer questions like How to do transactions cross different blockchains and how to overcome the scalability problem of blockchains. In this paper, a summary of the current and potential approaches to cross-chain interoperability is presented. Then, challenges towards interoperability implementation will be discussed. Finally, we will talk about some use cases of interoperability.

Contents

1.1 Introduction

Since Satoshi Nakamoto proposed Bitcoin in 2008 [1], there has been a surprising growth of cryptocurrencies powered by blockchain technology. According to coinmarketcap, there are more than 1500 types of cryptocurrencies available in the market [2]. More than just cryptocurrencies, blockchain technology has experienced a thriving development. For example, Ethereum acts as a successful platform which enables convenient development and execution of Smart Contracts [3].

Different organizations and various use cases have lead to many blockchain implementations. Hence it becomes more and more important that these individual blockchains can "*talk*" to each other. Taking Supply Chain for instance, the adoption of blockchain solution could help improve the efficiency and transparency of products' flow. However, many organizations which are related to a specific supply chain may implement their own blockchain respectively. Hence, there comes the need for information sharing between these blockchains across the supply chain flow. Another problem that cross-chain interoperability can mitigate is scalability. The data storage capability of blockchain is limited by the time needed to create a block and the block size. One possible method is to store the data in multiple blockchains in parallel, with the help of interoperability. Cross-chain interoperability has been identified by Underwood as one of the key challenges to blockchain technology [4].

This report summarizes approaches aiming at solving the cross-chain interoperability problem. Then, use cases in which cross-chain interoperability plays an important role are discussed to overview different approaches towards interoperability.

1.2 Approaches to interoperability

In this section, we summarize the approaches which already have or could potentially enable cross-chain interoperability.

1.2.1 Notary

Technically speaking, the *Notary* is the simplest way to facilitate cross-chain interoperability [5]. In this mechanism, one trusted or a set of trusted entities is used to claim whether an event happened on a blockchain. In order to explain this mechanism more intuitively, we introduce the Interledger, which is an advanced implementation of Notary mechanism [6]. *Interledger* is a protocol which enables interledger transactions with the help of the nodes which have accounts on both ledgers (blockchains). There are three kinds of roles in Interledger: *sender*, *receiver* and *connector*. A sender is someone who wants to initiate a cross-ledger transaction with the receiver. A connector is a node who facilitates the transaction by coordinating the asset transfer on multiple ledgers.

Figure 1.1 shows how Interledger makes cross-ledger transaction possible. Figure 1.1 *a* shows connector's functionality: to transfer the asset from A to B, A can first transfer it

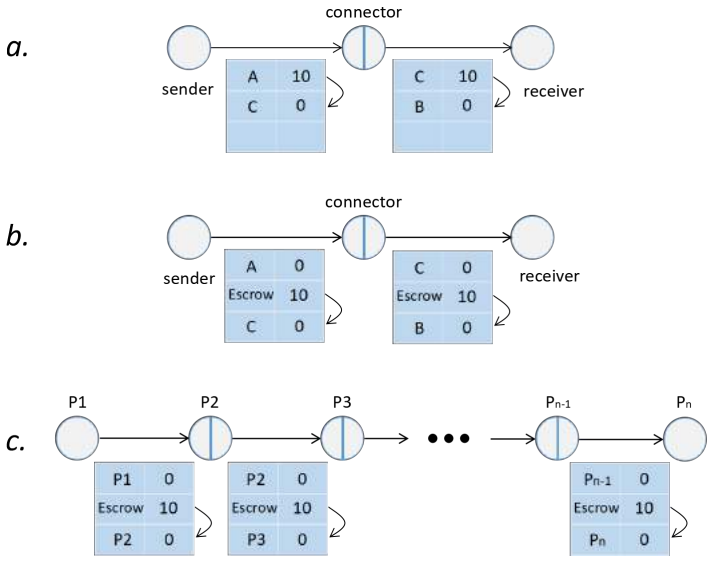

Figure 1.1: Cross-ledger transactions enabled by Interledger protocol

to C on one ledger, and then C transfer it to B on another ledger. This can be extended to the case which involves an arbitrary number of connectors as shown in picture *c* (so-called payment chain). However, in picture *a* nothing prevents C from misbehavior like stealing the money. So Interledger protocol introduces *escrow* and *notaries*. At the beginning of the transaction, the sender will select his trusted notaries, whose role is to coordinate and synchronize the transaction. Escrow is an intermediate between participants. All the participants only transfer their assets to their respective escrows, who will only execute to transfer the asset to the next participant when they received the *Execute Message* from (most) notaries. Taking picture *c* in Figure 1.1 for instance, after P_{n-1} has transferred the asset into its escrow, P_n has to sign a receipt and send it to all notaries. The notaries will decide whether they receive the receipt in time. If they do whey will sign a Message of *Execute* and send it to all participants. Finally, the transaction will execute, and every participant claims their assets.

1.2.2 Sidechain/Relay

Sidechain(or Relay) approach is proposed by Adam Back et al. which aims to enable bitcoins and other ledger assets to be transferred between multiple blockchains [7]. A sidechain is a blockchain that validates data from other blockchains. Instead of relying on intermediaries as Notary does, blockchains do information validation by themselves. The blockchain from which the sidechain verifies data is called as the main chain or parent chain.

The sidechain will use the standard verification procedure to verify the block header from the main chain. To explain how it works, we take the blockchain with proof-of-work as the consensus algorithm as our example. We will first explain how a sidechain validate a transaction happened in the main chain. Then based on that, we introduce how to enable asset portability with the help of sidechain.

1.2.2.1 Verify a transaction in another chain

Suppose a blockchain A is going to verify whether a transaction *TX* took place in blockchain B. In this case, chain A is the sidechain and chain B is the main chain. Two steps are needed: first A will verify the block header containing *TX* has been finalized, then A will verify the specific *TX* against the block header.

Validate the block header: chain A uses the same consensus algorithm as chain B which is proof-of-work, which works as follow:

1. *verify proof-of-work:* chain A fetches the block header from chain B which contains the targeted transaction. The information contained in the block header is shown in Table 1.1. To generate a block in the proof-of-work based blockchain, a certain amount of computational efforts should be invested. Validators must solve a hashing problem in which a solution named Nonce(as shown in Table 1.1) should be found. The difficulty of this hashing problem is constrained by Difficulty target. The inherent characteristic of hashing problem ensures that the exploitation of solution is hard while the verification of the solution is easy. Only Blocks with the correct Nonce are regarded as valid. What chain A does is verifying whether this Nonce is the correct answer to the hashing problem. If it is, chain A thinks the block is valid.

