White Paper

First purpose built protocol for supply chains based on blockchain

Authors

Branimir Rakic MSc, Tomaz Levak, Ziga Drev, Sava Savic PhD(c)., Aleksandar Veljkovic PhD (c).

October 5, 2017 v1.0

Abstract

Blockchain technology has huge potential to decentralize trust in supply chains and bring measurable benefits and value to the public and private sectors. To unlock this potential, the OriginTrail protocol was designed purposefully to tackle the main challenges which limit the fluent exchange of, and integrity of data in product supply chains. With supply chain data becoming increasingly fragmented, scalability and cost concerns of current decentralized solutions become evident.

OriginTrail is a unique solution allowing IT providers in supply chains to set up blockchain supported data sharing in multi-organizational environment. It helps them build transparency beyond the "one step down, one step up" traceability principle. Furthermore, it improves the integrity of product data and drives efficiencies for stakeholders. The first version of the OriginTrail solution is proven and currently deployed in the food industry. The upcoming open source version will be suitable to any product supply chain such as automotive, consumer goods, pharmaceutical etc.

Visit our website:
www.origintrail.io

Table of contents

1.Document purpose

This document was created with the goal of positioning OriginTrail's protocol concept.The protocol was developed based on **five years of experience building solutions to enable information transparency in supply chains.** It includes our vast practical experience gained during our work on **live business cases**[1] with new technological possibilities created by the blockchain community. Despite quick advancement in development of blockchain, we see a gap between the advantages blockchain can bring and the actual implementations in supply chains in a seamless and efficient manner.

The positive momentum created by the blockchain community functions as a catalyst to implement the technology for one of the biggest challenges faced by the global economy - opaque, inefficient and untrusted supply chains. Employing the capacity of fast growing blockchain networks, OriginTrail will bring decentralization, interoperability, data and information integrity and trust to supply chains.

This is an open source project and by definition relies on (industry and technical) community feedback to grow stronger.

OriginTrail protocol, with certain parts currently centralised, is in pilot programs in Europe and China. The results of these pilot programs will be shared in forthcoming documents.

If you have suggestions or comments on any of the topics in whitepaper, you can get in touch with our team at office@origin-trail.com or join us on our telegram channel.

[1] You may request whitepapers on 3 cases in meat, dairy and vegetables sectors at www.origin-trail.com. Some consumer facing instances may be found at:

- Perutnina Ptuj Ltd link
- Celeia Dairy Ltd link
- Eta Kamnik Ltd link
- Android mobile application
- iOS mobile application

2. Vision

If you take a quick look around your office or home,there is a very high probability that most of the objects surrounding you came through some form of regional, national or global supply chain. The fact that you have very little information with regards to how those products made their way to you is just one of the signs that supply chains are facing more pressure to be transparent.. And most of those issues boil down to a very limited ability to share data along the entire supply chain.

OriginTrail changes that with a decentralised protocol that is tailor made for sharing supply chain data based on blockchain. This brings a profoundly new way of building transparency in supply chains. OriginTrail uses Blockchain and builds on well established industry standards from GS1 and provides a necessary foundation to build new value - increased trust, optimised supply chain efficiencies, automated compliance and enforce quality assurance processes.

Using OriginTrail, all stakeholders can securely share their data and keep sensitive data fully encrypted at all times. By supporting global standards for data exchange (GS1, IoT, compliance standards), OriginTrail assures compatibility with existing ERP systems, making implementation process quick and efficient. Finally, it is fully decentralised. It removes the possibility of collusions and introduces full accountability for the data provided.

OriginTrail is not a company, it is an ecosystem. It’s based on token economy with direct relations between users and network nodes free of arbitrary fees. Contribution to OriginTrail ecosystem is a pledge towards more transparent, collaborative, fair and trusted supply chains.

3. Supply chain challenges

With the globalization of trade there is increasing complexity in supply chains. This, in turn, increases the amount of **information asymmetry** - such that information is unevenly distributed among participating stakeholders within a supply chain. When participating stakeholders have misaligned incentives, such as the case in which participating stakeholders are different companies, there is no incentive to provide complete information which contributes further to information asymmetry.

As a result, end-buyers of products have no economical way of authenticating what they are purchasing, which creates ideal conditions for **moral hazard and fraudulent behaviour**. Manifestations of such phenomena are the flood of counterfeit goods in the market (e.g., US$200 billion in cost to legitimate businesses in the United States[2]), safety issues, violations of labour standards, just to name few. Stakeholders at greatest risk are end-buyers, consumers, the environment, workers and companies involved in sustainable production and honest practices.

Having served our supply chain clients in resolving the challenges above for the past five years, we have identified **two key factors** impeding data collection and sharing in supply chains:

1. **Data is fragmented.** Data siloes and low data interoperability exist across the supply chain in both multi-organisation and single-organisation supply chains. There is a crucial technical challenge for various IT providers for supply chains (software and IOT) that need to be resolved in order to collaborate and establish full supply chain transparency;
2. **There is no suitable decentralised solution for supply chain data**. There are no solutions that can provide the needed level of **performance, scalability and trust** for interconnected data in supply chains while at the same time are **cost-effective**. Current blockchain and decentralised solutions are prohibitively costly, do not provide advanced data relational functionalities, and also have scaling issues.

Nevertheless, all stakeholders have the same goal - being member of the chain and improvement of the whole process regarding volume and efficiency.

[2] http://www.ipwatchdog.com/2010/08/30/counterfeiting-costs-us-businesses/id=12336/

