Massachusetts Institute of Technology
Media Lab’s Digital Currency Initiative
Sloan School of Management

Choosing and Implementing an Enterprise Blockchain to Solve Supply Chain Challenges

Authors:Dickson Li (MIT Sloan), Yih Lin Teh (MIT Sloan), Roland Xu (MIT Sloan)

Keywords: Supply Chain, AI, Enterprise Blockchain

Blockchain Lab Program
2018-2019 Working Groups Cycle
Taught by N. Narula, G. Gensler, S. Johnson, and M. Casey
Member Company: AT Kearney
Project Group: Blockchain and AI in Supply Chain

MIT
MANAGEMENT
SLOAN SCHOOL

Choosing and Implementing an Enterprise Blockchain to Solve Supply Chain Challenges

Written by: Dickson Li, Yih Lin Teh, Roland Xu
Advised by: Professor Simon Johnson, Eilon Shalev (TA)
*MIT Media Lab Digital Currency Initiativ*e
in partnership with A.T. Kearney
May 14, 2019

Abstract

In this paper, we develop a framework that allows a company to evaluate different enterprise blockchain offerings when looking to procure a solution that addresses solves their supply chain challenges. We recommend companies to consider business factors, technical factors, and implementation factors.

The business factors to consider are the vision and purpose of the blockchain's creation, the governance and track record of the blockchain, and the blockchain's business model. For technical factors, we recommend looking at a blockchain solution in terms of four layers (the fundamental, smart-contract, data storage, and interface layer), and then comparing the nuances of those across the offerings. Finally, we highlight certain implementation considerations for companies.

Along the way, we also evaluate the landscape of supply chain challenges that could be solved by blockchain. We see these challenges as in three main categories - traceability, low digitization, and trust & costly intermediaries. Our evaluation of these problems and the solutions presents a primer for companies who have decided to implement a blockchain solution and are evaluating different offerings.

1. Table of Contents

2. Project Problem & Scope

Companies seeking to understand how blockchain will solve their supply chain problems often have to settle for abstracted explanations of the benefits of a distributed ledger with append-only logs that require multi-party consensus. These abstractions tend to generalize across the varied offerings that characterize the enterprise blockchain landscape. To analyze the costs and benefits of implementing an enterprise supply chain, companies need to understand the business and technical nuances of each enterprise blockchain offering.

In this paper, we answer the question: For companies seeking to solve supply chain problems, how should the different characteristics of blockchain protocols influence their procurement decisions? The goal of this paper is to contextualize this question by specifically mapping companies' supply chain needs to the blockchain platform offerings in the market. We focus on five of the main enterprise blockchains including Corda, Hyperledger, EOS, Quorum, and Ethereum. We framed our investigation using two preliminary questions:

1. What are the universe of supply chain problems which can begin to be addressed by blockchain?
2. How does the blockchain protocol that a platform runs on affect its features and suitability for solving such business problems?

Based on the results of over 15 interviews and various academic papers, we organize our research findings in two sections:

1. The first section (3.1) provides a framework for identifying different supply chain problems that blockchain could potentially address. Companies could look at the examples of problems described here and find analogous situations in their own supply chains.
2. The second section (3.2) provides a a primer on enterprise blockchain and the describes established benefits of applying blockchain solutions. It also directly answers the motivating question of what a company should consider when implementing a blockchain solution for its supply chain problems. We established three areas that one will need to consider when implementing blockchain solutions for supply chain issues:
    a. Firstly (3.2.1), we outline the business considerations of different blockchain protocols, such as the company or community behind the development of the blockchain.
    b. Secondly (3.2.2), we break down the technical layers behind a "blockchain solution" and explain the implications of the technical layers for businesses. This helps companies to understand what purchasing a "blockchain solution" entails, and gives a technical understanding of the functional differences of different enterprise blockchains.
    c. Finally (3.2.3), we articulate certain implementation realities based on our conversations with various providers.

3. Research Findings

3.1. Identification of supply chain problems

The benefits of blockchain to supply chain generally summarized under four established categories - disintermediation, transparency with pseudonymity, security, and automation (Yli-Huumo et al., 2016; Gupta, 2017; Iansiti and Lakhani, 2017). In this paper, we look at the other side of the coin to propose a framework of four categories of supply chain problems blockchains have been known to address. These are *traceability*, *low digitization*, and *trust & costly intermediaries*.

Traceability: This represents the largest use-case area for blockchain in supply chain. To provide further structure and clarity to this area, we further divide it into three sub-areas. These sub-areas generally align with the different stakeholders in a supply chain:

- *Quality Assurance*: Primarily of concern to supply chain **operators**, this covers issues such as maintain cold-chain storage conditions.
- *Provenance*: Primarily of concern to supply chain **consumers**, this covers issues such as ascertaining the origin of rare minerals used in consumer electronics.
- *Fraud Prevention*: Primarily of concern to supply chain **investors**, this covers issues such as proving the quality and existence of inventory for financing purposes.

Low Digitization: The benefits of digitizing paperwork in supply chains are, in general, massive. At present, many aspects of supply chains are analog and depend on paperwork, and blockchain in and of itself does not necessarily enable digitization as there are non-technological reasons (i.e. no clear incentives to make capital investments, force of habit, historical quirks) for not digitizing paperwork or analog processes. However, the problems we focus on in this paper show how blockchain is necessary to solve specific types of digitization - such as solving the double-spending problem in transferring bills of lading.

Trust & Costly Intermediaries: In supply chains, intermediaries play an important and expensive role in connecting different parties in the supply chain (e.g. ship brokers, banks in trade financing). Removing intermediaries in many of these relationships can be more of a human than a technological challenge - for example, one consultant for a major blockchain implementation company shared: “I spend a lot of my time convincing clients that even <<blockchain company name redacted>> cannot access their data on the blockchain.” Therefore, de-intermediation has to be contextual and specific to each individual relationships. In this paper, we show how specific blockchain characteristics enable arrangements to be made in low-trust environments.

3.1.1. Traceability

a) Quality Assurance

Industry	Problem	Detail
Quick Service Restaurants (QSRs)	Cold-chain temperature monitoring	GSF transports fresh beef to QSRs. Piloted tagging its products with RFID, IoT to monitor temperature, and blockchain to define business rules in supply chain. This assures all operators in the supply chain that the temperature of the fresh beef remained within the safe zone.
Automotive	Brake pad recalls	Brake pad recalls are expensive. In the event a problem surfaces, manufacturers have difficulty tying a specific brake pad to a specific supplier.
Food	Food recalls	In the event a food recall is necessary, tracing the exact source of the product takes a long time. As a precautionary measure, a large volume of the product is discarded - this causes waste.

b) Provenance

Industry	Problem	Detail
Food	Responsible sourcing	Customers want to know which farms their Thanksgiving Turkeys are sourced from. Cargill provided a way for customers to trace their turkeys.
Automotive	Responsible sourcing	Car manufacturing process uses a lot of lithium / rare metals. The more pressure they put on suppliers, the more suppliers start to source from undesirable providers
Automotive	Airbag recalls and quality certification[1]	An estimated 0.1% of cars have counterfeit airbags. Repair shops who purchase airbags from brokers have no way of determining authenticity.

