Applying blockchain technology for hyperconnected logistics

Wout Hofman, Jacco Spek, Christopher Brewster[1]
1. TNO, Soesterberg, the Netherlands
Corresponding author: wout.hofman@tno.nl

Abstract: *Blockchain technolog recei es a lot of interest and in estments the last three ears. It promises a trusted en ironment for (un)permissioned data sharing. With respect to logistics, enterprises and authorities can (near) real time share state information. Whene er a stakeholder changes the state of one or more objects like discharging a container from a essel, all that ha e access ill kno this change instantaneousl . The Ph sical Internet requires a large ariet of stakeholders to optimi e their capacit utili ation and combine shipments ith the objecti e to reduce costs and emissions compliant ith (inter)national regulations. These stakeholders all need to collaborate and share data to reach these objecti es. This contribution sho s b means of a case that blockchain supports functional requirements for h perconnecti it , but is not et mature enough for large scale application b a large number of (autonomous) objects, indi iduals, and organi ations.*

Keywords: *h perconnected logistics, data transparenc , blockchain technolog , e ent ledger*

1 Introduction

Hyperconnection or universal connectivity is mentioned as one of the most important aspects of the Physical Internet (Montreuil, Meller en Ballot 2013). It encompasses 'super-fast connectivity, always on, on the move, roaming seamless from network to network, where we go – anywhere, anytime, with any device' (Biggs, et al. 2012). Examples of the implementation of hyperconnection can be found in city logistics (Crainic and Montreuil 2016). A hyperconnected world not only comprises individuals with embedded sensors in their smart devices, but includes all types of devices (e.g. vessels, trucks, containers, and trains), where these devices can be considered as assets used for value delivery. Different sensors and supporting communication technology are used for the identification and tracking of different assets. Several research papers identify supply chains and logistics as the main areas for implementing Internet of Things (Atzori, Iera and Morabito 2010) (Gubbi, et al. 2013), like done for the Physical Internet (Montreuil, Meller and Ballot 2013). These developments lead to intelligent objects (Whitmore, Agarwal and Xu 2015) or what is known as ubiquitous computing (Weiser 1991). Cars implementing the NVIDIA chipset1 can be considered as computing platforms, thus implementing ubiquitous computing. Automatic Identification System (AIS) with Global Positioning System (GPS) is for instance used for vessels and barges and trucks have on-board units and CANbus acting as sensors. The introduction of LoRa technology (www.lora-alliance.org) and 5G (Boccardi, et al. 2014) for communication extends battery life of sensors that can be used for machine-to-machine interaction or for instance intelligent cargo or – π-boxes (Montreuil, Meller and Ballot 2013). The combination of ubiquitous computing and long battery life provides the capability for intelligent cargo, where each box can find its way through a logistics network.

Hyperconnection is mostly described in terms of businesses collaborating in chains (Schonberger, Wilms and Wirtz 2009) like the Hyperconnected City Logistics (Crainic and

[1] http://www.nvidia.com/object/tesla-and-nvidia.html

Montreuil 2016) supported by hardware and communication technology providing computational capabilities and level one interoperability (Wang, Tolk and Wang 2009). Neither the information that any two stakeholders have to share, nor their interaction choreography (Schonberger, Wilms and Wirtz 2009) are described. Data integration is required to achieve state awareness (McFarlane, Giannikas and Lu 2016), also known as situational awareness (Endsley 1995). Conceptual interoperability (Wang, Tolk and Wang 2009), which is currently not implemented by supply and logistics stakeholders (The Digital Transport and Logistics Forum (DTLF) 2017), needs to be achieved to support supply and logistics innovations (McFarlane, Giannikas and Lu 2016) (Montreuil, Meller and Ballot 2013). This paper proposes to use blockchain technology for situational awareness because the technology is able to provide a trusted, distributed environment by which agents can share real time state information. Application of blockchain to Internet of Things, which requires processing streaming data, is still in the research phase (Zhang and Wen 2017). First of all, data sharing requirements for the Physical Internet are analyzed, secondly the state of the art of blockchain technology is presented. By means of implementing a case with blockchain technology, the applicability of this technology for logistics is assessed. The case, its implementation by blockchain technology, and a discussion are presented separately. Conclusions will complete this paper.

2 Physical Internet

This section presents a layered approach to the Physical Internet and analyses the requirement for sharing state information in the different layers. A distinction between the physical – and the administrative state will be made, where the administrative state can cause delays in the physical state. Whereas state information is relevant to optimize processes, not all stakeholders are willing to share this data. Data governance is discussed as a separate issue.

This section only addresses the state of a logistics system, not the transaction that leads to a particular state or affects the state (Dietz 2006), although planning the execution of a transaction depends on the state of a relevant part of the logistics system.