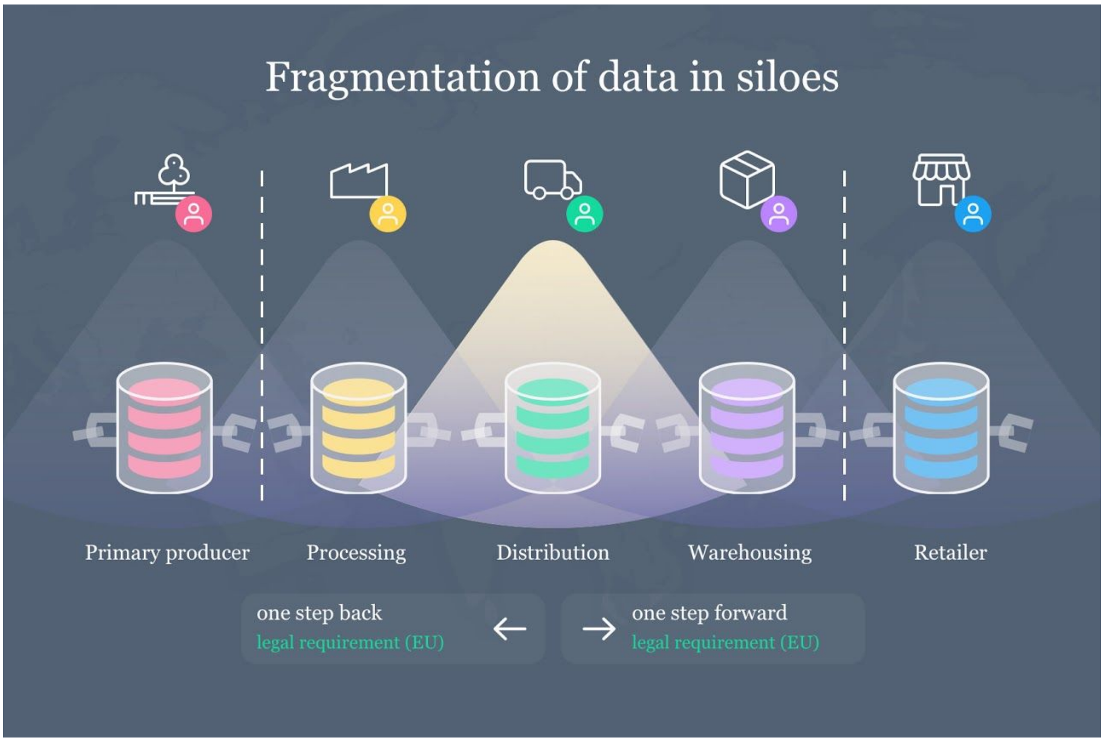
3.1 Fragmentation of data in siloes and opacity of supply chains

The current state of supply chain data management solutions involves a number of localized information systems, ERP systems and custom solutions. In order for them to communicate, custom integrations need to be implemented. Often referred to as "data silos", these centralized systems lack a common technical environment, security, and exchange protocols to facilitate data sharing.

Because of this **low interoperability of data** and other technical hurdles (e.g., different security policies, separate infrastructures and environments), useful real time knowledge on supply chain product context has not been available to interested stakeholders (e.g., consumers, certification and governmental bodies, and operating companies in the supply chain). With each stakeholder in the supply chain receiving and sending data about product attributes "**one step back, one step forward**"[3], trust is easily broken and value chains integrity compromised.

Many organizations today aim to bring more order and integrity to complex supply chains, including global standard providers in supply chains (e.g., GS1), certification organizations (e.g. Global GAP, ISEAL, Rainforest Alliance, Bureau Veritas) and information systems providers (e.g. IBM). Yet, none of these organizations can ensure entire chain integrity by creating a stand-alone solution **due to centralized logic of data collection and sharing**. Typically, only parts of global supply chains get audited and involved which leads to partial data collection, poor verifiability of collected data, and eventually diminished trust.

[3] The supply chain is a series of separate operations in sequence (raw production to market), each linked by the products supplied to them from a preceding operation (the one step back or down) and the products they supply to the next operation (the one step or forward). Each operator in the chain records information which links the separate operations with their own traceability system. Each operator is responsible only for the stage of the chain under their control.



3.2 Shortcomings of current decentralised solutions

Blockchain based, open and decentralized solutions are highly compatible technologies to overcome the above mentioned challenges. **However, none of the current solutions provides high performant functionalities of storing, processing and interacting with highly interconnected data that is inherent to supply chains**. Solutions such as IPFS and Storj are great for decentralized storage of documents, but they fail to provide the functions needed for advanced search, cumulative analysis and flexibility in handling interconnected data, which is the domain of professional database solutions. Ethereum, IOTA, Hyperledger Fabric and similar solutions are not designed for such data storage and manipulation either, and are comparatively more expensive to operate than traditional centralised database solutions. Finally, BigchainDB provides some database functionalities, though not a fitting, flexible model for supply chains, and has a different intended use in it’s permissioned governance model.

Platform	**Ethereum**	**Hyperledger Fabric**	**IPFS / Filecoin / Storj**	**BigchainDB**
System type	public[4]	permissioned	public	Permissioned
Intended use	General purpose virtual machine, Smart contracts	Customizable blockchain framework for closed environments	Decentralized file storage	Decentralized document database
Data storage cost	High	N/A	low	low[5]
Database functionalities	no	no	no	yes
Suitable for highly interconnected data	no	no	no	no

[4] A public (open) system allows for anyone to enter the network and become a participant (node), without the control of a governing authority. In permissioned (also called federated) systems there is an authority which decides on who can be a participant of the network

[5] BigchainDB states that the cost of indefinite storage of 1GB is $100

4. OriginTrail - First purpose built protocol for supply chains based on blockchain

OriginTrail is a protocol solution **allowing IT providers to easily set up blockchain supported data sharing in supply chains**. It enables building transparency and tracking beyond the “one step down, one step up” principle, protecting brands from fraudulent behaviour and driving efficiencies for all stakeholders.

OriginTrail brings :

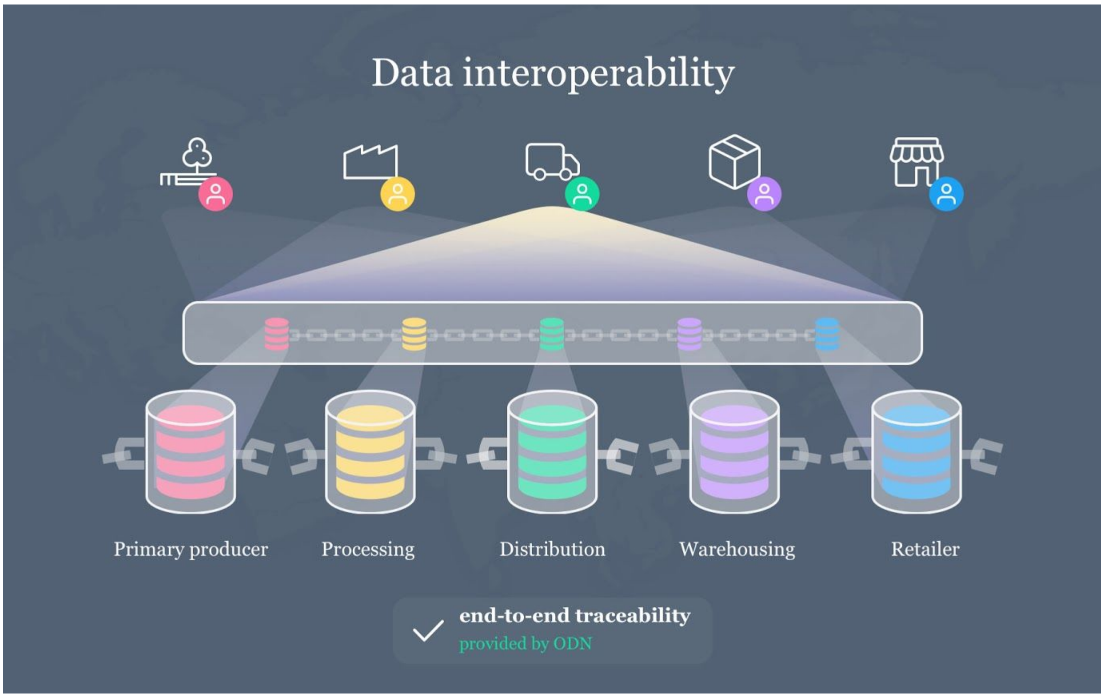
1. **Seamless and automatic data connection and interoperability** between IT systems of different stakeholders in multi-organisation supply chains, with consensus mechanisms for ensuring integrity of data;
2. **A public decentralized solution for performance, cost and scalability issues** by providing a tailored decentralized system for supply chain data based on blockchain.