[1] M. Ahlers, Feds Warn of Counterfeit Airbags Being Installed as Replacements, CNN, 2017.

c) Fraud Prevention

Industry	Problem	Detail
Logistics	Fraud in commodities exchange	People who sell commodities misreport what time commodities changed hands; price of commodity changes and the parties involved profit privately
Agriculture	Verifying inventory warehouse receipts	SMEs who need to obtain financing using their inventory as collateral use warehouse receipts as proof. These receipts are paper-based and open to fraud[2]

3.1.2. Low Digitization: Title Transfer, Payments, etc.

Industry	Problem	Detail
Shipping	Bills of lading / purchase orders are paper-based	Bills of lading have to be couriered between shipping and receiving parties before ownership of goods can be transferred.
All	Different suppliers in a supply chain need to reconcile funds to each other	Settlement between suppliers can sometimes take a long time, or is done through costly channels.

3.1.3. Trust & Costly Intermediaries

Industry	Problem	Detail
Agriculture / Industrials	Enforcing inventory transparency with suppliers	Although complete transparency of inventories can help with inventory optimization across the supply chain, Cargill's commodity suppliers do not want to provide full transparency as this allows arbitrage within the commodity markets.

[2] Chod, Jiri and Trichakis, Nikolaos and Tsoukalas, Gerry and Aspegren, Henry and Weber, Mark, Blockchain and the Value of Operational Transparency for Supply Chain Finance (September 15, 2018)

3.2. Solving supply chain challenges with enterprise blockchain solutions

The benefits of blockchain to supply chain are generally summarized under four established categories - disintermediation, transparency with pseudonymity, security (through immutable records), and automation (Yli-Huumo et al., 2016; Gupta, 2017; Iansiti and Lakhani, 2017). Theoretically, the issues in traceability can be addressed if immutable records for a product resided on the blockchain, and all stakeholders in a supply chain could access those records. The issues in digitization of paperwork can be addressed with smart contracts coded into the blockchain. Similarly, the issues with costly intermediaries if business contracts and relationships were coded into a set of automated rules in smart contracts on the blockchain.

These abstracted solutions have been explored extensively in media and the public space. This paper does not attempt to outline the merits of using a blockchain in supply chain; it is meant to provide guidance on the business, technical, and implementation considerations for companies who have already decided that they want to implement a blockchain solution.

Enterprise blockchains, in contrast to non-enterprise blockchains, are concerned with creating a system of record keeping between multiple parties in an industry who may not fully trust each other to achieve consensus on a set of shared facts. Many enterprise blockchain solutions exist today for solving supply chain challenges such as traceability, digitization and trust. In implementing enterprise blockchain solutions, users need to be cognizant of a number of considerations that have implications on how these solutions can deliver functions appropriate for their business needs. We have identified three sets of considerations that will be important for blockchain applications in supply chains:

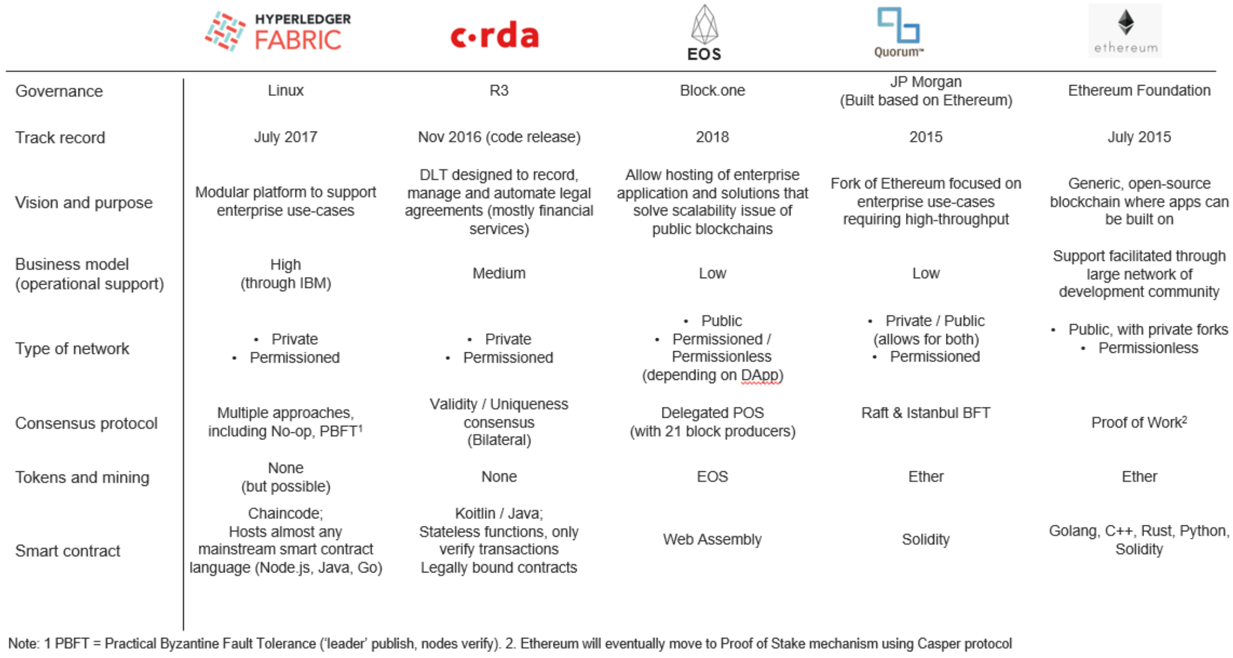
Business considerations: The governing entities implementing blockchain solutions drive the fundamental structure and evolving needs of the network of which the blockchain community resides. Business considerations such as initial vision for chain development and track record should play a critical role in users’ decision making process as it affects functionality and continuity of the chain. There are also traditional IT procurement considerations to be made here, including an evaluation of the responsiveness of the development and support team.

Technical considerations: Blockchain solutions consist of several different layers, and different blockchain solutions have different specifications and features in each layer. When adopting blockchain solutions, one should examine these specifications and features, and evaluate if, for example, the type of network, consensus mechanism, etc. are well-suited for the use-case at hand.

Implementation considerations: Various challenges have been found to determine the success of blockchain solution deployment in the supply chain context. When implementing these solutions, one should be cognizant of issues such as bridging physical assets and digital representations; data standardization, protection and privacy; collective action, etc.

The figure below provides an example high level comparison of five popular blockchain platforms (Hyperledger Fabric, Corda, EOS, Quorum and Ethereum) that have been adopted for supply chain use-cases. The following subsections then describes in further detail the three areas of considerations as described above.