2.1 State information and Quality of Service

The Physical Internet combines innovation in logistics by introducing new concepts like bundling and synchromodality, innovation in packaging, the so-called π-boxes, and innovation in autonomous operating assets with innovation in Information and Communication Technology (ICT, (Montreuil, Meller en Ballot 2013)). The Physical Internet is a network of hubs interconnected by corridors for routing of standardized packages (also known as PI-containers) by (semi-)autonomously operating assets like trucks, vessels, and automated guided vehicles. All these objects have particular capabilities and goals, for instance an autonomous truck or barge will be able to transport containers along one or more corridors and a container will have a sensor with (limited) processing capabilities like controlling the temperature setting of the cargo inside and data like its identification and relevant cargo details for handling.

For optimal routing of physical objects in this logistics network and optimal utilization of the network, autonomous objects, hubs, and organizations have to share data (Endsley 1995). Historic patterns of cargo flows and durations of handling by hubs and along corridors, current goals and capabilities, and predicted durations for next legs and hubs for particular cargo need to be available to meet goals of individual packages and at the same time make optimal use of available capacity by bundling. Goals can be related to all objects and cargo, but will differ. Cargo will have a goal to reach a particular destination within a time frame and costs; transport means will have a goal to optimize capacity utilization for trips with minimal

emissions. Capabilities relate to logistics services and potentially timetables and spare capacity on particular trips.

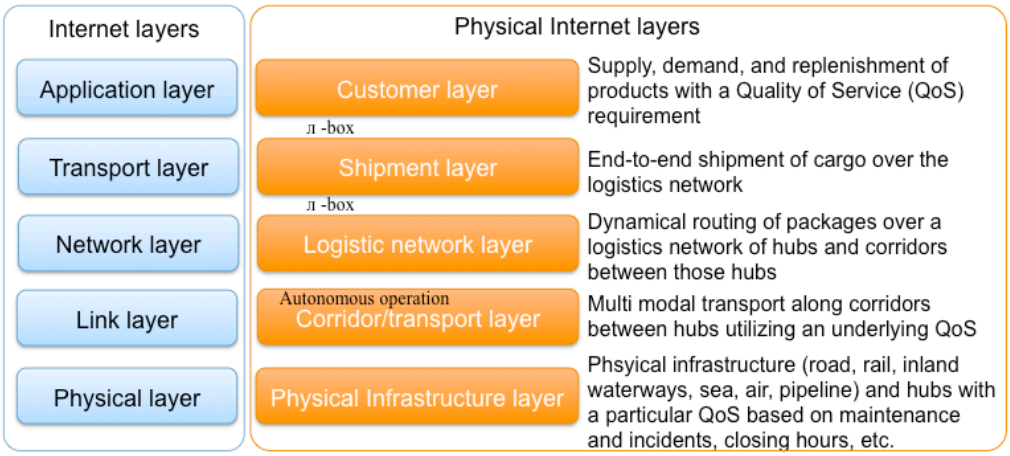

Figure 1: la ering of the Ph sical Internet[2]

Like the Internet (Tanenbaum 1996), various layers can represent the Physical Internet, where each layer adds functionality to the higher layer (figure 1). The concept of Quality of Service (QoS, see also (Tanenbaum 1996)) can be introduced for each layer independent of the implementation of that service by a particular organization. It makes an upper layer service user agnostic of how the service is implemented. For instance, the transport layer provides transport services to the logistics network with different QoS values for each service related to for instance a modality, where the network layer can select the required service based on its QoS. The transport layer hides the QoS of the infrastructure layer, e.g. average delays due to congestion and accidents of a particular modality are hidden to the network layer, thus supporting synchromodality. Standardized Physical Internet – or π-boxes are relevant to the shipment layer; autonomous operation can be in hubs at the network layer and on corridors between hubs with autonomous transport means.

State awareness is applicable to different layers of the Physical Internet. We will distinguish between the physical state of the logistics (sub)system and its administrative state. The state of the physical system is addressed by Intelligent Transportation Systems (ITS). ITS focusses on optimization of infrastructure utilization with vehicle communication (Dimitrakopoulos and Demestichas 2010) like corridor management and optimization of turnaround times (Merrienboer, et al. 2014) combined with autonomous operation of hubs and transportation means, also known as collaborative ITS. The state of the Physical Infrastructure Layer is influenced by its utilization by the Transport Layer. Thus, the Transport Layer affects to the QoS of the Physical Layer. By sharing the goal of a transport means expressed by its position, speed, direction, and expected route of a transport means, the Physical Layer can provide its QoS for that transport means expressed by for instance the Estimated Time of Arrival (ETA) at a destination, a turnaround time at a hub, or vessel departure in a port. ETA and turnaround times are part of the QoS of the Transport Layer on particular corridors or for particular hubs. The Logistics Network Layer utilizes the Transport Layer QoS to decide on routing of cargo through the network. At Customer Layer, individual customers need to be aware of the state of their shipment and have the ability to change shipment flows (McFarlane, Giannikas and Lu 2016).