Direct users of the OriginTrail are therefore **developers creating various supply chain applications** using the described protocol. Users can be third party technology providers (supply chain software companies, ERP providers, IoT providers, software development companies) or in-house supply chain technology teams. Applications where OriginTrail’s protocol delivers value are:

- product authentication,
- product journey visibility,
- product recall efficiency,
- product freshness for perishables,
- chain of custody with accountability,
- CSR activities support,
- supply chain mapping and optimisation,
- inventory management,
- alert systems (exception management),
- supply chain compliance assurance,
- customs, audit and regulations process optimisation,
- and any other supply chain application that requires transparent supply chain as a starting point.

4.1 Automatic data connection and interoperability beyond the "one step back, one step forward" principle

OriginTrail protocol enables exchange of different data sets between multi-organisation supply chains no matter its complexity while ensuring the data quality and integrity. Input and sharing data with OriginTrail is based on a common set of data standards which allow multiple organizations (companies involved in production, distribution or retail of goods) to exchange data beyond the "one step back, one step forward" principle.



4.1.1. Data interoperability format

In order to provide for a uniform data flow, all information must be standardized within the ecosystem. While the XML is a widely adopted file format for data exchange, content within the file must be also standardized. Supply chain can span across the globe, where each member has its own local standards. For example, date and time formats are very different even in neighbouring countries. Date 01/10 can be the first of October in one system, and the 10th of January in another. This defines the challenge that data sent to OriginTrail must be standardized, and vice versa. This requires standardization not only the attributes and nodes within attributes of XML file, but the content also.

OriginTrail adopts widely used GS1 standards. GS1 standards support the information needs of end users interacting with each other in supply chains, specifically the information required to support the business processes through which supply chain participants interact.

OriginTrail supports data such as, but not limited to:

- **Master Data**: data shared by one trading partner to many trading partners, that provides descriptive attributes of real-world entities identified by GS1 Identification Keys, including trade items, parties and physical locations.
- **Transaction Data**: trade transactions triggering or confirming the execution of a function within a business process as defined by an explicit business agreement (e.g., a supply contract) or an implicit one (e.g., customs processing), from the start of the business process (e.g., ordering the product) to the end of it (e.g., final settlement), also making use of GS1 Identification Keys.
- **Visibility Data**: details about physical or digital activity in the supply chain of products and other assets, identified by keys, detailing where these objects are in time, and why; not just within one organisation's four walls, but across organisations.

OriginTrail is focused on the EPCIS framework[1] because it suits the protocol in its core foundations. The framework is designed to be:

- **Layered**: In particular, the structure and meaning of data in an abstract sense is specified separately from the concrete details of data access services and bindings to particular interface protocols. This allows for variation in the concrete details over time and across enterprises while preserving a common meaning of the data itself. It also permits EPCIS data specifications to be reused in approaches other than the service-oriented approach of the present specification. For example, data definitions could be reused in an EDI framework.
- **Extensible**: The core specifications provide a core set of data types and operations, but also provide several means whereby the core set may be extended for purposes specific to a given industry or application area. Extensions not only provide for proprietary requirements to be addressed in a way that leverages as much of the standard framework as possible, but also provides a natural path for the standards to evolve and grow over time.
- **Modular**: The layering and extensibility mechanisms allow different parts of the complete EPCIS framework to be specified by different documents, while promoting coherence across the entire framework. This allows the process of standardisation (as well as of implementation) to scale.

Other data sets will include IoT and compliance data. This allows for exchanging and tamper-proof recording of product properties, which leads to accountability and data integrity in product supply chains.