2. *wait until the block is finalized:* a valid block is not enough to confirm the transactions in the blockchain. To mitigate the risk of reorganization, chain A has to wait until a few more blocks have been generated on chain B. Reorganization occurs when more than one block has formed and added to the blockchain simultaneously, which lead to the formation of forks. Then further blocks will be added on these forks. In proof-of-work based blockchain, only the longest fork, which represents the fork with the most computational efforts invested, will be kept and all other forks will be discarded. Hence chain A need to wait for a few more blocks to be generated to ensure the target block is finalized.

Table 1.1: Information contained in the block header

Field	Description
Version	The version number of blockchain
Previous block hash	the hash code of previous block
Merkle root	the hash code of the root of the Merkle tree
Timestamp	the timestamp of the block
Difficulty target	the difficulty target for the formation of the block
Nonce	the counter used by validators to generate a block

Verify the target transaction: after the validation of block header, chain A will then verify the target transaction. As shown in Table 1.1, there is a field named Merkle root within the block header. Merkle root is the root hash code of the Merkle Tree, which is a layered tree structure of the hash code. The leaves of the Merkle Tree are the hash codes of each transaction stored in the block. Every non-leaf node is the hash code of its child

nodes. The root hash code of this tree is included in the block header. An example of Merkle Tree is shown in Figure 1.2.

With the help of Merkle Tree, chain A can verify a particular transaction by downloading a single branch of Merkle Tree, instead of downloading the whole block. For instance, if chain A wants to verify transaction *L4* in Figure 1.2, chain B will give him following nodes: *Hash 1-0* and *Hash 0*. Chain A will then compute *Hash 1* from hash code of *L4* and *Hash 1-0*, and next he will compute *Top Hash*(Merkle root) from *Hash 1* and *Hash 0*. If the computed *Top Hash* is the same as the Merkle root he got from block header, the transaction *L4* is valid.

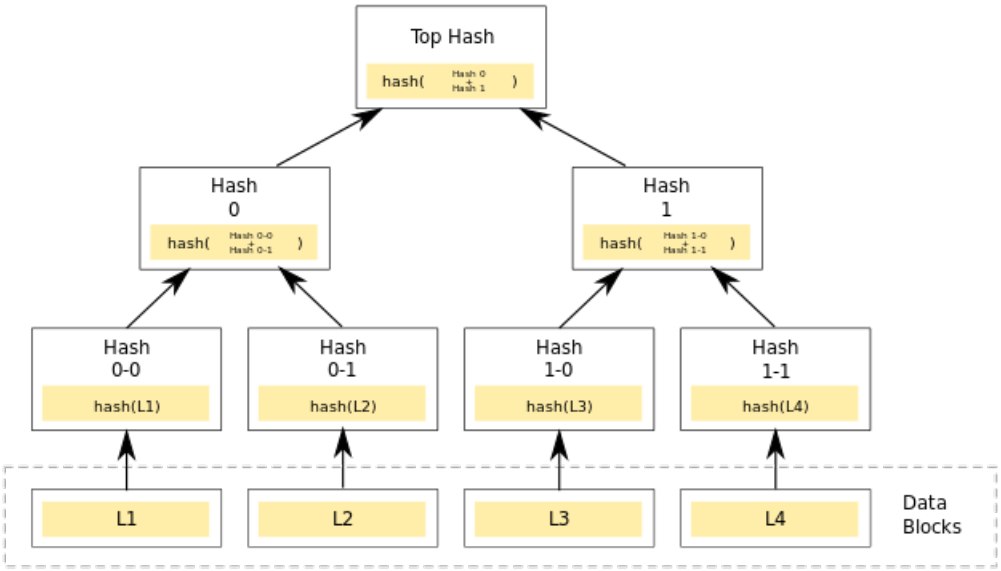

Figure 1.2: An example of the Merkle Tree structure [8]

1.2.2.2 Asset portability enabled by sidechain

Above we explained how the sidechain validates a transaction in the main chain, the same approach can be used to enable the asset portability between sidechain and the main chain. Figure 1.3 explains how the asset of user A on the main chain can be transferred to user B on the sidechain. The first user A launches a transaction, sending his asset to a special address on the main chain which will lock this asset. Then the sidechain will rely on the validation procedure described in the last subsection to verify this transaction on the main chain: A has sent the asset to the special address. Note that once the asset has been sent to the special address successfully, it can only be unlocked by the sidechain, who will use its consensus algorithm to confirm that A's asset has not been spent elsewhere. After the confirmation of A's transaction, the sidechain will create a new asset on itself, which A can use for further transactions on the sidechain without constraints. For example, A can transfer it to B as shown in Figure 1.3.

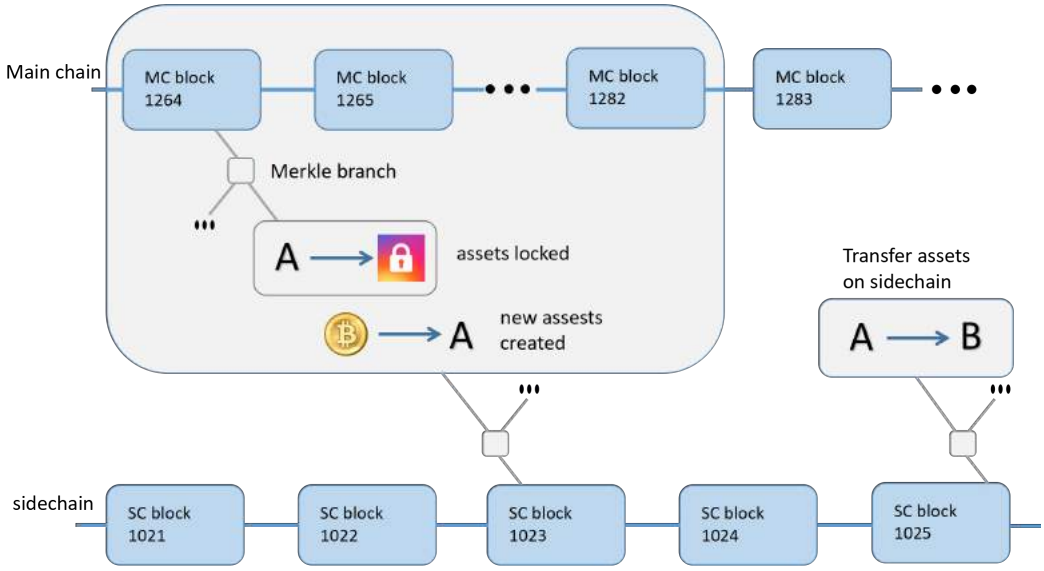

Figure 1.3: Asset transfer enabled by sidechain

1.2.3 Hash-locking

Hash-locking is another well-known technique to achieve cross-chain assets exchange[5]. One of the biggest advantages of Hash-locking is that it requires no notary or sidechain. Its limitation is also obvious: it can only use for atomic operations, not for asset portability.