Figure 1: Mapping of operating systems against business and technical considerations (fundamental and smart contract layers only)



	HYPERLEDGER FABRIC	corda	EOS	Quorum	ethereum
Governance	Linux	R3	Block.one	JP Morgan (Built based on Ethereum)	Ethereum Foundation
Track record	July 2017	Nov 2016 (code release)	2018	2015	July 2015
Vision and purpose	Modular platform to support enterprise use-cases	DLT designed to record, manage and automate legal agreements (mostly financial services)	Allow hosting of enterprise application and solutions that solve scalability issue of public blockchains	Fork of Ethereum focused on enterprise use-cases requiring high-throughput	Generic, open-source blockchain where apps can be built on
Business model (operational support)	High (through IBM)	Medium	Low	Low	Support facilitated through large network of development community
Type of network	• Private • Permissioned	• Private • Permissioned	• Public • Permissioned / Permissionless (depending on DApp)	• Private / Public (allows for both) • Permissioned	• Public, with private forks • Permissionless
Consensus protocol	Multiple approaches, including No-op, PBFT[1]	Validity / Uniqueness consensus (Bilateral)	Delegated POS (with 21 block producers)	Raft & Istanbul BFT	Proof of Work[2]
Tokens and mining	None (but possible)	None	EOS	Ether	Ether
Smart contract	Chaincode; Hosts almost any mainstream smart contract language (Node.js, Java, Go)	Koitlin / Java; Stateless functions, only verify transactions Legally bound contracts	Web Assembly	Solidity	Golang, C++, Rust, Python, Solidity

Note: 1 PBFT = Practical Byzantine Fault Tolerance ('leader' publish, nodes verify). 2. Ethereum will eventually move to Proof of Stake mechanism using Casper protocol

Source: Team analysis based on various blockchain platform whitepapers.

3.2.1. Business considerations

(i) Vision and purpose

The development of different enterprise solution is driven by very different visions in mind. This can impact the underlying structure in which the blockchain is formed (e.g., consensus framework, how transactions are handled), which is an important consideration when applying solutions to use-cases.

As an example, we compare three popular enterprise blockchain solutions that currently exist: Hyperledger Fabric, Corda, and Ethereum. The development of Fabric and Corda are driven by concrete use cases. Corda to-date focuses primarily on financial services and has applications in financing and settlements, while Fabric intends to provide a modular and extendable architecture that can be employed in various industries. Ethereum, on the other hand, presents itself as independent of any specific field of application, and is known for flexibility of deployment across "any" use-case.

Depending on the complexity of the use-case, users should take into account how their requirements are being met by the existing functionalities a blockchain platform is able to offer, as well as the flexibility built into these structures to cater to new functionalities in the future.

(ii) Governance and track record

The market has observed the entrance of many new 'proprietary' blockchain solutions, building on existing codes of Ethereum or other existing blockchains. In addition, with AWS Blockchain Templates, anyone can quickly set up quickly set up Ethereum- or Hyperledger Fabric-compatible blockchain networks. AWS templates are certified to work as open-source frameworks, which allow easy, instant deployment and configuration of blockchain software that helps a business create their own instance of a decentralized network as per their choice.[3]

While this offers a wide range of options to users, many of these 'newer' chains lack infrastructure in comparison to 'more established' chains. When considering these new chains, users need to be cognizant of issues such as:

- Risk of discontinuity given insufficient number of users or potential change in priorities of governing entities
- Small network of developer community leading to low number of DApps or slow evolvement or lack of updates to the chain
- Interoperability challenges across applications and chains
- Inconsistency with latest protocols if 'chain updates' are not implemented well (e.g., solutions that are based on other codes but with some tweaks / functionalities changed)

Another aspect of governance that should be considered is 'ownership' and incentives of other users to use the chain. For example, given that Quorum is governed by J.P. Morgan, there may be push-back and conflict of interest from other competing financial services providers in adopting the Quorum chain.

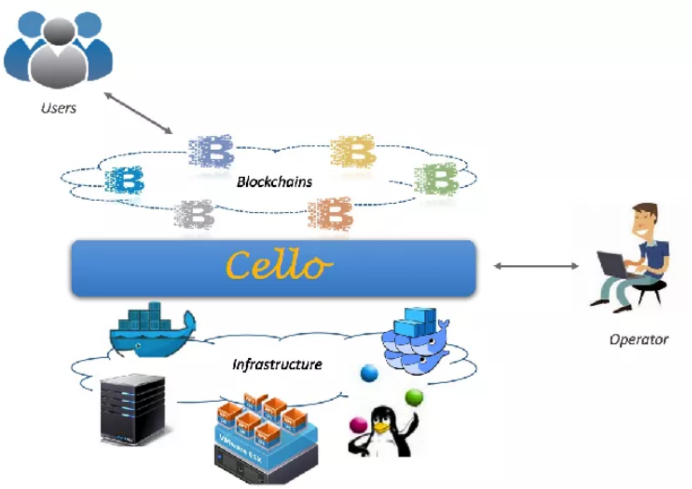
(iii) Business model

A range of business models exist, each with different degree of operational support to users.

Some service providers, such as IBM, offer a Blockchain-as-a-Service (BaaS) as part of their cloud service offerings. The primary platform used by IBM is Hyperledger Fabric. Companies like IBM can afford to dedicate resources towards operational support to its clients, to manage tasks and activities that keep the infrastructure agile and operational.

[3] https://aws.amazon.com/blockchain/templates/

Figure 2: Example of BaaS[4]



This to some extent alleviates technical complexities and operational overhead involved in creating, configuring, and operating the blockchain, and maintaining its infrastructure, which often act as a deterrent for enterprise blockchain adoption. However, not having the right technical knowledge in-house across a corporate or consortium that adopts enterprise blockchain solutions may deter proactive productive use of the blockchain which could lead to obsolescence quickly.

In addition, IBM typically builds both the blockchain platform as well as the applications (see figure 3 below), whereas R3 works with a partners across multiple industries to develop applications on the Corda Platform[5]. For one company, IBM's end-to-end solution involving the use of Hyperledger Fabric might be suitable, while for another, a turnkey solution developed by one of R3's partners built on the Corda platform might be more suitable.

3.2.2. Technical considerations

An enterprise blockchain solution consists of two elements - the blockchain operating system (or blockchain 'platform'), and the DApps (Decentralized Applications) that run on the operating system.

Operating System: The operating system provides the fundamental structure for any blockchain solution. It establishes a platform whereby developers can develop and deploy high-level applications. The goal of designing such general platform is to achieve flexibility of the platform itself so that they could directly bridge the benefits of blockchain technologies to developers, instead of through specific

[4] https://www.investopedia.com/terms/b/blockchainasaservice-baas.asp

[5] Based on interviews with industry experts

use cases. An example of this type of solution is the famous Ethereum, which is the first and most popular operating system platform provider in the blockchain world.

DApps: DApps build upon operating systems to design solutions for specific use cases. DApps provides both the data storage layer and user interface layer to allow users of the network to interact with one another, and receive and send information. ShipChain is an example of a service provider implementing a DApp supply chain solution - they are developing and implementing end-to-end logistics platforms for companies within the shipping value chain.