4.1.2. Data consensus check as a tool for trustworthiness

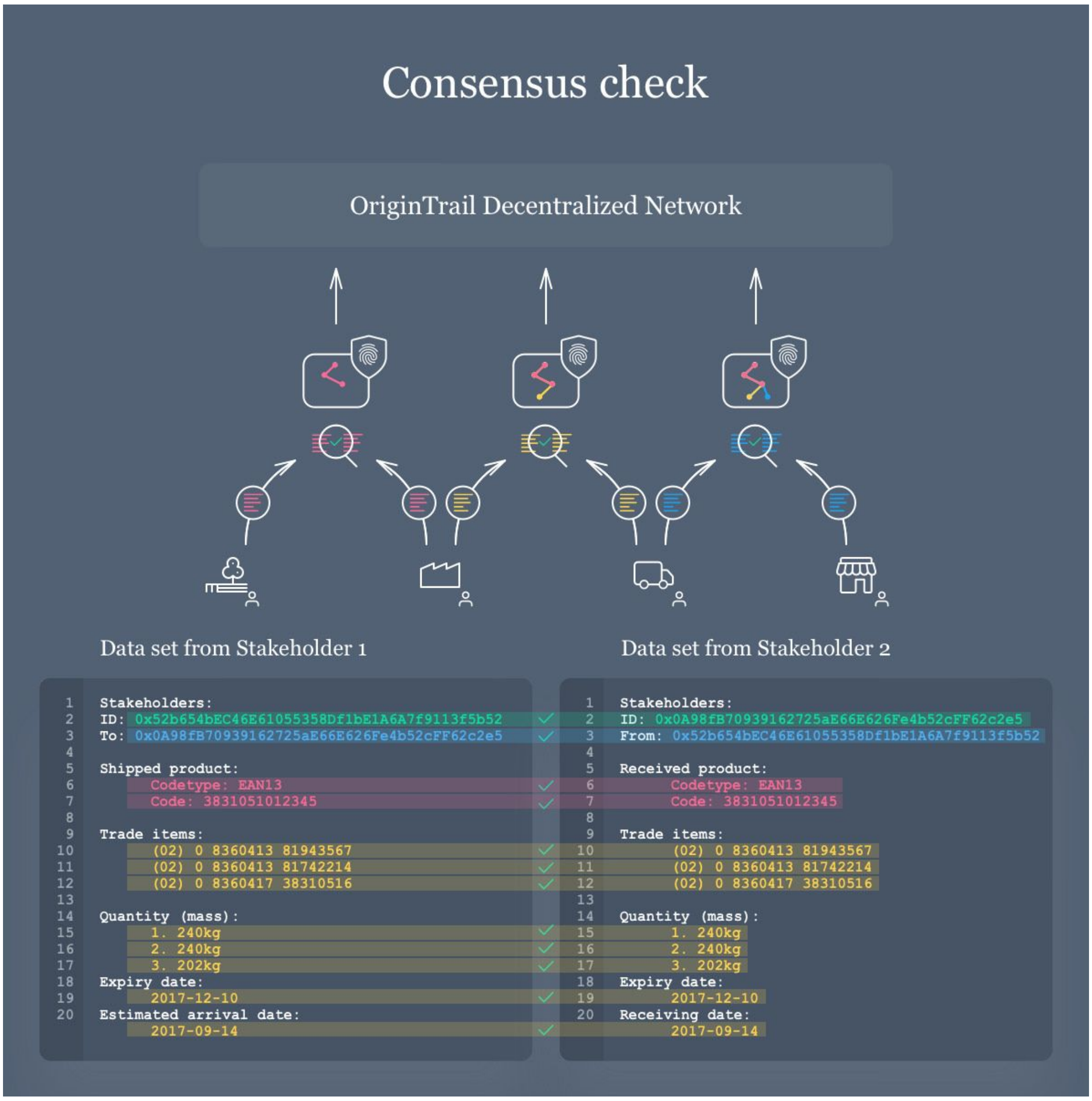
When receiving information from stakeholders, OriginTrail protocol performs a “consensus check” that verifies there are no discrepancies between data provided by different stakeholders. The check is performed in several steps:

Step 1. Each stakeholder has to be approved by the previous and the following supply chain stakeholder, creating a chain of accountability.

Step 2. Matching of dynamic batch information is verified, including the critical information of batch identifiers, appropriate timestamps and transactional data. As this step involves company private data (e.g. quantities of sales), a **Zero Knowledge Proof[6] mechanism implementation will provide a way to check that private information matching is provable without revealing the information itself.** Other dynamic data may include data collected from sensors and compliance data.

Step 3. As an additional layer of credibility, auditing and compliance organisations can validate data by supplying their confirmations.

[6] In cryptography, a **zero-knowledge proof** or **zero-knowledge protocol**[2][3] is a method by which one party (the *prover*) can prove to another party (the *verifier*) that a given statement is true, without conveying any information apart from the fact that the statement is indeed true.



Consensus check

Data set from Stakeholder 1

```
Stakeholders:
ID: 0x52b654bEC46E61055358Df1bE1A6A7f9113f5b52
To: 0x0A98fB70939162725aE66E626Fe4b52cFF62c2e5

Shipped product:
      Codetype: EAN13
      Code: 3831051012345

Trade items:
      (02) 0 8360413 81943567
      (02) 0 8360413 81742214
      (02) 0 8360417 38310516

Quantity (mass):
      1. 240kg
      2. 240kg
      3. 202kg
Expiry date:
      2017-12-10
Estimated arrival date:
      2017-09-14
```

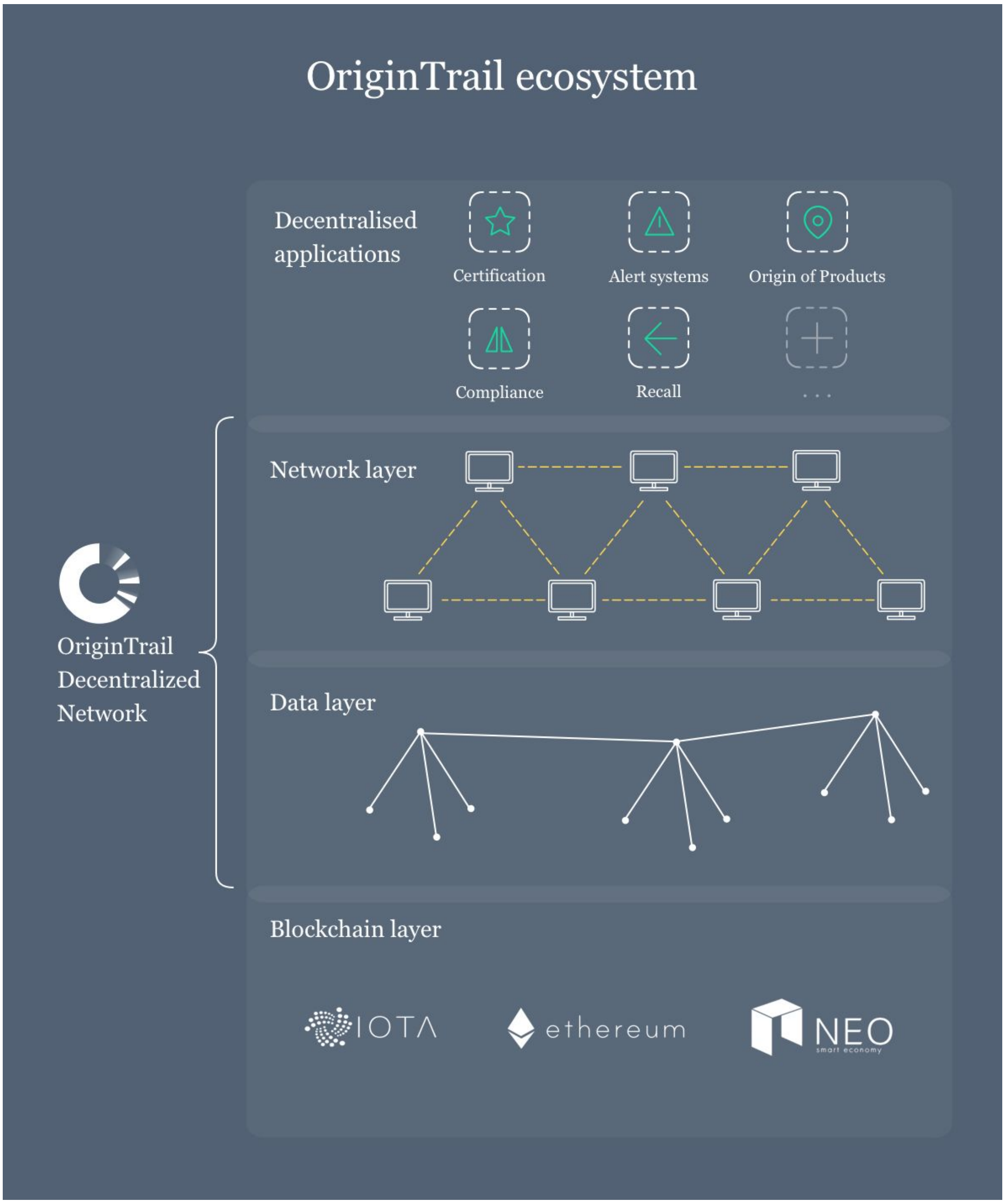
Data set from Stakeholder 2

```
Stakeholders:
ID: 0x0A98fB70939162725aE66E626Fe4b52cFF62c2e5
From: 0x52b654bEC46E61055358Df1bE1A6A7f9113f5b52

Received product:
      Codetype: EAN13
      Code: 3831051012345

Trade items:
      (02) 0 8360413 81943567
      (02) 0 8360413 81742214
      (02) 0 8360417 38310516

Quantity (mass):
      1. 240kg
      2. 240kg
      3. 202kg
Expiry date:
      2017-12-10
Receiving date:
      2017-09-14
```