[2] Inspired by a presentation of Rod Franklin, KLU, ETP Alice WG3, May 2014.

Not only Transport Layer QoS is relevant for routing, additional state data determines the behavior of the system. These can be clustered under the heading trade facilitation (Rukanova, et al. 2011) and supply chain finance (de Meijer and de Bruijn 2014) for cross-border supply chains. Trade facilitation refers to all types of laws with respect to security, safety, illicit trafficking, and tax evasion that result in customs clearance and coordinated border management between various national authorities (Kieck 2010). Improved state information can result in more targeted risk assessment (Heshket 2010), reducing unnecessary inspections that may affect product quality. Supply chain finance refers to payment and liability resulting in the so-called Incoterms (Malfliet 2011). Trade facilitation and supply chain finance include authorities and financial institutions. Since many cross-border trade still relies on paper documents, practical aspects like opening hours of offices can affect the ETA of a transport means, e.g. a barge delivering goods in Switzerland.

2.2 Data governance

Organizations have different reasons not to share state information (Eckartz, Hofman and Veenstra 2014). These can be clustered as ownership and liability, privacy, laws, and commercial sensitivity. For instance, privacy laws prevent authorities to share the position of a barge, since this is also the home of the skipper. Data ownership refers to the actual owner of the data that decides whether or not to make the data publically available. It refers to trust: what happens when the data is shared with others, how will the data be used, and will the owner be liable to any damage caused by actions of others based on the data. An example is sharing the predicted water depth of a river or canal and actions taken by skippers based on this prediction that may lead to accidents. Ownership also refers to cultural aspects: the (un)willingness to share data. Privacy refers to the ability to trace back data to individuals, like the aforementioned position of a barge. Besides privacy laws, other laws are applicable to goods transport that refer to liability, e.g. the Rotterdam Rules for international container transport and the CMR for road transport stating that a carrier should only be liable for damage based on physical characteristics of the cargo and not the actual content. Finally, commercial sensitivity refers to cargo value and identification of cargo flows between origin and destination. Shipment bundling at the Shipment Layer might require sharing origin and destination, which customers might not want to reveal to competitors.

Besides laws governing data sharing, liability and commercial sensitivity refer to trust, which is addressed by blockchain technology.

3 Blockchain technology – state of the art

This section discusses blockchain developments, with a specific focus on Hyperledger Fabric. For the use case, Hyperledger Fabric3 blockchain technology is selected, since it is a permissioned blockchain technology supporting access control required for data governance. Firstly, the state of the art in blockchain technology is presented and secondly characteristics of Hyperledger are introduced.

3.1 Blockchain technology

The last three years have seen an explosion of interest in Blockchain Technology (BCT) with a great many companies and research institutions focusing on potential applications of this technology across a range of financial, industrial and social sectors. The breakthrough that led to the current interest in BCT was the work of Satoshi Nakamoto who wrote the white paper on Bitcoin and released the code (Nakamoto 2008). The underlying technology, the Bitcoin Blockchain, is what has subsequently inspired much work on BCT. However, most research and development has occurred in the context of open source projects such as Ethereum,

[3] https://hyperledger-fabric.readthedocs.io/en/latest/

Hyperledger or BigChainDB, and this work is recorded either in white papers (such as (Wood 2015), (Buterin 2014-2017), (McConaghy, et al. 2016)) or else in blog posts. Specific projects (open and closed source) have also written their own white papers providing details of their approach and sometimes their technical architecture e.g. (Hyperledger 2016) (Greenspan 2015). For example, Provenance.org have described their intention to use blockchain technology as part of their supply chain solution for the agrifood sector (Steiner and Baker 2015).

Characteristics of the technology that have made it so attractive include the following: BCT provides an integration of networks with databases resulting in a peer-to-peer based distributed database spread across multiple entities, with no single owner or single point of failure. It enables to a certain degree an absence of trust because immediate synchronisation ("near real time") across entities means no trusted third party is involved. BCT also provides a permanent record, because due to the inbuilt transparency no record is ever deleted, only appended (hence the "ledger" title some authors use). BCT is distributed and usually decentralised in its conception, meaning there is no single entity that can stop or control operations on the blockchain (specifically true of "permissionless ledgers" where all data is transparent to all users). BCT also makes extensive use of cryptography to prove identity and authenticity using digital signatures, and in some cases to provide perceived anonymity of transactions. The most important technological development since Bitcoin has been Ethereum, which is an attempt to create a blockchain computer to run smart contracts (Buterin 2014-2017) (Wood 2015). The concept of "smart contracts" has been taken up by other platforms such as Hyperledger where they are called "chaincode." One of the key expectations includes the opportunity to develop "distributed autonomous organisations" (DAOs), run by software and entirely outside of the control of any individual or institution, and effectively impossible to "stop".