Figure 1.4 shows how this mechanism works. In this case, A wants to exchange assets with B. Firstly A generates a random secret s, computing its hash h and sending it to B. Secondly A locks his asset into a smart contract, B also locks his counterpart asset into a smart contract after he verifies A did so. Then it comes to the stage of claiming assets. If B receives the correct secret s from A within X seconds, B's asset will be transferred to A. Similarly, if A receives secret s from B within $2X$ seconds, A's asset will be transferred to B. Note that in order to claim the asset from B, A must reveal the secret s within X seconds, which ensures B has at least X seconds time window to claim the asset from A because A has to wait for $2X$ seconds according to the mechanism.

1.3 Challenges to cross-chain interoperability

In this section, we will discuss the challenges to interoperability as mentioned above mechanisms, as well as promising techniques to solve these problems.

1.3.1 Scalability

One of the biggest limitations of blockchain for widespread use, especially in financial markets, is lack of scalability. This problem largely comes from the underline consensus

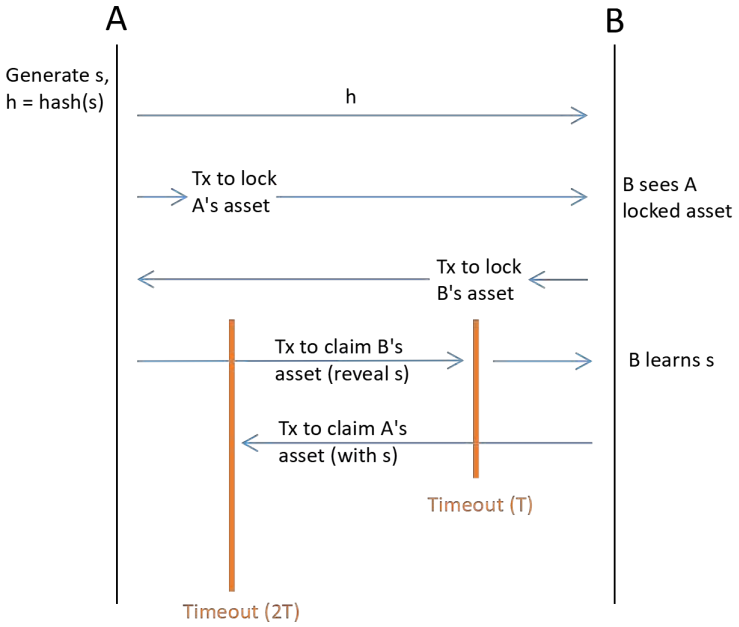

Figure 1.4: Hash-locking mechanism

algorithms adopted by blockchains. Taking the sidechain explained in Figure 1.3 for instance, whose consensus algorithm is proof-of-work. When the sidechain wants to verify that A has locked his asset successfully into the special address, firstly it should wait the time of block generation, which is about 10 minutes with large variance. After the block is generated, to lower down as much as possible the risk of reorganization(forks may emerge), the sidechain has to wait until several additional blocks have been created. This considerable latency undermines many commercial applications, which requires nearly instant execution.

A few researchers are trying to solve this problem and proposed different strategies. One promising solution is so-called *Strong Federation*. *Strong Federation* is a sidechain which introduces some advanced properties such as publicly verifiable, privacy protection, and most importantly, low transaction confirmation time [11]. To significantly lower the transaction latency and eliminate the risk of reorganization, *Strong Federation* adopts a group of fixed signers named *Blocksigners* and replace proof-of-work with a multisignature scheme. The formation of different forks in proof-of-work is because there are many subgroups of validators working on generating different blocks. Hence using a fixed group of signers on sidechain to validate the transaction will eliminate the risk of reorganization. Furthermore, the adoption of multisignature which is a mechanism requiring blocks to be signed by a certain threshold of signers, will largely lower down the confirmation time of one block.

Another solution which is potentially suitable for commercial adoption is *Herdius*, which provides a different method dealing with the scalability problem [9]. *Herdius* is another sidechain solution which adopts proof-of-state as the consensus algorithm. This will lower down the time of transaction confirmation largely. Furthermore, it introduces the concept of *stretched block*. As shown in Figure 1.5, Herdius uses a transaction queue to gather

transactions formed from the time of the generation of the last block up to now. If the transactions do not exceed the normal size of one block, Herdius chain will generate a singular block next time. If the number of transactions is too large for a singular block to hold, a stretched block will be generated, with a tree structure of a few more blocks whose root points to the regular block on the main chain. To verify these blocks, validators will be split into subgroups. In this way is Herdius chain able to handle a large number of transactions in parallel and hence the transaction throughput will increase significantly.