Across these two elements, there are four layers that make up a blockchain solution.

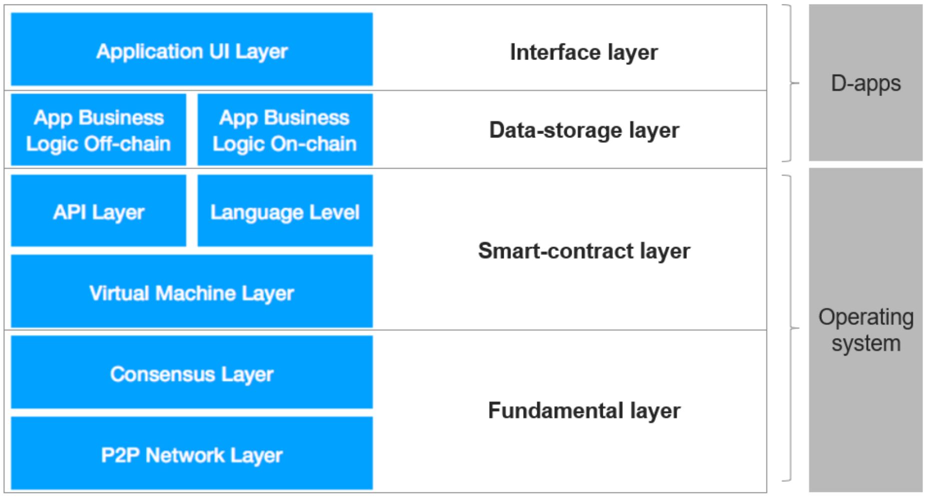
Within operating systems, there are typically two main layers:

- *Fundamental layer:* This layer consist of a P2P Network Layer, which is the core unit for inter-block communications (similar across most blockchain solutions); and a Consensus Layer, the most crucial in blockchain technology that governs the creation and validation of blocks by nodes in the network.
- *Smart-Contract layer:* This layers provides flexibility and infrastructure for developers to deploy their own applications on the selected blockchain operating system. There are a number of different smart contract programming languages, and these programming languages are operated upon the virtual machine layer for each blockchain operating system.

Within DApps, there are typically two main layers:

- *Data-storage layer:* Two types of storage mechanisms exist: either on-chain, or off-chain solutions. Some DApps can also use a mix of on- and off- chain storage that interacts with each other.
- *Interface layer*: This layer is where the digital applications reside. Other than the logic that operated on the smart-contract (on-chain application), this layer is usually treated as an off-chain application layer, which refers to all the client-side or server-side development codes for DApps.

Figure 3: Four layers of blockchain solutions



Note: Left hand side of figure above is directly sourced from [*https://github.com/pandoraboxchain/blockchain-abstractions-layers*](https://github.com/pandoraboxchain/blockchain-abstractions-layers)

We have attributed two of the four layers (fundamental and smart-contract layers) to operating systems and the other two layers (data-storage and interface layers) to DApps, as this is how majority of blockchain applications in the market works. However, the development of solutions are constantly changing - for example, some operating system service providers are now incorporating data-storage layer within their operating systems. In reality, there is also no clear line between where the operating system 'ends' and where DApps 'begin'.

For simplicity, we organize our research findings on technical considerations based on the four layers described. We pay more attention to the first two layers related to operating systems, given the interest we have observed in these layers. When assessing operating systems, we primarily focus on how various operating systems might differ from each other, and why that matters in supply chain contexts. We found, in our research, that the data-storage and interface layers are more implementation-specific and as such we focus on the operating system layers.

(i) Fundamental layer

Within the fundamental layer, there are three key aspects to consider:

- Type of network - Does the solution require a private or a public blockchain? Should it be permissioned or permissionless?
- Consensus protocol - What is the protocol for achieving consensus across multiple parties?
- Token mining - What are the economic incentive mechanisms for creating new blocks?

Type of network

The fundamental layer determines the type of network the blockchain provides: (i) a public/private network and (ii) permissioned/permissionless network. These are defined as the following[6]:

- *A public blockchain* is a blockchain, in which there are no restrictions on reading blockchain data (which still may be encrypted) and submitting transactions for inclusion into the blockchain.
- *A private blockchain* is a blockchain, in which direct access to blockchain data and submitting transactions is limited to a predefined list of entities.
- *A permissionless blockchain* is a blockchain, in which there are no restrictions on identities of transaction processors (validators that are eligible to create blocks of transactions).
- *A permissioned blockchain* is a blockchain, in which transaction processing is performed by a predefined list of entities with known identities (example Banks).

On the surface, private blockchains are able to provide a 'higher degree' of privacy, given that unlike public blockchains where anyone can access information, the data on private blockchains is only accessible to those who are within the private network. In the context of supply chains, majority of trade information is likely to be proprietary and sensitive, leading to the need for using a private blockchain.

However, there have been instances of providing privacy through creating additional protocols within a public blockchain, such as introducing privacy features can be introduced within the DApp or interface layer of a public blockchain. Ultimately, there are a number of different ways to achieve privacy mechanisms required by supply chain users, and there is no hard rule for why a public or private blockchain should be chosen over another purely on the basis of privacy.

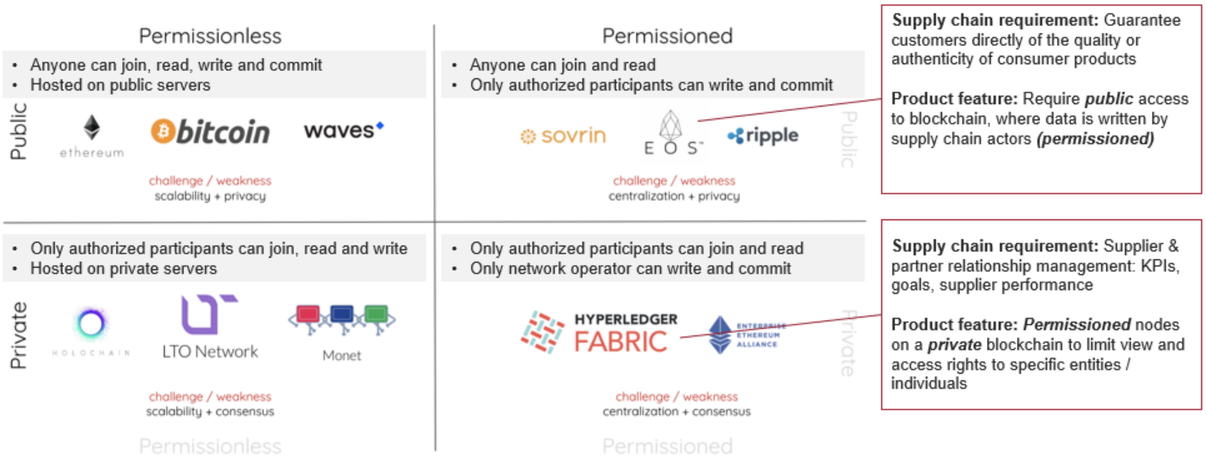
It is important to note, however, that in some cases, private chains may limit use-cases for the solution. In situations where public access is critical to the use case (e.g., involves a public rating / review system where public users play the role of writing to the chain), a private blockchain may not be suitable.