Putting these characteristics together has made many researchers, entrepreneurs and pundits predict that BCT will revolutionise many different commercial sectors from finance and insurance, through health records and tax collection, to supply chains, the music industry as well as the gambling industry (e.g., it allows the emergence of decentralized casinos and gambling websites (Andrychowicz 2014)).

The use of BCT in logistics has already been proposed, to a limited extent, by a few authors (Smith 2016). Most of the focus has been on the exploitation of BCT to achieve greater supply chain transparency as proposed and implemented by Povenance.org (Steiner and Baker 2015). Badzar has argued for the application of BCT for contract fulfilment (Badzar 2016). Everledger.org has implemented a system for tracking diamonds using a cryptographic fingerprint (Caffyn 2015). Bakker's work showed that experts considered logistics, specifically "smart containers", to be a very good use case for the application of BCT (Bakker 2016). The start-up Blockfreight believes that BCT enables "new era for the digital security, trust, authentication, record keeping and chain of custody data" in logistics. Their solution is built on top of Ethereum and Tendermint and depends on their own cryptocurrency to function.

3.2 Hyperledger Fabric characteristics

Hyperledger Fabric allows for smart-contracts to define function-level access, meaning only certain parties can execute functions within the smart-contract. Hyperledger Fabric uses the term "chaincode" for smart-contracts. This chaincode is a compiled application that is deployed and runs on the blockchain. The goal of Hyperledger Fabric is to be as modular as possible. So in theory it is possible to write smart-contracts for Hyperledger Fabric in any language and compile this to chaincode (a bit like how regular computer applications are

often compiled to assembly, a lower level programming language). Currently there only exists a chaincode compiler for the Go language[4].

The technology is implemented as a network of connected peers, like any blockchain application. These peers all maintain a copy of the ledger and validate any incoming transaction. There are three types of transactions in the Hyperledger Fabric: Deploy- Invoke- and Query- transactions.

Deploy transactions are transactions containing the compiled chaincode, and some additional information (invocation arguments necessary to instantiate the contract, and a list of public keys, of which the owners of the private key can access the smart contract). These transactions deploy chaincode to blockchain. When a peer receives a valid deploy transaction, it generates a unique identifier for this contract and starts a secure docker[5] container running this smart contract. This container is inaccessible to anyone and only interacts with the world by exposing Invoke and Query transactions.

Invoke transactions are transactions that can possibly alter the world-state of a smart contract. A smart contract maintains an internal world-state in the form of a key-value pair storage. Invoke transactions are only added to blocks when the validating peers reach a consensus on these transactions, meaning that they are valid and all yield the same result given the input.

Query transactions are transactions that do not alter the world-state of a smart contract**.** Compared to deploy and invoke transactions, they are quick to execute, since these transactions are not stored on the blockchain. This is not necessary because they do not alter the world-state.

To interact with the blockchain, Hyperledger Fabric exposes a REST API (API: Application Programming Interface) using the gRPC[6] protocol (gRPC is an open source Remote Procedure Call framework developed by Google). End-users can develop their own front-end applications that connect to the API or integrate their own back-end systems to this API.

4 The case: container transhipment via a port

By means of a case, the applicability of blockchain and its advantages for realizing the Physical Internet is demonstrated. The example considers sharing the container status amongst autonomously operating enterprises during transshipment via a port. Both the physical and administrative status is considered. Firstly, the current situation is introduced and secondly its implementation by blockchain technology.

4.1 The current situation

At arrival of a vessel in a port like the port of Rotterdam and on-carriage of discharged containers via a terminal to the hinterland, various enterprises are involved utilizing different modalities for on-carriage. In most cases, a container can only be transported from a port to its destination in case sea transport charges are paid (commercial release), the container is actually discharged (container available), and customs has released the container (customs release). This status information is shared amongst the various enterprises by messages according a customer-service provider relation. Figure 2 shows an example of the value chain for transshipment. A shipping line operating the vessel has a contract with a stevedore for loading and discharging containers on the vessel. A shipping line informs a so-called notify of arrival of its containers in a port of discharge. In this example, a forwarder acts as notify. For

[4] https://golang.org/
[5] https://www.docker.com/
[6] http://www.grpc.io/

this case, the consignee is considered to be the notify and a forwarder acts on behalf of the consignee by subcontracting on carriage to a carrier and arranging commercial – and customs release.