This ensures the entire supply chain is in accords regarding that batch of products. If there is no consensus, discrepancies can be quickly reported, investigated and reconciled. Reconciliation of discrepancies is also recorded on OriginTrail - the additional information is uploaded as a special “reconciliation” data set which is again subject to the same consensus mechanism.

4.2 OriginTrail Decentralized Network

In order to provide the optimal solution we implement the OriginTrail protocol that runs on an off-chain decentralized peer to peer network, called the OriginTrail Decentralised 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 in order to reduce cost and inefficiency.



OriginTrail incorporates **blockchain as the platform to ensure data integrity**. For all the information that gets included in the system, a tamper proof "fingerprint" (a cryptographic hash) is generated and stored on the blockchain at the time of arrival. The cryptographic hash is commonly used to prove the received data has not been modified in any way, and having the hash immutable in blockchain as a reference to the original input completes this mechanism. If there is a need to check if data was tampered with, a simple hash comparison between the stored hash in the blockchain and the newly generated hash from the same data in ODN shows if changes have been made.

OriginTrail supports many different blockchain implementations. The current version of OriginTrail utilizes Ethereum blockchain to provide proof of concept and initial implementation, the fully developed solution will provide interfaces to many different blockchains.

There are **multiple reasons** for adopting this principle:

- competing blockchain solutions will evolve in unexpected ways, which will influence the pricing of blockchain usage,
- more advanced blockchain solutions in the future could be integrated,
- supply chain stakeholders already using blockchain solutions for various purposes will be able to use the same blockchain for OriginTrail.

On top of the blockchain layer are two system layers - the network and data layers, which combined form the ODN system. Because of computational and storage efficiency, the ODN is able to deliver a cost-efficient solution for data integration and manipulation in for supply chain stakeholders.

4.2.1 The 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 factor to focus on in the data layer**.

In order to leverage data relationships in the most efficient way, the system needs a database technology that stores relationship information as a first-class entity[4]. The technical solution which fulfills this requirement is a graph database. Below is a comparison of different data storage solutions.

	File storage	**Key-Value stores**	**Relational databases (Row & Column based)**	**Document databases**	**Graph databases**
Example	File system, IPFS, Storj	Redis, Riak, Voldemort	HBase, Cassandra	MongoDB, CouchDB	Neo4j, ArangoDB, OrientDB
Pros	- Simple - low-cost	- Simple data mode - Optimized for simple lookups - Scalable	- Naturally indexed - Optimized for aggregation - Scalable	- Simple, powerful data model - Scalable	- Powerful data model - Optimized for connections - Easy to query
Cons	- Poor for interconnected data	- Poor for complex data - No foreign keys - Poor for interconnected data	- Poor for interconnected data	Poor for interconnected data - Query model limited - Map/Reduce for larger queries	- Less efficient data aggregation

Graph databases provide high performant traversing, high flexibility in terms of data models and thus high agility when it comes to development. They are already utilized in enterprises like Walmart, eBay, the adidas Group[5] and many other companies for various use-cases involving online retail.

Furthermore, supply chain information is inherently graph-like, both in terms of the flow of products as well as connections that this data forms. Using a decentralized graph database is therefore what provides great conditions for:

- interoperability, as graphs can be extended and modified easily with low operational cost
- high performance, as graph databases are great for quick traversal and connection forming
- high availability, because of distribution

Each specific product supply chain is presented with it's own graph. Once data is inserted, it can no longer be changed - only additional information can be added to the graph. Every time new data is introduced to the graph, a new graph fingerprint (cryptographic hash) is created and stored on the blockchain layer. This allows for a versioned graph, where each step in the graph growth can be later verified for credibility against the immutable fingerprints present on the blockchain.

It is important to acknowledge that graph databases, with all their strengths, are not a silver bullet for every possible data processing scenario. We are currently experimenting with several different graph DBs, of which some are multi model databases that would allow for even more data manipulation flexibility - i.e. more efficient cumulative analysis. It is also important to restate that these data processing scenarios are not the main problem OriginTrail is aiming to solve.

4.2.2 ODN Network layer

The network layer takes care of the accessibility and data governance of the underlying data layer. It consists of network nodes which all contain parts of the decentralized database and store graphs of the data. Access to the data is achieved through the provided data exchange API.

The peer to peer network is built on a distributed hash table based on Kademlia[6] which is responsible for efficient routing within the network. The messages between peers are signed, while the Kademlia node ID presents a valid Ethereum address which the node is able to spend from. This enforces long-term identity and helps with Kademlia routing and Eclipse attacks.

The network distinguishes between two types of nodes in regards to their interaction with supply chain data - data creators (DC) and data holders (DH). It is important to state that DC and DH nodes are the same in terms of their system capabilities, but are rather viewed differently in the context of the data they hold in the system in regard to specific supply chains.