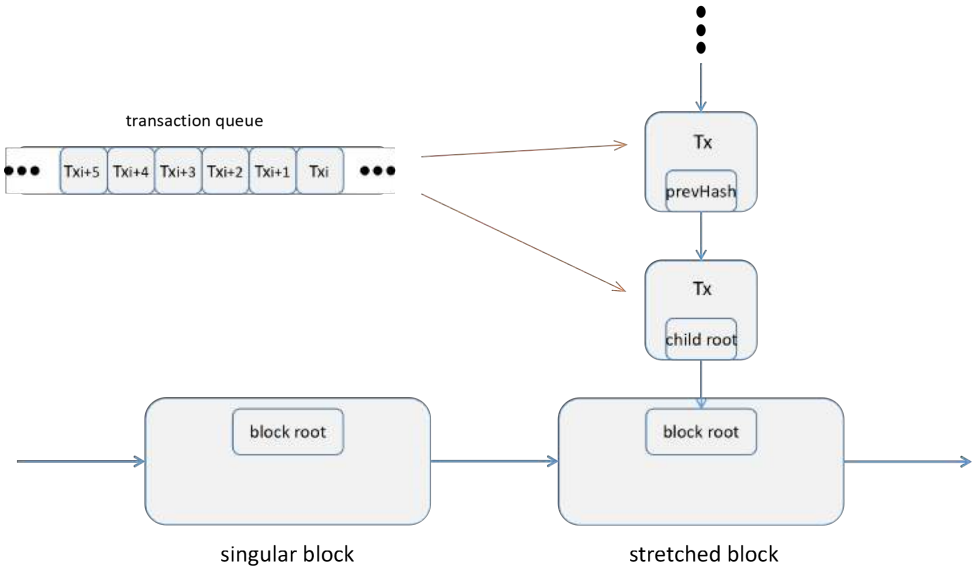

Figure 1.5: The block structure and transaction queue of Herdius

1.3.2 Security

Security has always been one of the most critical issues in blockchain and its ecosystem. However, in a multi-chain context, the security model is difficult to construct. Interoperable blockchains have different security models, and cross-chain information transmission also needs to be modeled. For instance, 51% attack is not likely to happen if the profit of reversing one block cannot cover the costs of large-scale resource manipulation, which means economical, infeasibility prevents attackers from attacking. However, in the context of multi-chain, transactions can involve multiple participants from different blockchains as well as multiple applications running on these blockchains. It may be that a 51% attack is not feasible when the involved value does not exceed 50000 in a single application, but it can be profitable when the attack can affect thousands of such applications. Therefore, security model in a multi-chain case should be treated differently.

1.3.3 Practice

As mentioned in [5], use cases involving blockchain interoperability will take a long time to come to fruition, since the set of dependencies is large and it is indeed resourced consum-

ing. Cryptocurrency exchange is one driving force for the implementation of interoperability system, and further use cases such as finance also motivate the works. Even though there are a few solutions proposed, most of them are still at the theoretical stage(theories and mechanisms are described in their whitepapers). The best practice and systematic methodology of cross-chain interoperability are still on its way to being explored.

1.4 Use cases of interoperability

1.4.1 General Purpose

A general purpose system for blockchain interoperability using 2-way pegged sidechains is *Strong Federations* by Blockstream. A Strong Federation is a group that serves as a mutually-incentivized protocol adapter between an "anchor chain" and one of its sidechains and acts as a unit to ensure forward progress of the sidechain. Using cryptographic tools and secure hardware, the participants construct a Byzantine-robust smart contract wherein each "functionary" is economically incentivized to operate in the best interest of the network by the mutually agreed upon rules.

While leveraging proof-of-work provides Bitcoin with unprecedented security for transaction history, this benefit comes at a cost in latency and throughput. Strong Federations address the delay by introducing a deterministic set of participants each with two responsibilities: generating valid blocks and enforcing withdrawal rules. Transactions are published in blocks that must be made visible to all participants in the network and validated. Pre-commitments are made and then blocks signed. This coordination is measured in seconds as opposed to minutes for Bitcoin. As in Bitcoin, the knowledge of a private key is sufficient for the "right to spend" without the permission of any third party [7].

The system process flow includes the following steps:

1. The user sends their asset to a special address that is designed to freeze the asset until the sidechain signals that asset is returned.

2. Using the *in* channel of a federated peg, the user embeds information on the sidechain stating that the asset was frozen on the main chain and requests to use it on the sidechain.

3. Equivalent assets are unlocked or created on the sidechain, so that the user can participate in an alternative exchange under the sidechain rules, which can differ from the parent chain.

4. When the user wishes to move her asset or a portion thereof, back via the "out channel," she embeds information in the sidechain describing an output on the main blockchain.

5. The Strong Federation reaches a consensus that the transaction occurred.

6. After consensus is reached, the federated peg creates such an output, unfreezing the asset on the main blockchain and assigning it as indicated on the sidechain.

A high level overview of the system is shown in Figure 1.6

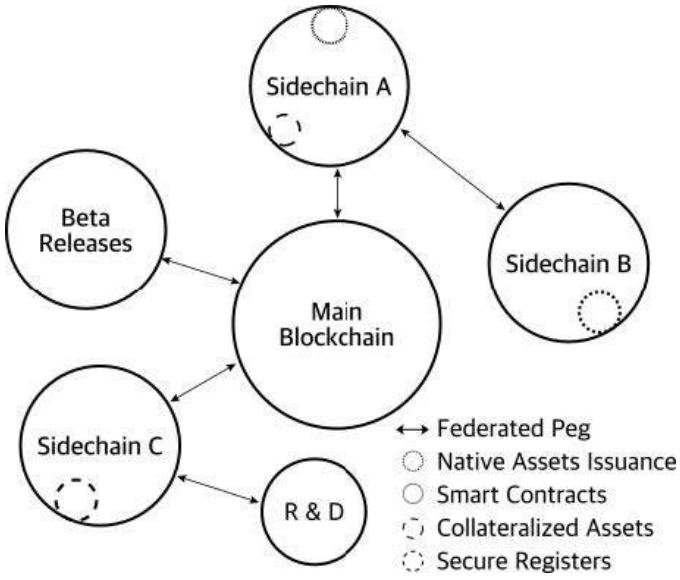

Figure 1.6: Pegged sidechains allow parties to transfer assets by providing explicit proofs of possession in transactions

Strong Federations could potentially be implemented to serve different kind of decentralized blockchain based applications and use cases.

1.4.2 Financial markets and Assets Portability

An implementation of Strong Federations aimed to serve the financial markets is *Liquid* which is an interoperable sidechain that extends the Bitcoin blockchain while adding an auditable, cryptographically-strong commercial privacy component. Using this arrangement, Liquid leverages the reliability and security of the Bitcoin network without trusting a centralized third party. This new construction establishes a security profile inherently superior to existing methods of rapid transfer and settlement, and is directly applicable to other problems within existing financial institutions.

Using sidechain technology, Liquid reduces ISL by allowing for rapid transfers between accounts held by the varied participants in a separate, high-volume and low-fee cryptographic system that preserves many of the security benefits of the Bitcoin network. This, in addition to increasing the security of funds normally subject to explicit counterparty risk, fosters conditions that increase market liquidity and reduce capital requirements for on-blockchain business models [15].