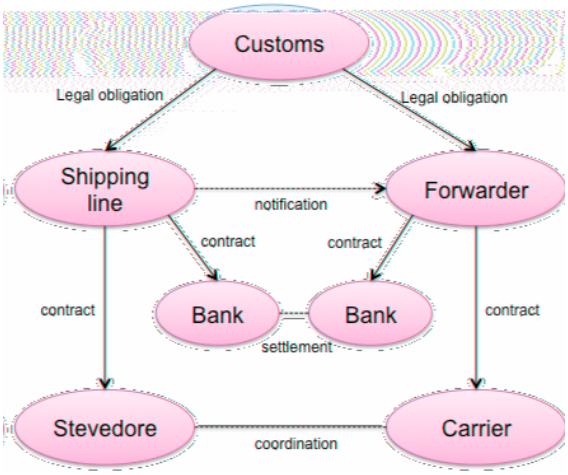

Figure 2: alue chain for container transshipment in a port

Both the stevedore and the carrier have to receive the actual status of a container for its on-carriage. However, they don't have the complete status information: commercial release is generated by a bank and known to the forwarder and the shipping line, customs release is generated by customs and known to the forwarder, and the stevedore generates the discharge status to the shipping line. The shipping line can make commercial release available to the stevedore and the forwarder commercial – and customs release to the carrier. The stevedore still has to receive the customs release and the carrier must know the discharge status to be able to perform on-carriage. The carrier also must be known to the stevedore to pick-up a particular container. Messaging causes delays in physical handling due to errors (the wrong carrier got status information), lack of status information (a stevedore is not informed of customs release), and delays in sharing the status (a stevedore currently submits a discharge list to a shipping line after the vessel has left the port). Delays in the physical processes leading to extra container storage at a terminal are currently caused by delays in information sharing and should be planned based on customer requirements. A (port) community system can address these issues by storing the container status, but it requires trust in the system and clearly specified Identification, Authentication, and Authorization (IAA) mechanisms (Johnson 2010).

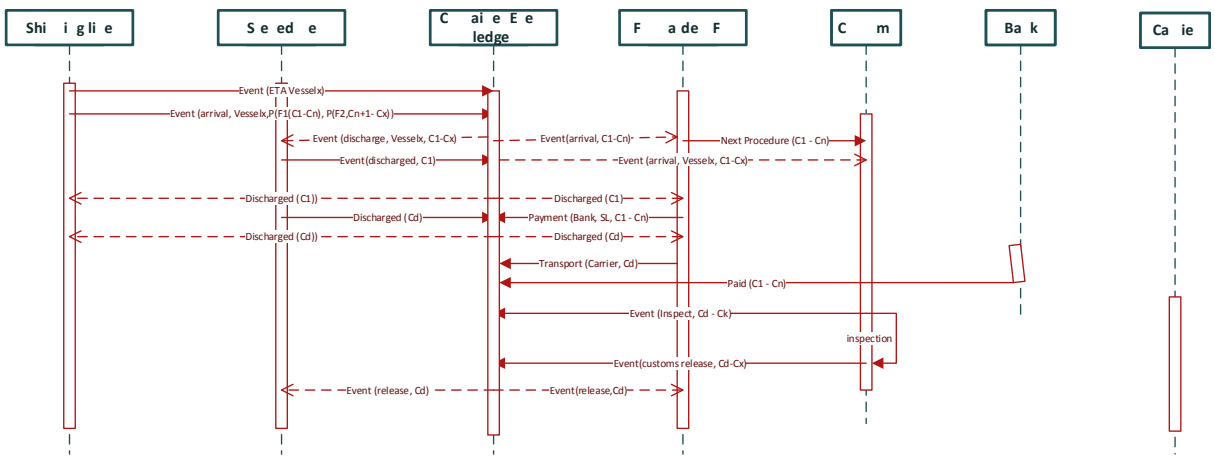

Figure 3: sequencing of operations for a container e ent ledger

4.2 Implementing the case with permissioned blockchain technology

To illustrate the potential added value of the BCT in this domain, we have developed a proof-of-concept application implementing the aforementioned use-case with Hyperledger Fabric. By sharing real time status data and permissions via a trusted blockchain environment, on-carriage processes can be planned different leading to less storage time at a terminal. Figure 3 gives an example of blockchain for sharing events for transhipment of containers. Each arrow depicts an event with a function and permissions P of containers C to roles like forwarder F. The functions of the event reflect milestones in the processes (Hofman 2017).