Private chains could also have heightened security risk, given the cost of multiple entities colluding to create forks within the chain is much lower compared to a public chain. In these instances, one can increase the security of private chains by hashing block headers that are intermittently submitted to a permissionless chain, publicly available, such as Bitcoin.

Most supply chain applications will likely require a permissioned framework, regardless of whether it is hosted on a public or private blockchain. Permissioned blockchain solutions allow identification and control of known participants (suppliers, retailers, distributors) that can be provided with the authority to access, edit *or* verify information on the blockchain.

[6] Source: http://bitfury.com/content/5-white-papers-research/public-vs-private-pt1-1.pdf

Figure 4: Mapping of private v. public and permissioned v. permissionless blockchain solutions



Permissionless

- Anyone can join, read, write and commit
- Hosted on public servers

Permissioned

- Anyone can join and read
- Only authorized participants can write and commit

Supply chain requirement: Guarantee customers directly of the quality or authenticity of consumer products

Product feature: Require ***public*** access to blockchain, where data is written by supply chain actors ***(permissioned)***

- Only authorized participants can join, read and write
- Hosted on private servers

- Only authorized participants can join and read
- Only network operator can write and commit

Supply chain requirement: Supplier & partner relationship management: KPIs, goals, supplier performance

Product feature: ***Permissioned*** nodes on a ***private*** blockchain to limit view and access rights to specific entities / individuals

Source: Mckinsey (June 2018), https://medium.com/ltonetwork/the-rise-of-private-permissionless-blockchains-part-1-4c39bea2e2be

Consensus protocol

The consensus protocol governs the mechanism in which multi-party agreement is achieved, for data to be written to the blockchain. The goal of any consensus mechanism is to achieve reliability in a decentralized system, where malicious actors or faulty processes may be present. The consensus protocol provides the mechanism in which the validation process takes place, of blocks created by other nodes in the network.

There are many existing consensus protocols. The dominant five are proof of work(PoW), proof of stake(PoS), delegated proof of stake(dPoS), proof of authority(PoA) and other practical byzantine fault tolerance(PBFT). The first four mechanisms are widely used in the public chain setting, whereas the last one is only used in the private chain setting.

Details on a few examples of popular consensus mechanisms are provided below:

- *Proof-of-work (POW).* POW is the original consensus mechanism adopted by BitCoin (a public blockchain network), that achieves consensus through a mining protocol. Miners are rewarded with incentive tokens in exchange for the work/energy consumption that they have done to maintain the network.
- *Proof of stake (PoS).* In a POS based network, participants on the network can verify the validity of new blocks based on the amount of their stakes they own in the network.
- *Delegated proof of stake (D-POS)*. D-POS is a variant of POS, which allows network participants to select a group of block witnesses that will be responsible for verifying transactions and maintaining blocks. The voting mechanism is typically based on the rate of one vote per share per witness.
- *Proof of Authority (PoA).* PoA is another variant of proof of stake. Instead of giving more voting power to those who have more coin holdings, it uses the idea of authority or reputation to decide individuals’ voting power in the network.

- *Practical byzantine fault tolerance(PBFT).* In PBFT based network, there exist leader nodes that publishes information in the network. This information is broadcasted to other back-up nodes that can confirm the information being published.

The consensus protocol plays an important role for governing the interaction and dynamics between multiple users of the blockchain within supply chain applications. Some considerations when deciding between different consensus mechanisms for supply chain blockchain applications include:

- *Cost of operating and maintaining platform.* While Proof-of-Work (POW) provides a high degree of security and trust, it is a a very expensive way of maintaining the blockchain. In cases where a blockchain platform is used for 'within-enterprise' application, or amongst a set of actors that have some degree of trust and relationship, such stringent requirement for consensus may not be required.
- *Relationship between users.* In certain supply chain use-cases, not every participant on the blockchain is required to sign-off in order to reach consensus. In these situations, mechanisms such as federated consensus can be used. This mechanism allows participants in a network to delegate the responsibility of verifying legitimacy of a transaction or block creation to a selected set of signers.
- *Transaction speed.* Some supply chain decisions often require real-time information to be processed and received by multiple parties. In these cases, it will be important to select a consensus mechanism that balances the requirement for security /reliability and the speed that it enables. The table below provides an indication of 'theoretical' transaction per second (TPS) for selected cryptocurrencies employing different types of consensus mechanisms.

Figure 5: TPS for selected cryptocurrencies[7]

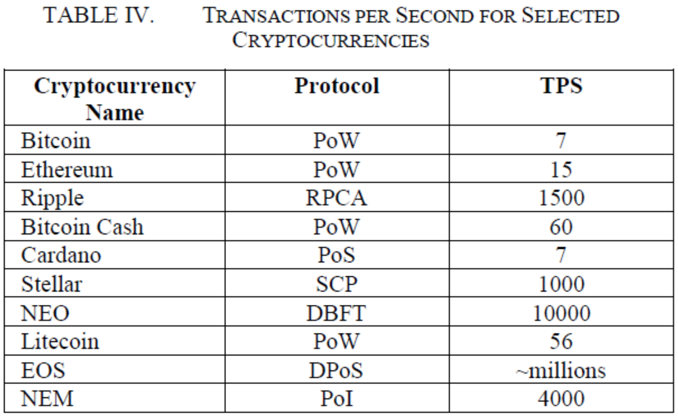


TABLE IV. TRANSACTIONS PER SECOND FOR SELECTED CRYPTOCURRENCIES

Cryptocurrency Name	Protocol	TPS
Bitcoin	PoW	7
Ethereum	PoW	15
Ripple	RPCA	1500
Bitcoin Cash	PoW	60
Cardano	PoS	7
Stellar	SCP	1000
NEO	DBFT	10000
Litecoin	PoW	56
EOS	DPoS	~millions
NEM	PoI	4000

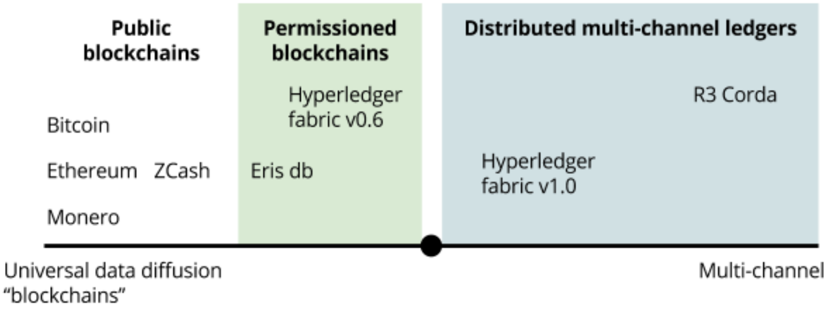
- *Information sharing.* The consensus protocol can also determine the type of information that is shared between participants on the blockchain. One interesting mechanism to point out is one being deployed on the Corda blockchain, which is maintained using a 'bilateral' consensus

[7] "Comparative Analysis of Blockchain Consensus Algorithms" Bach, Mihaljevic & Zagar (2018)

mechanism. While other platforms generally reach consensus at a ledger level, using Corda, consensus over transaction validity is performed only by parties to the transaction in question. Therefore, data is only shared with those parties which are required to see it. Thus, any given actor in a Corda system sees only a subset of the overall data managed by the system as a whole. This is useful in supply chain contexts where sensitive data can only be shared between two parties that are transacting.