A data creator node is responsible for injecting supply chain data into the network and replicating it over a specific number of data holder nodes. This distinction comes as a form of data governance decentralization - for each n DC nodes that are involved in a specific supply chain, an additional number of at least $n+1$ DH nodes are selected to keep the specific data replicas. All these nodes keep a copy of the data graph in their local databases (a data creator is also a data holder), so in total this means that the minimum ODN replication factor is $2n+1$, where n is the number of supply chain involved nodes.

In this way the system attempts to ensure that the DC nodes are always outnumbered by the DH nodes to minimize the possibility of collusion between supply chain actors who might have an interest in changing an incriminating piece of information inside the supply chain data graph they have assembled.

4.3 Data input

To introduce a data set into OriginTrail a standardised interface format is utilized, either via a web service or via an XML file provided to the OriginTrail importer (new implementations of the file importer can be created separately to allow other formats if needed as well - this is in the domain of service providers developing solutions on top of OriginTrail). Both mechanisms of input have been tested and improved upon in a production environment since the initial centralised version of OriginTrail has been set up in 2014.

Once a new data set is introduced to a DC node, it performs:

1. the initial data format check in regards to both syntax and semantic errors (regarding data standards). In case of an error it is reported back in the importer log and through the web service response
2. Following the check, the data set is converted into graph form in the database, called the Proto graph, which is identified uniquely by the product and batch identifiers
3. The node then attempts to introduce the new graph to the network via the data distribution protocol
4. If successful, the data will get merged into the master graph of the product supply chain, while being validated via consensus check and fingerprinted on the blockchain layer

Definitions for further reading

Vertex v^i - Graph node; ordered pair of node identifier vid^i and D^i data contained in the graph node

$$v^i = (vid^i, D^i), vid^i \in \mathbb{N}$$

Set V^i - Set of graph vertices

$$V^i = \{v_1, v_2, ..., v_k\}, k \geq 0$$

Relation r^i - Graph edge, relation; ordered tuple of connecting nodes and D^i set of key-value pairs

$$r^i = (v^{start}, v^{end}, D^i)$$

Set R - Set of graph edges; relations between the nodes

$$R = \{r_1, r_2, ...r_l\}, l \geq 0$$

Graph G - Supply chain graph; tuple of sets V and R and graph owner identifier

$$G = (V, R, oid), oid \in ID$$

Set Σ^i - Set of supply chain graphs stored on data node i

$$\Sigma^i = \{G_{i,1}, G_{i,2}, ..., G_{i,s}\}, s \geq 0$$

Node N^i - Network node with id_i which operates on the graph set Σ^i

$$N^i = (id_i, \Sigma^i), \in \{0, 1\}, id \in ID$$

Set N - Set of network nodes

$$N = \{N^1, N^2, ..., N^k\}$$

Set W - Set of node public keys

$$W = \{wid_1, wid_2, ..., wid_e\}, e \geq 0$$

4.4 Data distribution protocol

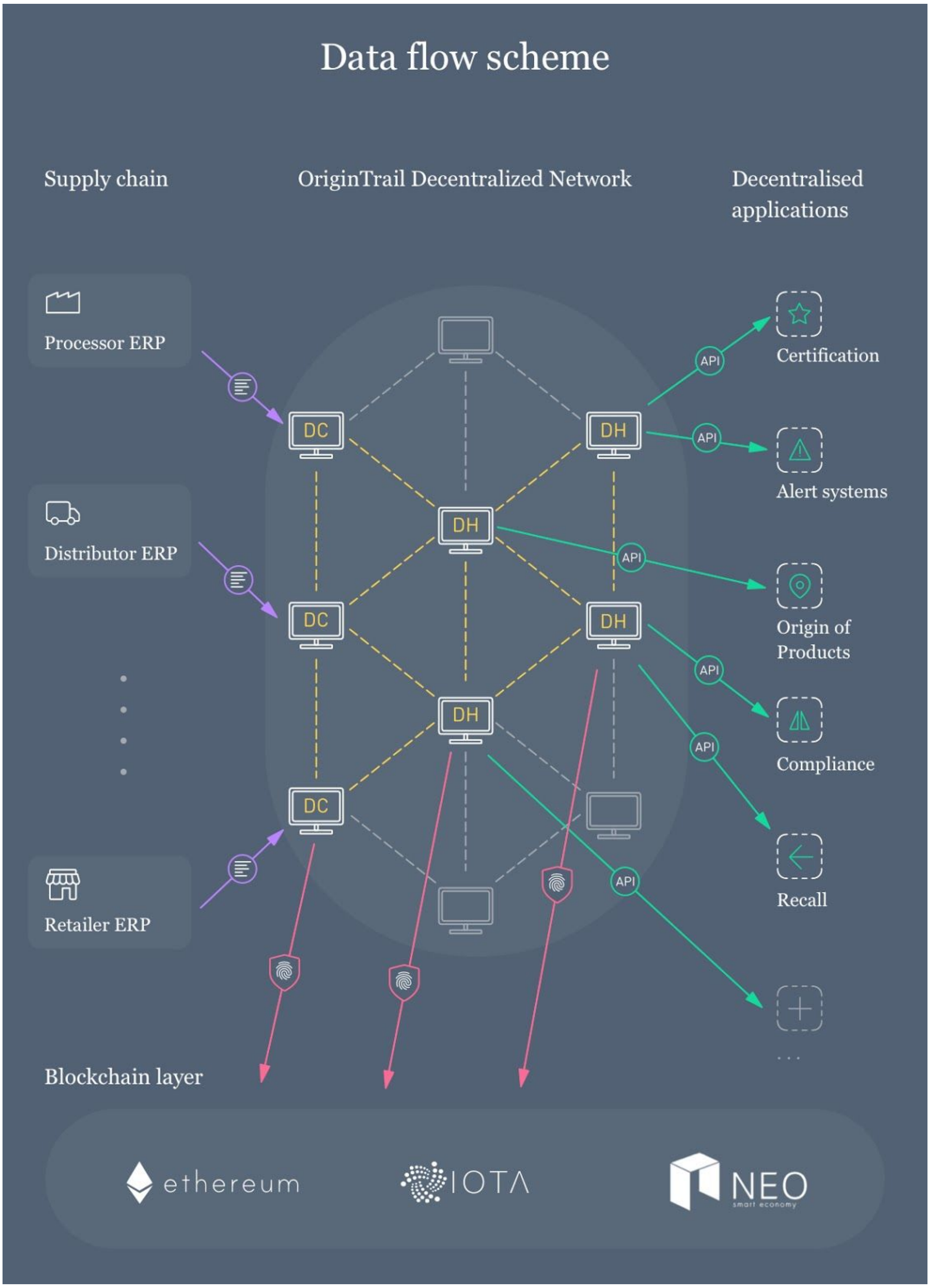
When a DC node N^i is requested to upload a supply chain graph G_i, that contains the information on product P, it first checks the blockchain layer to see if there are any nodes storing the master graph G^P data on product P. If the nodes are not found, N^i broadcasts a request for storage service of their supply chain graph data for a product P. When service proposals are received from data holder nodes, N^i selects $2n$ of nodes with the best proposals, where n is the number of actors involved in the OriginTrail supply chain data sharing. N^i notifies every selected node N^j, where $j = 1 \text{ to } 2n$ about the proposal acceptance and encodes supply chain graph G_i with the encoding function Enc_j[7].