The first event in the example is the Estimated Time of Arrival of a vessel, followed by an arrival event. The dotted lines indicate that this information is available to customs and the stevedore, but forwarder F1 only has access to containers C1-Cn based on his permission P. A carrier has to request permission (P-req) on behalf of forwarder F1. By adding the transport order of the forwarder to the event ledger, the carrier could automatically receive the release event and the stevedore would be aware of the carrier picking up container Cd. The permissions would be simplified. Smart Contracts or, in this example, event ledger applications provide functionality to the participants, where they behave according rules agreed in a community and permissions control accessibility. Event data structures specify the data that can be retrieved or written to the blockchain. Generation of events by trusted sensors (IoT, Internet of Things) could provide validation.

Each event in the sequence diagram of figure 3 represents an operation on a data structure. For the sake of simplicity, we have chosen to develop one smart contract supporting the use case. The smart contract includes all relevant data structures and operations on these data structures. Each operation results in an API; a role has a set of APIs representing its operations. All stakeholders have to be registered and assigned roles within this smart-contract. When they are registered, they can trigger invoke- and query transactions associated to these roles according the APIs. Invoke-transactions validate the data entered via the API against the data structure of the smart contract, the proper role assigned to a particular stakeholder, e.g. if in this example a shipping line notifies a forwarder, the notifier has to have that role, and the participation of a stakeholder in the blockchain. For instance, if a stakeholder acting as notify is not registered, an error occurs and the transaction is not added to the blockchain. If this stakeholder is registered as carrier and not forwarder, the transaction is also not added.

For the proof-of-concept we developed a NodeJS[7] web application that connects to the blockchain. This application exposes a traditional JSON (Java Script Object Node) API and has methods for enrolling users on the blockchain, deploying the smart-contract to the blockchain and interacting with our specific smart-contract. For demonstration purposes, we have developed a front-end application with the Angular framework[8].

5 Discussion

We have chosen to develop one smart contract for the case. The smart contract shows that status information and data entered by each stakeholder is immediately available to each other stakeholder based on the APIs of its role. Permissions are implemented by invoke- and query transactions of a particular role. One stakeholder grants permissions explicitly to another based on its invoke-transactions. From the perspective of the Physical Internet, blockchain would best fit data sharing and support interoperability between any given stakeholders.

[7] https://nodejs.org/en/
[8] https://angular.io/

Aspects of Hyperledger Fabric that we have not yet explored, seem to support local data sharing, thus distributing data only to members of a community.

The approach taken for development of smart contracts reflects the current implementation of interactions between stakeholders in a port community. The smart contract can be extended to implement all roles and rules of container transhipment via that port, with each interaction by a particular stakeholder modelled as a data structure of that smart contract. A Port Community System like Portbase could develop such a smart contract for the Rotterdam, the Amsterdam and other ports of the Netherlands. The smart contract will manage all contracts and parties in one application with distributed data storage. This smart-contract does not interact with other smart contracts, which simplifies version management. The smart contract can however be very complex and therefore difficult to develop and to test. The smart contract for the use case is already over 3.000 lines of code and still captures only a small part of the functionality. There is no estimate yet of the number of lines of code required to support all procedures and data sharing in a port like the Rotterdam port. Another complexity is the lack of flexibility. When a (small part of a) procedure in the port changes, the entire smart contract will have to be revised, and all data stored in the smart-contract will need to be migrated to the revised version of the smart contract. This data is required by the new smart contract to support operations on relevant data.

Another approach is development of a smart contract representing container and a smart contract for every role interacting with the container smart contract. Whenever a new container enters the port community, its smart contract is instantiated. It results in very flexible smart-contracts for each role. Each role can utilize its concepts and language in its smart contract that is matched to the concepts representing ‘container’ in this example. Transhipment of every new container can be based on the latest version of the smart contract source-code. A downside of such a design is that over time it will be difficult to keep each smart contract compatible with all the versions of contracts that it has to interact with, unless there are uniform rules for interaction between any two stakeholders specified according a choreography (Hofman 2012), (Dietz 2006).

We have only considered a particular community with its rules. Whenever a stakeholder participates in more than one community or has more than one role in a community, it has to implement the smart contract of its role in each community. Consider for instance a forwarder shipping cargo via Rotterdam and Antwerp port via sea and Schiphol via air. Each smart contract provides a set of APIs for a role, where commonality between those APIs is not guaranteed since smart contracts have different developers. Unless agreements can be made on data semantics and choreography for logistics that are implemented in smart contracts, the costs of implementing blockchain to support the Physical Internet will be too high for individual stakeholders and hyperconnection is not feasible. Development of smart contracts for roles based on agreed semantics and choreography also simplifies testing: each smart contract can be validated against (the part of) the choreography it supports, including data shared. Many validation rules of smart contracts can thus also be generated. It also allows various stakeholders to develop these smart contracts, thus rapidly increasing the deployment. Potentially, each community can add its particular smart contract for a role to a generic smart contract of that role, thus supporting localization. It would allow for instance different procedures for container pickup and drop off by a carrier per port.