A more generalizable form is displayed in Figure 5: in which the public blockchains like Bitcoin have universal data diffusion, while R3 Corda has multi-channel (or more specifically, bilateral) diffusion of data.

Figure 5: Different levels of data diffusion (universal v. multi-channel)[8]



- *The reliability of consensus mechanism.* The adversary tolerance - or the fraction of the network that can be compromised without the consensus mechanism being affected - differs across the different mechanisms. Provided below are the approximate thresholds[9]:

Consensus mechanism	**Adversary Tolerance**
Proof of Work	25% - 50%
Proof of Stake	Depends on how the PoS algorithm is defined
Practical Byzantine Fault Tolerance	<=33%

[8] https://medium.com/@colin_/thoughts-on-the-taxonomy-of-blockchains-distributed-ledger-technologies-ecad1c819e28

[9] "Understanding Blockchain Consensus Models", Dr. Arati Baliga, Whitepaper

Tokens and mining

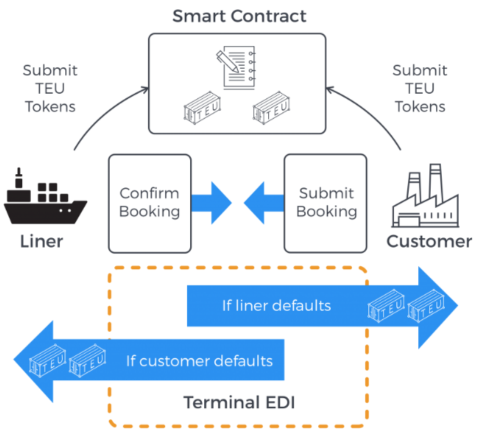
Mining is a supportive procedure to the consensus mechanism. Mining is designed to validate the legitimacy of a transaction and avoid the double-spending problem. Miners are rewarded with cyptocurrencies, or tokens, for performing verification tasks.

Tokens implemented on public and private chains have different implications. Tokens on a public chain can be used to incentivize the whole community of network participants, and these tokens itself can be publically traded on cryptocurrency exchanges, which makes their value subject to speculation. On the other hand, tokens on private blockchains are rarely traded on blockchain exchanges and they are mainly designed to calculate rewards to the internal contributors within the private network.

Most blockchain applications in supply chain do not require a native token. While there exist a few specific use-cases for tokens in supply chain contexts, attempts to utilize tokens to-date have been very experimental. Some potential use-cases include:

- *Providing incentive mechanisms to deliver more 'high value' product to customer.* One idea that emerged from our research is to link tokens to commodities (e.g., cobalt), where those that are 'more-responsibly sourced' could be sold for a premium. This could incentivize producers to employ better labour or sustainability practices, given they are able to sell premium commodities for a higher price.
- *Create new voting mechanisms for Delegated-Proof-of-Stake consensus mechanisms*, where participants can pay tokens to vote, but the cost of voting exponentially increases as the individual pays for more votes (Eximchain).
- *Create accountability mechanisms in shipping trade.* 300cubits is in the process of issuing a digital currency called TEU Tokens, as a shipping booking deposit executed through a Booking Deposit Module to tackle the industry pain points of No Show and Rolling. The figure below provides a visual for how 300cubits intend for TEU Tokens to work.

Figure 6: Example of utilizing tokens in the shipping industry



Source: 300cubits

(ii) Smart-contract layer

Smart contract code simply denotes software written in a programming language. It acts as a software agent or delegate of the party that employed it with the intention that it fulfills certain obligations, exercises rights and may take control of assets within a distributed ledger in an automated way. Thus, it takes on tasks and responsibilities in the distributed ledger world by executing code that models or emulates contract logic in the real world, though its legal justification may be unclear[10].

We provide here a summary of the different smart-contract languages that exist, from Github[11], for an understanding of the landscape:

- *Ethereum virtual machine(EVM) languages. The most popular smart contract language is Solidity. Others include Serpent and its offspring pre-released Viper. Some highly-experimental attempts languages are Bamboo (state-machine functional-type language) and Babbage (visual programming language) both being presented at DEVCON3.*
- *WebAssembly-compiled languages. Due to LLVM-to-WASM, this includes the whole rich family of LLVM languages, including C/C++, Swift, Python, Ruby, Rust and many, many others. Unfortunately, modern blockchain VM do not support the whole set of LLVM instructions, not mentioning language standard libraries, without which language usage could become quite painful. Thus, these are quite early experiments not ready for real-world projects yet.*
- *CLR languages. CLR is "Common Language Runtime" that includes all .NET-based languages from Microsoft (C#, F#, VisualBasic.NET etc). Currently they are supported by the Neo blockchain, and, probably, we will see more support in the nearest future.*
- *Functional smart contract languages, derived from variants of original functional languages like Haskell etc. These languages put big emphasis on formal verification methods, proving that smart contracts will function as expected at their design. Briefly, this approach can be called "it will work if it was compiled" approach.*
- *Non-Turing complete languages for specific blockchains, like ones used by OMNI Layer and Counterparty.*

Within the smart-contract layer, there are a few key aspects to consider:

- *Flexibility of smart-contract language* . Ethereum is well-known in the developer community to provide a flexible smart contract layer that allows for almost any type of use-case. This plays an important role for adopters, given its flexibility allows for interoperability with different DApps providing different functionalities. Hyperledger Fabric is another example of a blockchain operating system that provides high degree of flexibility in its programme language (called chaincode).
- *Reliability of smart-contract language.* Going back to the Ethereum example, it uses the ERC-20 token standard, which has been regarded to be the 'gold standard' in the developer community. However, it is not free of fault and bugs. Some issues have previously been reported, leading to losses to participants in the network.[12] That said, little research has been done to date in

[10] "Comparison of Ethereum,Hyperledger Fabric and Corda", Martin Valenta, Philipp Sandner, FSBC Working Paper, 2017.
[11] https://github.com/pandoraboxchain/blockchain-abstractions-layers
[12] https://github.com/pandoraboxchain/blockchain-abstractions-layers

assessing the reliability of different languages.