After encoding G_i, N^i sends the encrypted graph $G_{i,e} = Enc_j(G_i)$ and performs an initial proof of storage test for node N^j to initiate the service according to the accepted proposal. Each node N^j then includes a hash fingerprint of the received decrypted graph G_i to the blockchain layer for data integrity proof.

If there already are data nodes storing the G^P, N^i fetches graph G_e^P from a selected data node, decodes it with the same key used for encoding G_e^P and checks the compliance of G^P with the hash h_{G_P} stored on the blockchain layer. If the check is successful, N^i then performs the consensus check between graphs G_i and corresponding G_{i-1} and G_{i+1} (if existent), which are part of G^P.

When supply chain data consensus check is successfully completed for a new connected graph $G_P^{'} = G_i||G_P$, the DC node stores the newly formed graph hash $h^{'}{}_{G_P}$ and broadcasts the graph G_i to other selected DH nodes which perform the same algorithm, except further propagating through the network. The result is that each of the $2n$ DH nodes, and the additional DC node all keep the same data graph $G_P^{'} = G_i||G_P$ with their own computed data fingerprint hash on the blockchain layer.

[7] The function *Enc* is a function of the node's address (public key) and current time



Reading data from OriginTrail involves requesting the information from a node with a specific set of input information depending on product identifiers and potentially many other parameters (such as batch IDs, timestamps, industry specific information, unique product packaging identifiers, sensory information etc).

A data node which has received a read request for a specific subset of data involving G^P, first checks for the location of DH nodes holding G^P in the blockchain layer. In case the node itself is the DH, it responds with the data immediately from their local database, otherwise it forwards the request to the closest DH node in the network that does hold the requested information.

4.5 Possible system attacks

Sybil & Outsourcing Attacks

Creating multiple (Sybil) identities would theoretically allow for malicious nodes to pretend to store more copies of the same data, but having them stored only once and quickly fetched from the storing location when required to prove they are providing the service. This issue is addressed by establishing a similar mechanism to the Proof-of-Replication[7] introduced in the Filecoin whitepaper[8] applied to graphs, with the consideration that the data in OriginTrail is public by design. With these preconditions in place, encryption is used to prove replication and not used to obscure data - it is up to the data creator to encrypt the input information they require to be obscured before inputting data into the system.

Therefore, each node N^k in OriginTrail is required to periodically prove that they are storing a specific encoding E_k of the graph G, where each encoding is distinguishable and incompressible. The encoding function Enc has to be a slowable PRP (pseudo-random permutation) with cipher block chaining that is:

- slow enough so that it is easy to distinguish the time between an honest and a malicious response, as the attacker would need to first encrypt and then transmit G through the network
- arbitrarily tunable in terms of running time, running at $O(nt)$, where t is the number of times each graph element is encrypted before proceeding to the next (sequential encryption where the result of each encrypted element is input for the encryption of the next element) and n is the number of elements in the graph
- Publicly verifiable, which is achieved by using a unique cryptographic key based on the node public key w_k
- Has a quick inverse decode function so that decoding can be done quickly (with parallelization at $O(t)$)

The 51% Attack

A 51% attack is usually defined as an ability to control an overwhelmingly large amount (at least 51%) of power in a decentralized system (i.e. hashing power in Ethereum), which then grants the ability to manipulate data. In terms of data integrity in OriginTrail, such an attack is not a problem as for each graph G it is deterministically verifiable that the data hasn't been changed by comparing the hash h_{G_P} extracted from the DH node with the cryptographic fingerprints in the blockchain layer. Additionally, DH nodes are incentivised to store G^P in its proper form in order to be able to prove storage and receive compensation. If a node fails to provide proof of retrievability / replication, it can be easily substituted by another node in the system by the data creator.

Byzantine faults

Byzantine faults are defined as faults caused by nodes to deliver supply chain graph data either by being unavailable or having an incorrect data response. An incorrect response is defined as a response that cannot be validated by an appropriate hash fingerprint on the blockchain layer of OriginTrail. Because of the datan governance consensus of replicating the graph data in $2n + 1$ data holder nodes (where n is the number of distinct supply chain data creator nodes), the probability of failure to deliver the requested data significantly diminishes with the number of involved nodes in the exchange.

When a DH node fails to deliver service for a required period of time, the data distribution protocol is used to find a new candidate node and replicate the data to keep the required number of copies on the network.

Eclipse attacks

Isolating a node or a multitude of them from the network by having all outbound connections reach malicious nodes is called the eclipse attack. This is addressed by using public key hashes as node IDs in Kademlia. To eclipse a node on the network the attacker has to generate key pairs that position themselves closer in Kademlia to the targeted node than it's nearest non-malicious neighbor, as well as maintaining that position when new nodes join with closer IDs. This problem grows in complexity as more nodes are introduced to the network and essentially presents a form of proof-of-work problem.