BigChainDB takes a more fundamental approach to managing data objects representing for instance ‘container’ and ‘vessel’ (McConaghy, et al. 2016). It considers each data objects as ‘asset’ with a particular owner with its particular permissions. Ownership of these assets can be transferred to other stakeholders that will thus have a right to change a particular asset. We have still to investigate the possibilities of this approach combined with smart contracts running on the database.

6 Conclusions

This contribution has taken a functional perspective with respect to the utilization of blockchain technology to support interoperability for the Physical Internet. It illustrated that for container transhipment a smart contract can be developed and deployed, where each stakeholder with a role has immediate access to state changes based on a set of APIs implemented by the smart contract. From a functional point of view, blockchain can provide hyperconnectivity for the Physical Internet. This paper did not discuss non-functional requirements like performance and scalability, complexity of data structures, etc. These are still for further research.

We have discussed two approaches for development of smart contracts, a community and a role based approach. Another approach would also be that of a dominant player providing particular functionality to its suppliers or customers (see also (Choudry 1997), that identifies three approaches for inter-organizational systems). Eventually, they all result in bilateral solutions for each stakeholder involved, thus not addressing the issue of large-scale interoperability required for the Physical Internet. As we have argued, the level of conceptual interoperability (Wang, Tolk and Wang 2009) is required based on agreements of semantics and choreography.

Besides the development of smart contracts, there is also the issue of permission operations on data objects or assets called in BigChainDB. A combination of BigChainDB with smart contracts needs further research into data ownership and permissions.

Considering these requirements, we can argue that blockchain technology is not yet mature to support interoperability for the Physical Internet, where potentially a large number of autonomous objects, individuals and organisations need to share state space data. There is no development, testing, and validation environment for of smart contract development by different stakeholders, which is also required from a software engineering perspective. ‘Smart contract stores’ allowing different developers of smart contracts to offer their solutions, similar to the Apple store or Google Play for apps on smart devices, are also not yet feasible. There are already industry initiatives to apply blockchain technology for secure document exchange, thus providing paperless transport[9]. However, these applications do not necessarily represent the state of a logistics (sub)system, nor are they compatible with other solutions like Blockfreight.

References

Andrychowicz, M. et al. “Secure multiparty computations on Bitcoin.” *2014 IEEE S mposium on Securit and Pri ac* . 2014. 443-458.

Atzori, Luigi, Antonio Iera, and Giacomo Morabito. “The Internet of Things: a suyrvey.” *Computer Net orks* (Elsevier) 54 (2010): 2787-2805.

Badzar, Amina. *Blockchain for securing sustainable transport contracts and suppl chain transparanc - an e plorati e strud of blcokcahin technolog in logistics.* Master Degree Thesis, Service Management and Service Studies, Lund University, Lund University Libraries, 2016.

Bakker, J. “Blockchain technology, an explanatory case study to identify the underlying principles and to determine the corresponding capabilities.” MSc., Leiden University, 2016.

Biggs, Phillippa, Toby Johnson, Youlia Lozanova, and Nancy Sundberg. “Emerging Issues for a Hyperconnected World.” *The global information technolog report*, 2012: 47-56.