Benefits of Liquid:

1. **Faster Trading** Near instant bitcoin transfers between exchanges allow your users to take advantage of arbitrage opportunities like never before.
2. **Enhanced Efficiency** Market makers can improve their capital efficiency by reducing balances held across multiple exchanges.
3. **Better Privacy** Liquid supports Confidential Transactions for bitcoin amounts transferred in the system, which protects your users from exposure.
4. **Superb Reliability** Built using the battle tested Bitcoin code-base, Liquid software is highly reliable. Also, since Liquid uses signed blocks instead of mining, blocks are always one minute apart instead of an unknown amount of time like Bitcoin.

The participating exchanges and Bitcoin businesses deploy the software and hardware that make up the Liquid network so that they can peg in and out of the Bitcoin blockchain and offer Liquid's features to their traders. Liquid provides a more secure and efficient system for exchange-side bitcoin to move across the network. End users benefit from the greater liquidity Liquid enables between exchanges.

Liquid is a federated sidechain, so it will never be as decentralized as Bitcoin. However, Liquid is designed to remove control from any single party, geographic location, or political jurisdiction. The Liquid Network is operated by functionary servers, each securely hosted by geographically dispersed, independently owned and operated Bitcoin exchanges. Updates are deployed by consensus of participants within the network. No single party, including Blockstream, can control the Liquid network, and furthermore, no single entity is in control of more than a single Liquid functionary server.

Liquid is built using the Elements blockchain platform and therefore has multi-asset issuance capabilities built in through the Confidential Assets feature. In its first release, Liquid will support Bitcoin only. Future versions of Liquid may support other assets, such as other cryptocurrencies or assets issued by participants [14].

1.4.3 Cross-chain Contracts

Aelf is a crosschain blockchain protocol that creates a highly efficient and customizable OS that will become the "Linux system" of the blockchain community. It focuses on defining and providing the most basic, essential and time-consuming component of the system and making significant improvements based on existing chains in the market. The system allows developers to customize it to meet their own needs, particularly commercial requirements for various industries. Firstly, the Aelf kernel is defined and implemented which includes fundamental functions of a blockchain system, namely the minimum viable blockchain system. Secondly, a "shell" is developed as the basic interactive interface to the Core. Users can either use the complete Blockchain OS or rapidly develop a customized OS based on the Core via redefining the Core through interfaces. [16]

Aelf consists of one main chain and multiple sidechains which are attached to it as it is shown in Figure 1.7.

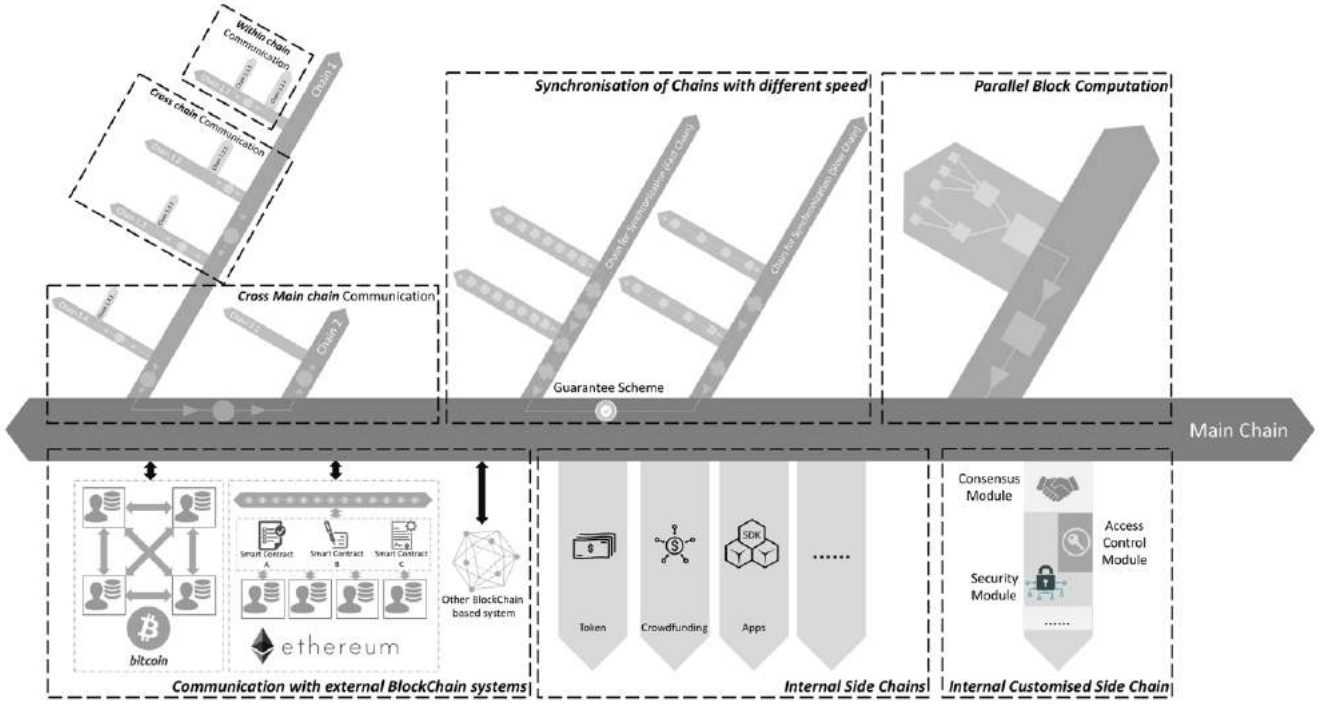

Figure 1.7: Overview of Aelf structure

Aelf will support the below main features [16]:

1. Introduces the concept of Main Chain and multi-layer Side Chains to handle various commercial scenarios. One chain is designed for one use case, distributing different tasks on multiple chains and improve processing efficiency.

2. Enables Aelf to communicate with external blockchain systems via messaging (e.g., Bitcoin, Ethereum).

3. Permits parallel processing for non-competing transactions and cloud-based services.

4. Defines basic components of minimum viable Block and Genesis Smart Contract collection for each chain to reduce data complexity and achieve high customization.

5. Permits stakeholders to approve amendments to the protocol, including redefining the Consensus Protocol. Permits sidechains to join or exit from the main chain dynamically based on COnsensus protocol, therefore introducing competition and incentive to improve each sidechain.