- *Efficiency of smart-contract language.* Supply chain context may call for complicated computation and algorithms for triggering payments and outcomes. Hyperledger Fabric is one example that adopts smart-contract language that prioritizes efficiency, and it achieves this through supporting 'Go' language, known for its fast compile time.
- *Legal implications.* Some smart contract advocates believe in the ideal of "Code is law" - that smart contracts become legally binding, apart from automatically setting in motion a series of actions once certain criteria is met. Apart from the philosophical considerations of this belief, businesses should pay attention to the interest of regulators in working with certain blockchain protocols to turn the ideal of "Code is law" into a reality. For example, the Monetary Authority of Singapore conducted a pilot project to automate a final settlement system for tokenized currencies across platforms using smart contracts[13]. One can foresee that this settlement system, operated on smart contracts, can eventually become legally binding. For Corda, specifically: Smart language adopted in the Corda platform does not only consist of code, but are allowed to contain legal prose that can be formulated in a way that accounts for highly regulated environments. One might argue that this makes it more amenable to working with regulatory environments.

(iii) Data storage layer

In general, there are two approaches for data-storage:

- *On-chain data storage.* If the general purpose blockchain (ETH, EOS and etc) is used, there are two ways to store data on-chain. One approach is to put additional information regarding the transaction on the transaction itself. This method can only be applied to cases where the amount of additional data is small. The other way is to store it on a blockchain operating system that is specifically designed for data storage, for instance Filecoin, Sia and Storj.
- *Off-chain data storage*. Instead of storing all raw data on the blockchain, one can store only hashes of each data files. The raw data can remain in traditional relational datastores or file systems, e.g., AWS, Google Cloud, Oracle Database. The benefit of this solution beyond just storing it on a cloud or database is auditability - anybody looking to verify the raw data can hash their copy of the data and check that it matches what's on the blockchain. It also allows for any alterations in historical data to be easily detected, given any changes in earlier hashes will affect the hashes for the rest of the chain.

A mix of off-chain and on-chain data storage, or fully off-chain data storage are most suitable in supply chain context. The main reason for disregarding the fully on-chain data storage solution at this stage is due to the challenges relating to scalability and privacy of blockchain data. Due to data sensitivity, supply chain actors are likely to want to keep some of its data off-chain. Given that supply chain involves a large amount of data inputs, it may not be the most efficient to store all of the required data on the blockchain.

[13] http://www.mas.gov.sg/Singapore-Financial-Centre/Smart-Financial-Centre/Project-Ubin.aspx

The following approaches can be adopted for data-storage in supply chain context:

- *Fully off-chain data approach*. Blockchain service providers such as GuardTime and ShipChain provides solutions whereby the user of the blockchain continue taking full ownership of managing its data, however, at specified intervals, the user creates hashes that are then written to the blockchain, to reduce the possibility of tampering and detect data breaches.
- *Mixed off- and on-chain data approach*. Most supply chain use cases require at least some degree of actual data (transaction data or product information) to be written to the blockchain. When developing the data storage layer, it will be important to identify up front across different users of the blockchain which data is explicitly required to be written onto the chain and provide sufficient justification for why those information needs to be accessed, edited, or verified by the blockchain participants.

(iv) Interface layer

This is the layer that hosts digital application interfaces. Other than the logic that operated on the smart-contract(on-chain application), this layer is usually treated as an off-chain application layer, which refers to all the client-side or server-side development codes for DApps, including user-friendly or other client-friendly interfaces.

The relevant considerations for the interface layer include the network of developers who could develop solution-specific DApps for a company's blockchain. For example - as previously mentioned - R3 relies on an ecosystem of developers, support providers, application builders, and systems integrators, amongst others, to develop its offerings on the interface layer. While a similar ecosystem could exist around with Hyperledger as well, Hyperledger Fabric was conceived for corporate and industrial use and IBM has a suite of offerings in the interface layer.

3.2.3. Implementation considerations

Execution is key to implementing any blockchain solution. In the supply chain context, we highlighted a number of implementation considerations, which are detailed below.

(i) Proof vs. Truth.

One key issue raised regarding the application of blockchain solutions in solving supply chain challenges is the fact that the blockchain solution itself does not guarantee the accuracy of the data. Once information is written to the blockchain, however, distributed ledger technology enables tamper-proof records and the provenance of the data.

While blockchain does not seek to resolve the issue of data manipulation at point of entry, any implementation of blockchain solutions in supply chain that ignores this challenge will be obsolete very quickly, as it erodes the very trust that blockchain seeks to instil between multiple parties. Solutions such as using an Internet-of-things (IOT) device as a node to write to the blockchain allows for some degree of bridging this 'physical to digital' gap. However, these methods may not be fool-proof, given the IOT device itself may be subject to tampering.

(ii) Need for collective action.

Many use-cases in supply chain requires coordination across multiple players in an industry. For example, in the shipping industry, smart contracts on the blockchain is able to facilitate more efficient transactions and reduce lead times, but will require all parties (e.g., brokers, truck providers, financier) in the shipping chain to be part of the system. Without a minimum scale of deployment, the value from adopting such a solution may be limited.

Many supply chain solutions also require parties who are previously in competition with each other to cooperate. This is tricky not only because it requires data-sharing amongst competitive parties, but also because it is often difficult to identify one party which the consortium trusts to design and implement the blockchain solution to begin with. In these cases, it is important to establish a neutral third party who can facilitate initial deployment of the solution. Ideally, this party that has a broader representative interest in driving benefits for the industry as a whole, but few such examples exist. A few examples of blockchain consortiums that are currently in place are: Global Shipping Business Network, which connects 9 leading ocean carriers and TrustChain, a collaboration in the jewellery industry led by IBM.

(iii) Standardization of data.

Standardization of data structures is a key requirement for deployment of any supply chain blockchain solution, especially those that utilize smart contracts to facilitate automatic transactions across different parties. This poses a key challenge to many supply chain blockchain use-cases, given there is often no standardized data processes and specifications.

Before deploying any blockchain solutions, one will first need to start with ensuring that data is standardized across the participants of the blockchain to ensure compatibility. Two examples of progress in this area include projects conducted by the International Organization for Standardization (ISO) to provide blockchain standards, and ShipChain, a blockchain service provider who is working with its clients to develop a set of harmonized open data standards across the shipping industry.

(iv) Data protection and privacy.

An important feature of any blockchain solution is that the data written to the chain is 'immutable'. On the one hand, it ensures that data that is written to the chain cannot be tampered with, but on the other hand, it may imply that data cannot be deleted once on the blockchain. This could be a concern in the case where a consumer requests for their data previously stored on the chain, to be wiped. As per GDPR requirements, the service provider of data storage system is legally bound to delete such data. In a distributed ledger, it is unclear how this could be implemented, given there is no one entity that has 'ownership' of the data, and also the fact that the data is stored across multiple servers. Any changes to the data will also 'break' the chain, given it will interfere with hash functions.

In implementing supply chain solutions, users should be aware of the limitations in functionality of blockchain technologies, in respect to data protection and privacy. One could take action by ensuring that no sensitive, private data is written to the chain.