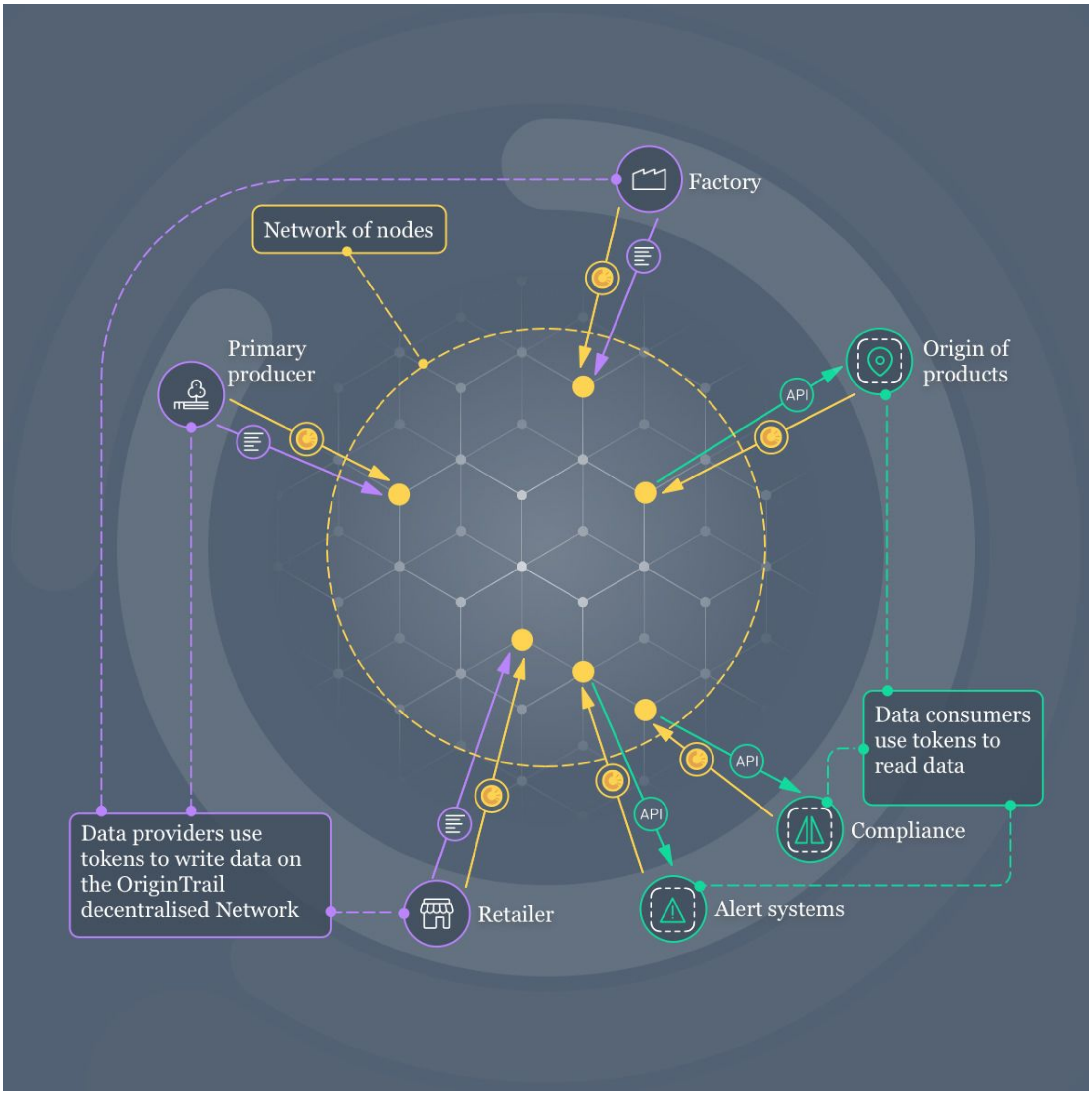
Hostage data attacks

A malicious node might refuse to deliver certain graph data in order to extort data owners for additional tokens. This possibility is mitigated by replicating graph data across a multitude of nodes.

4.6 Trace token economics

OriginTrail ecosystem is enabled by the tokenisation of data exchange and supply chain processing functionalities. The system consists of a network of machines (nodes) that are all running OriginTrail full software clients. Their supply is met by the demand of users of the protocol (supply chain data producers and consumers) that wish to share supply chain data using OriginTrail.

The Trace token is the means of compensation between supply chain data producers and data consumers on one side and the OriginTrail node holders on the other. It provides the incentive to the nodes in the peer to peer network to perform the system functionalities. Maintaining and operating the p2p network takes resources: time, electricity, computing power, storage space and communication bandwidth.


Factory
Network of nodes
Primary producer
Origin of products
API
API
API
Data consumers use tokens to read data
Compliance
Data providers use tokens to write data on the OriginTrail decentralised Network
Retailer
Alert systems

Therefore, OriginTrail nodes are incentivised to do two groups of tasks:

- Data processing - Supply chain consensus checks, data quality and replication checks
- Storing, managing and delivering the data in graph form

Write (introducing new supply chain information to OriginTrail) and storing operations are performed by nodes for which they receive the compensation in Trace tokens based on the agreement reached, in regards to the data distribution protocol mechanism described in this document.

It is important to note that OriginTrail uses a blockchain layer which presents an independent system and thus adds additional cost depending on the chosen underlying blockchain for some OriginTrail functionalities. In case of Ethereum being the underlying blockchain, this means that a small amount of gas (Ether) is also needed to store the necessary hashes on Ethereum for the storage operation.

Read operations are also compensated with Trace tokens. An exception where read operations can be free of cost is if certain conditions are met: if one has access to i.e. an Ethereum node for free reads from Ethereum (or another chosen blockchain from the blockchain layer), and if they hold a local OriginTrail node which contains the necessary graphs.

The amount of tokens to be awarded for the nodes providing the service is a function of supply and demand between nodes and users. Data creators will not be required to pay any additional arbitrary fees apart from what they agree to pay to the nodes. On the other side, nodes will receive full payment of what they have agreed with and provided to the user.

The Trace token is implemented as an ERC20[8] compatible token on Ethereum. This ensures interoperability with wallets and other tokens on Ethereum. The Trace token smart contract handles all transactions and balances in a secure and trusted manner.

[8] There is an emerging ERC223 token specification which improves the ERC20 standard and lowers the cost of token usage - if approved as a standard by the time of token sale, it might become the preferred solution for the Trace token

References

[1] GS1 - EPC Information Services (EPCIS) Standard - Sept. 2016

[2] ZcashCo - What are zk-SNARKs - https://z.cash/technology/zksnarks.html

[3] Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, Madars Virza - Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture, May 2015

[4] Ian Robinson, Jim Webber, Emil Eifrem - Graph Databases, 2nd Edition - New Opportunities for Connected Data, June 2015, O'Reilly Media

[5] Neo4j - Why Graph Databases? - https://neo4j.com/why-graph-databases/

[6] Petar Maymounkov, David Mazieres - Kademlia: A Peer-to-peer Information System Based on the XOR Metric

[7] Protocol Labs - Filecoin: Proof of Replication - https://filecoin.io/proof-of-replication.pdf

[8] Protocol Labs - Filecoin: A Decentralized storage network - https://filecoin.io/filecoin.pdf