Each node in Aelf is a computer cluster network (e.g., a cloud network) instead of a singular computer. By leveraging cloud networks as nodes, Aelf aims to further empower the network participants with higher computational power as well as the storage capability. The parallel processing algorithm is developed and integrated to each node to ensure the optimal utilization of all the participating computers in the cloud network. When a node

handles a complex set of transactions within a smart contract, it will dissect the transactions into groups of those that do not demonstrate interdependency and process them in parallel simultaneously. Consequently, the takt time is minimized and the overall processing speed maximized. A node's capacity can be easily scaled by adding new computers to the existing network without having to upgrade the node computer's hardware. [17]

As a multichain network, each side chain is independent of one another, and smart contracts reside are executed directly from a side chain, not through the main chain. This enables each side chain to be impervious to the high traffics on another chain, thereby localizing the traffic concentration and guaranteeing consistent transaction speed for smart contracts executed in other side chains. Each side chain would also have the ability to host its own set of nodes to guarantee low traffic and determine its processing speed. Each side chain is specialized for a specific business scenario, e.g., token issuance (ICO), an insurance database, in-game transactions, etc., and their consensus protocol, node delegation, chain privacy and various other chain qualities can be tailored to best support the specific business scenario. The main chain acts as the ledger and the communication hub, unlocking highly efficient cross chain communication, triggering of smart contracts across side chains and effective synchronization between chains with different speeds. [17]

Aelf aims to bestow its network participants an entirely self-evolving authority and capability through its voting protocol. Aelf coin holders will have the ability to vote on a diverse sets of critical decisions that will collectively shape the eco-system; this includes the decision for each side chain to host their own delegated node, choose whether the participating side chain will be public or private, determine the size and the speed of the side chain, remove or add side chains to the network, etc. Aelf utilizes Merkle tree root based chain indexing to communicate and interoperate with other consensus protocol based blockchains such as PoW and PoS. Aelf provides side chain templates to its developers for rapid smart contract development for those who do not have the in-depth understanding and capacity for ground-up smart contract coding. [17]

Aelf is intended to become the new "internet infrastructure" to support the next generation of "digital businesses." Some potential applications include:

1. **Financial Services** It is highly likely that multiple chains on Aelf will be developed specifically for financial services, such as cross-border payment, trade finance, supply chain financing, etc. The parallel processing feature is capable of handling business transactions at the international scale, and the inter-chain communication feature allows smooth coordination from asset registration, account management, real-time transaction.

2. **Insurance** A dedicated Aelf side chain for insurance will integrate various DAPPs for insurance, transforming the whole industry value chain, starting from user identity, to insurance contract execution, to claim to handle.

3. **Digital Identity and IPs** Aelf's multi-chain structure has a built-in chain for digital identity. This ensures the performance of such side chain if another side chain is busy. Within Aelf, digital identity can be used by other side Chains via "messaging." Using adaptor, Aelf is also capable of retrieving information and data from other established chains, such as Bitcoin and Ethereum.

4. **Smart City** Governments or organizations can customize the consensus protocol to meet national security requirement. Activities, such as utility recording, citizen identities, government agency information disclosure and polling can be realized on Aelf with high transparency and efficiency. A few countries are experimenting in this field, including Estonia, Singapore, China, etc.

5. **Internet of Things** Aelf supports light node and cloud service, which reduces the computational requirement for devices connected to it while maintaining high performance. This is critical to managing billions of devices and enables micropayment across them to link internet of things [16].

1.4.4 Supply Chain

Origin Trail is a decentralized protocol that has been designed to share supply chain data via the use of a completely transparent network. Origin Trail makes use of a blockchain that builds on well-established industry standards as well as promotes a P2P network that fosters consumer confidence, optimal supply chain efficiency, automated compliance and quality assurance. Every stakeholder is given the ability to securely share and store their data via encrypted channels. Origin Trail is also fully compatible with existing ERP systems, thereby making the implementation process highly streamlined and uncomplicated. Lastly, the Origin Trail ecosystem is regulated through a tokenized economy model that minimizes the possibility of collusion and introduces full accountability for the provided data. Users can develop direct relations with one another, and all of the network nodes facilitate internal transactions without incurring arbitrary fees; something that is commonly experienced by users of centralized data platforms. [18]

The two key factors that impede data collection and sharing in supply chains are:

- **Data is fragmented** Data siloes and low data interoperability exist across the supply chain in both multi-organisation and single-organization supply chains. There is a crucial technical challenge for various IT providers for supply chains (software and IOT) that need to be resolved to collaborate and establish full supply chain transparency. An illustration of data siloes is shown in Figure **??**

- **Supply chain data is centralized** It is aggregated by one or several entities prompting concerns about data integrity and omitting accountability. The centralized administration also allows for the possibility of data tampering and collusion between parties. By creating a decentralized system, we establish an environment of complete accountability for all data as well as entirely remove the possibility of data tampering and collusion. [18]

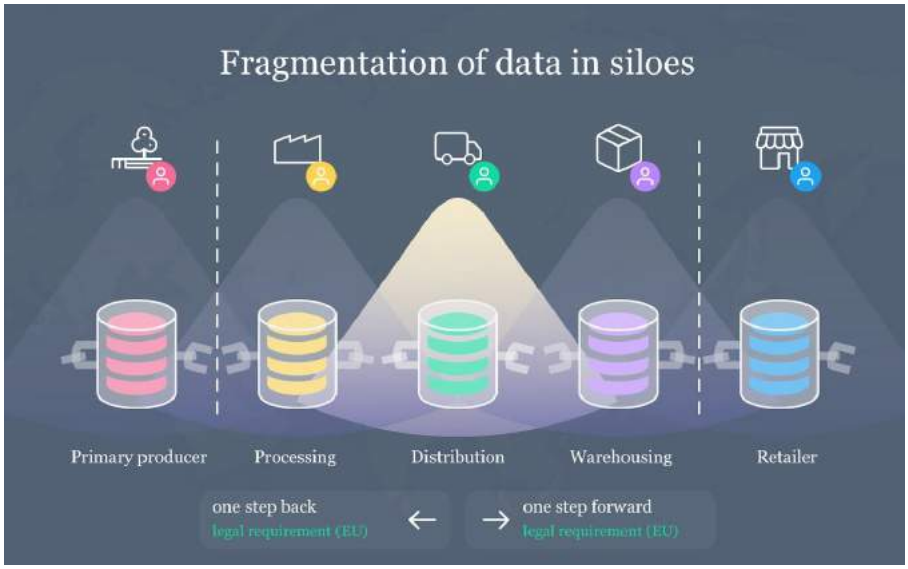

Figure 1.8: Supply chain data fragmentation in siloes

OriginTrail protocol runs on an off-chain decentralized peer to peer network, called the OriginTrail Decentralized Network (ODN). It enables peers on the network to negotiate services, transfer, process and retrieve data, verify it's integrity and availability and reimburse the provider nodes. This solution minimizes the amount of data stored on the blockchain to reduce cost and inefficiency. An overview of the Origin Trail solution stack is shown in Figure 1.9.