(v) Aligning with customer demand.
Many blockchain applications in supply chain are built upon assumptions that customers today demand transparency in supply chains - e.g., to determine quality, to avoid counterfeit, to ensure that supply chains are sustainable. However, it has not yet been proven that customers value such information, both subjectively as well as on a monetary basis (i.e., it has not been proven that customers would be willing to contribute payments to improvements of sustainability in supply chains as proven by the blockchain).

In implementing any customer-facing (e.g., traceability) types of blockchain solutions, one needs to balance the cost of implementation vs. the value that they will bring to the customers. This is especially so in cases where there exist expert regulatory bodies / certification agencies that can already verify certain attributes of the products. Companies will need to clearly assess the benefits of blockchain beyond what an 'independent' third party could provide.

4. Conclusions

Supply chain challenges often involve the issue of trust in the process of exchanging value and ownership. Given the benefits that blockchain technology provides in disintermediation, transparency with pseudonymity, immutability and automation, it has the potential to alleviate supply chain challenges and improve coordination across supply chain actors.

Many use-cases are currently being explored to solve long-standing supply chain challenges such as traceability, title transfer, payments and reliance on costly intermediaries. These use cases span across many industries (food, automotive, consumer goods, and mining), and from large industry consortiums, multinational companies to disruptive startups. However, many of these use-cases are in early stages of development and are highly experimental. There exist a plethora of protocols and applications that have been built for solving similar issues, but with slightly different approaches.

Key questions remain in adopting blockchain solutions for supply chains. One of these questions is the lack of common frameworks to assess different blockchain protocols and features, and how they map against business needs of companies adopting blockchain solutions for their supply chains. This paper provides an initial framework for considering three aspects for adopting blockchain solutions in supply chains: business considerations, technical considerations and implementation considerations. These considerations are a consolidation of inputs across academia, industry and the blockchain community, and is in no way intended to be exhaustive.

In addition, we believe that further exploration in the following areas will be beneficial for the blockchain community and industry supply chain management practise:

- In some cases, digitization in itself can already deliver huge benefits to supply chain management. Through our research, we have discovered a strong justification for digitizing manual documentation and processes, especially in the shipping industry.
- Data standardization is important for any digitization process, whether adopting a blockchain technology solution or any type of solution. Efforts in standardizing data structures and system architectures for supply chain actors will create significant impact on improving efficiencies of existing practices
- Finally, further work on articulating best practices in developing supply chain blockchain solutions across the various blockchain layers: fundamental, smart-contract, data storage and interface could accelerate adoption of the technology within the supply chain space, as well as guide industry players in building the most feasible and appropriate technologies for their use-case.

Appendix: Resource Report

A1. References

1. “Feds Warn of Counterfeit Airbags Being Installed as Replacements”, M. Ahlers, CNN, 2017.
2. “Blockchain and the Value of Operational Transparency for Supply Chain Finance”, Chod, Jiri and Trichakis, Nikolaos and Tsoukalas, Gerry and Aspegren, Henry and Weber, Mark, (September 15, 2018)
3. “Blockchain Abstraction Layers”, (https://github.com/pandoraboxchain/blockchain-abstractions-layers)
4. “Public versus Private Blockchains”, BitFury Group, Jeff Garzik, (https://bitfury.com/content/downloads/public-vs-private-pt1-1.pdf)
5. “Understanding Blockchain Consensus Models”, Dr. Arati Baliga, Whitepaper
6. “Comparison of Ethereum,Hyperledger Fabric and Corda”, Martin Valenta, Philipp Sandner, FSBC Working Paper, 2017.
7. “The rise of private permissionless blockchains — part 1” https://medium.com/ltonetwork/the-rise-of-private-permissionless-blockchains-part-1-4c39bea2e2be
8. Yli-Huumo, J., Ko, D., Choi, S., Park, S., Smolander, K., 2016. Where is current research on blockchain technology? A systematic review. PLoS One 11 (10), 1–27. https://doi.org/10.1371/journal.pone.0163477
9. Gupta, M., 2017. Blockchain for Dummies. John Wiley & Sons IBM limited edition.
10. Iansiti, M., Lakhani, K.R., 2017. The truth about blockchain. Harv. Bus. Rev. 95 (1), 118–127.
11. “Shipchain whitepaper”, 2017.
12. “Eximchain: Supply Chain Finance solutions on a secured public, permissioned blockchain hybrid”, 2018.
13. “The Corda Platform: An Introduction”, Richard Gendal Brown, 2018.

A2. Interview List

This appendix provides a list of interviews that we conducted during our research.

#	Name	Position	Company
1	Harshvadhan	Founder	kivio
2	Rado Dragov	Business Development	Ambrosus
3	Ben Duignan	Senior Managing Consultant, Blockchain	IBM
4	Raphael Yahalom	Research Affiliate	MIT Sloan School of Management
5	Joann de Zegher	Professor	MIT Sloan School of Management
6	George Calle	Research & Market Intelligence	R3
7	India Wells	Former Project Lead	BMW
8	Aline De Souza Oliveira Pezente	Global Digital Economy Strategy Lead	Cargill
9	Nicos Trichakis	Professor	MIT Sloan School of Management
10	Alisa Di Caprio	Head of Trade and Supply Chain	R3
11	Hope Liu	CEO	Eximchain
12	Felix Shnir	Executive Director	Qurorum
13	Daven Jones	Director, Product Management	Shipchain
14	Lee Bailey	CTO	Shipchain
15	Conor Svensson	Enterprise Ethereum Standards Chair	Enterprise Ethereum Alliance

A3. Interview prompts

This appendix provides a (non-exhaustive) list of interview prompts that were used during our research.

Supply chain problems

- What are the most common problems you see in a plant on a day-to-day basis
- What potential solutions have you adopted to resolve these issues? Why did they work / didn't work?
- Have you used blockchain to solve these issues? How do you think blockchain can solve these supply chain challenges?

Example of use cases

- Please provide some examples of blockchain use-cases in supply chains
- What issue does it address?
- How did you decide who to work with to implement this solution? Why did you choose them / the platform? How is it different to other alternatives?
- Who are the participants on the blockchain? Who are the notaries? How are they appointed?
- What data is stored on the blockchain? What architecture did you adopt?
- Was there a token? Why did you need one?
- What are key successes / challenges?
- In what cases have they found blockchain useful / not useful (revert to other types of traditional databases)? Why?

Blockchain solution features

- Does different supply chain problems require different blockchain solutions? Why, and how?
- What governance aspects or product features are critical to consider in supply chain use-cases?
- How do you think about interoperability across different blockchains?
- What types of smart contracts did you deploy?

Miscellaneous

- Have you encountered the oracle problem in implementing blockchain solutions in supply chain? How did you resolve this issue?
- What are some relevant market dynamics you consider when you advise companies (for consultants)?
- What are some of the more promising private blockchains that you've seen?