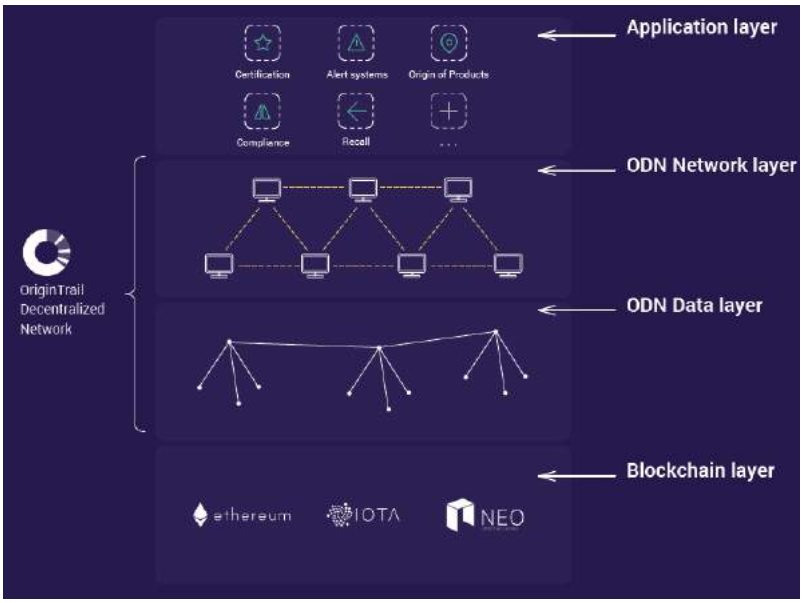

Figure 1.9: Supply chain data fragmentation in siloes

The main layers of Origin Trail solution stack are:

- **Application layer** The application layer is the built on top of the ODN network capabilities and presents the ground for implementing the consumer-facing instances - decentralized applications built by developers, explained further in this document.
- **ODN Network layer** The network layer takes care of the accessibility and data governance of the underlying data layer. It consists of a network of nodes which all contain parts of the decentralized database and store supply chain data in graph form. Access to the data is achieved through the provided data exchange API.
- **ODN Data layer** The data layer of ODN takes care of all the necessary data management and connectivity functionalities. Because of the need to connect many different data sets across the supply chain, while providing the flexibility to support many different connection options, data relationships are the key focus of the data layer. Therefore the basis of the data layer is a decentralized graph database.
- **Blockchain layer** OriginTrail incorporates blockchain as the platform to ensure data integrity and trusted payments. All the data entering the system gets immutably "fingerprinted" in the blockchain (using a cryptographic hash) which provides for a tamper-proof mechanism for supply chain data. The blockchain layer allows the OriginTrail network to utilize different blockchains for fingerprinting which provides flexibility and ensures the longevity of the protocol by not having "blockchain lock-in" to one single platform.

OriginTrail enables seamless and automatic data connection and interoperability between IT systems of different stakeholders in multi-organisation supply chains with consensus mechanisms for ensuring the integrity of data. Interoperability is delivered by integrating globally recognized GS1 standards for Master Data (descriptive attributes for products), Transaction Data (related to business relations), Visibility Data (related to tracing and tracking). Other data sets will include IoT and compliance data. A consensus among entities in the supply chain is achieved by performing cross-reference checks every time a new data set is added to the protocol. This ensures the entire supply chain is in accords regarding a particular batch of products. If there is no consensus, discrepancies can be quickly reported, investigated and reconciled. [19]

The consensus check is performed in 3 steps:

1. Creating a chain of accountability by mutual approval of supply chain stakeholders.
2. Matching of dynamic batch information is verified. Sensitive data is protected by a zk-SNARKs implementation.
3. Auditing and compliance organizations confirm the provided data.

Once the service providers configure the automatic data input (from supply chain ERPs, IoT devices, online and brick & mortar retails stores, etc.), it is introduced to Data Creator (DC) nodes which disseminate the data in the network to other Data Holder (DH) nodes for safekeeping, fingerprinting, performing data standardization checks, consensus checks and creating connections with other already available supply chain data in the system.

Finally, supply chain data is read from the nodes by the decentralized applications from the application layer. All the nodes are reimbursed for these services by Trace tokens in the amounts agreed upon with a bidding mechanism. A system overview is shown in Figure 1.10

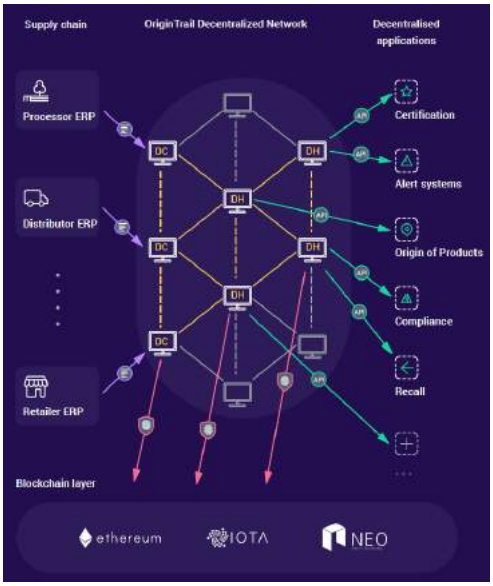

Figure 1.10: System Overview

OriginTrail will deliver the first generic open source applications built on top of the protocol showcasing some of the possible token utilities on the application level:

- **Tokenized data ownership** Creating models where data is sold up/down the supply chain using Trace will be created. This is especially important for primary producers where (production) data is a valuable asset that is currently insufficiently addressed. Using applications built on top of OriginTrail protocol they can take control over their data ownership and earn Trace from providing it to industry partners.

- **Tokenized reputation system** Will be stimulated to share reviews on products and services and contribute to the reputation system made possible by the protocol. Any supply chain stakeholder will be incentivized to provide a review of the product/service with Trace.

- **Tokenizing consumer engagement** Trace tokens will be awarded to end consumers in exchange for interaction with products and services.

Other application instances are to be created by direct users of the system IT providers. Examples of applications where OriginTrail's protocol delivers value are product authentication, supply chain mapping, inventory management, alert systems, supply chain compliance assurance and customs, audit and regulations process optimization. [19]

1.5 Conclusion

Due to the increasing adoption of the blockchain across the industry, Cross-chain interoperability has become an important solution to deal with problems like blockchain communication and scalability. In this report, we introduced the potential approaches to enabling cross-chain interoperability, which is Notary, Sidechain, and Hash-lock. Notary mechanism relies on a set of entities(notaries) to verify whether an event happened on other blockchains. Sidechain avoids intermediaries by validating transactions themselves. Hash-lock uses a smart contract to enable asset-exchange from different blockchains. Then we discussed the current challenges to cross-chain interoperability, including scalability, security, and practice.

The potential use cases for blockchain interoperability are numerous and difficult to predict. In this report, we identified and analyzed a few of the most prominent candidate use cases such as financial markets and assets portability, cross chain smart contracts and supply chain. Almost all currently known blockchain interoperability use cases fall under one of these three categories. It is important to note that blockchain as an industry is still in its infancy and the first applications other than cryptocurrencies entered the market commercially at the end of 2017. Therefore the immediate need for blockchain interoperability is not present. All use cases studied are still in a design phase, or at best in a proof-of-concept phase, we will only be able to verify if these solutions are functional, safe and usable after they have been implemented and deployed. If blockchain interoperability manages to address blockchain performance, privacy and lack of complex features limitations it could potentially give a significant boost in the overall industry and allow for a new global economy where easier, faster and more secure interactions of both systems and people from around the world are possible.

An important design decision will have to be made for all the potential projects that aim to solve blockchain interoperability. The designer of the new system will have to decide whether to entrust a third party or not to handle their transactions and consequently their data. If the Notary or Sidechain solution is used then a third party is needed no matter how the system is designed. By using the hash-locking technique there is no need to entrust a third party, but the implementation can be difficult and has to be done separately for each pair of blockchains. This assumes that all the chains will have to adjust their system and create interfaces. Our prediction is that when humans will have to choose between ease of use and security they will always choose the first over the latter, we have witnessed this kind of behavior repeatedly in the Information Technology sector both at a personal and an organizational level.

Furthermore, there comes other issues along with the adoption of interoperability solutions. The concept of sidechains, is to create extra functional blockchains which have the rights to verify data from other blockchains. This is potentially against the underlying decentralization purpose of blockchain technology. Suppose that in the near future a 'perfect' sidechain has been implemented by a powerful company, which overwhelms any other competitors and dominates the market. The consequence is that there will be more and more blockchains, with the need for data sharing and cross-chain transaction, adopting this dominant sidechain solution. This further enables the company to be the trusted

third party in the blockchain ecosystem, which is a direct contradiction to the purpose of blockchain and decentralized applications.

Bibliography

[1] S. Nakamoto: *Bitcoin: A peer-to-peer electronic cash system.*, Bitcoin, 2009. https://bitcoin.org/bitcoin.pdf.

[2] CoinMarketCap: *Cryptocurrency Market Capitalizations*, March 2018. https://coinmarketcap.com/.

[3] Ethereum Community: *Ethereum Blockchain App platform*, March 2018. https://www.ethereum.org/.

[4] Sarah Underwood: *Blockchain beyond bitcoin*, Communications of the ACM, Volume 59, 15-17, November 2016.

[5] V. Buterin: *Chain Interoperability*, R3.com, September 2013. https://www.r3.com/blog/2017/01/23/chain-interoperability/.

[6] S. Thomas, E. Schwartz: *A Protocol for Interledger Payments*, Ripple, October 2015. http://blockchainlab.com/pdf/interledger.pdf.

[7] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timon, P. Wuille: *Enabling Blockchain Innovations with Pegged Sidechains*, Blockstream, October 2014. https://blockstream.com/technology/sidechains.pdf.

[8] Wikipedia: *Merkle Tree*, March 2018. https://en.wikipedia.org/wiki/Merkle_tree.

[9] D. Balazs: *Herdius-Next Generation Decentralized Blockchain Financial Infrastructure*, Herdius, February 2018. https://herdius.com/whitepaper/Herdius%20Technical%20Paper.pdf.

[10] T. Euler: *A Cryptocurrency Transaction Layer: A Path for Blockchain to go Mainstream?*, Herdius, January 2018. https://medium.com/herdius/a-cryptocurrency-transaction-layer-86347c6688a3.

[11] J. Dilley, A. Poelstra, J. Wilkins, M. Piekarsk, B. Gorlick, M. Friedenbach: *Strong Federations: An Interoperable Blockchain Solution to Centralized Third-Party Risks*, Blockstream, January 2017. https://blockstream.com/strong-federations.pdf.

[12] Iuon-Chang Lin, Tzu-Chun Liao: *A Survey of Blockchain Security Issues and Challenges*, International Journal of Network Security, Vol.19, No.5, September 2017.

[13] A. Azaria, A. Ekblaw, T. Vieira, A. Lippman: *MedRec: Using Blockchain for Medical Data Access and Permission Management*, 2016 2nd International Conference on Open and Big Data, 2016.

[14] Liquid FAQ: *Liquid FAQ*, https://blockstream.com/liquid/faq/.

[15] Liquid Intro: *Introducing Liquid: Bitcoin's First Production Sidechain*, October 2015 https://blockstream.com/2015/10/12/introducing-liquid.html.

[16] Aelf: *Aelf - A Multi-Chain Parallel Computing Blockchain Framework*, 25 November 2017 https://grid.hoopox.com/aelf_whitepaper_EN.pdf?v=1.

[17] Aelf Summary: *In case you forgot, here is a little refresher on aelf*, 22 April 2018 https://medium.com/@aelfblockchain/in-case-you-forgot-here-is-a-little-refresher-on-aelf-3d5dbc5a1b47.

[18] B. Rakic, T. Levak, Z. Drev, S. Savic, A. Veljkovic: *Origin Trail First purpose built protocol for supply chains based on blockchain* 5 October 2017 https://origintrail.io/storage/documents/OriginTrail-White-Paper.pdf

[19] Origin Trail Overview: January 2017, https://origintrail.io/storage/documents/overview_document-english(Jan%209).pdf