[9] http://www-03.ibm.com/press/us/en/pressrelease/51712.wss

Boccardi, Frederico, Robert W. Heath Jr., Angel Lozano, Thomas L. Marzetta, and Petar Popovski. “Five Disruptive Technology Directions for 5G.” *IEEE Communications Maga ine* 52.2 (2014): 74-80.
Buterin, V. et al. *Ethereum White Paper - a ne t-generation smart contract and decentrali ed application platform.* 2014-2017. https://github.com/ethereum/wiki/wiki/White-Paper.
Caffyn, G. *E erledger brings blockchain to fight againt diamond theft.* 2015. www.coindesk.com/everledger-blockchain-tech-fightdiamond-theft/.
Choudry, Vivek. “Strategic Choices in the Development of Interorganizational Information Systems.” *Information S stems Research* (pubsonline.informs.org) 8, no. 1 (1997).
Crainic, Teodor Gabriel, and Benoit Montreuil. “Physical Internet enabled Hyperconnected City Logistics.” *Transport Research Procedia - The 9th International Conference on Cit Logistics.* Elsevier, 2016. 383-398.
de Meijer, Carlo, and Menno de Bruijn. “Cross-border supply-chain finance: an important offering in transaction banking.” *Journal of Pa ments Strateg & S stems* (Henry Stewart Publications) 7, no. 4 (2014): 304-318.
Dietz, J.L.G. *Enterprise Ontolog , Theor and methodolog* . Springer-Verlag, 2006.
Dimitrakopoulos, George, and Panagiotis Demestichas. “Intelligent Transportation Systems.” *IEEE Vehicular Technolog Maga ine* 5, no. 1 (March 2010): 77-84.
Eckartz, Silja, Wout Hofman, and Anne Fleur van Veenstra. “A decision model for data sharing.” *eGo 2014.* Dublin, Ireland: Springer, 2014.
Endsley, Mica R. “Toward a theory of situation awareness in dynamic systems.” *Human Factors: the journal of the human factors and ergonomics societ* 37, no. 1 (1995): 32-64.
Greenspan, G. *Multichain pri ate blockchain - hite paper.* 2015. www.multichain.com/white-paper/.
Gubbi, Jayavardhana, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. “Internet of Things (IoT): a vsiion, architectural elements, and future directions.” *Future Generation Computer S stems* (Elsevier) 29 (2013): 1645-1660.
Heshket, David. “Weakness in the supply chain: who packed the box?” *World Customs Journal* 4, no. 2 (2010).
Hofman, Wout. “Improving Supply Chain Processes by subscription to milestones.” *12Th ITS European Congress.* Strasbourg, France, 2017.
—. “Runtime logistic process orchestration based on business transaction choreography.” *Business Process Management - Process A are Logistic S stems orkshop.* Talinn, 2012.
Hyperledger. *H perledger hitepaper 2.0.* 2016. https://wiki.hyperledger.org/groups/whitepaper/whitepaper-wg.
Johnson, B.C. “Information Security Basics.” *Information Securit Association Journal*, 2010: 8:28-34.
Kieck, Erich. “Coordinated Border Management: unlocking trade opportunities through one stope border posts.” *World Customs Journal* (World Customs Organization) 4, no. 1 (2010).
Malfliet, Jonas. “Incoterms 2010 and the mode of transport: how to choose the right term.” *Management Challenges in the 21st Centur : Transport and Logistics.* 2011. 163-179.
McConaghy, T., R Marques, A. Muller, D De Jonghe, and G. McCullen. *BigChainDB: a scaleable blockchain database.* 2016. https://www.bigchain.com/whitepaper.
McFarlane, Duncan, Vaggelis Giannikas, and Wenrong Lu. “Intelligent Logistics: involving the customer.” *Computers in industr* , 2016: 105-115.
Merrienboer, Siem, Albert W. Veenstra, W.P. van den Haak, and L.A. Tavasszy. “Using floating truck data to optimise port logistics.” *10th Intelligent Transport S stems European Congress (ITS2014).* Helsinki, 2014.
Montreuil, Benoit, Russell D. Meller, and Eric Ballot. “Physical Internet Foundations.” In *Ser ice Orientation in Holonic and Multi Agent Manufacturing Robots*, by Theodor Borangiu, Andre Thomas and Damien Trentesaus, 151-166. Heidelberg: Springer-Verlag, 2013.

Nakamoto, S. *Bitcoin: a peer-to-peer electronic cash s stem.* 2008. https://bitcoin.org/bitcoin.pdf.
Rukanova, Boriana, Niels Bjorn-Andersen, Fred van Ipenburg, Stefan Klein, Godfried Smit, and Yao-Hua Tan. "Introduction." In *Accelerating Global Suppl Chains ith IT-inno ation*. Springer, 2011.
Schonberger, Andreas, Christian Wilms, and Guido Wirtz. *A requirements anal sis of Business-to-Business integration.* Bamberg: Fakultat Wirschaftsinformatik und angewandte Informatik Otto-Friedrich-Universitat, 2009.
Smith, J. *Blockfreight: the blockchain for global freight.* 2016. www.blockfreight.com.
Steiner, J., and J. Baker. *Pro enane Blockchain: the solution for transparanc in product.* 2015. www.provenance.org/whitepaper (accessed September 1, 2016).
Tanenbaum, Andrew S. *Computer Net orks (Third Edition).* Prentice Hall, 1996.
The Digital Transport and Logistics Forum (DTLF). "An outline for a generic concept for an innovative approach to interoperability in supply and logistics chains." Discussion Paper, EC DG Move, Brussels, 2017.
Wang, Wenguang, Andreas Tolk, and Weiping Wang. "The levels of conceptual interoperability model: applying systems engineering principles to M&S." *Spring Simulation Multiconference.* Society for Computer Simulation International, 2009.
Weiser, Mark. "The Computer for the 21st Century." *Scientific American*, no. 3 (1991): 94-104.
Whitmore, Andrew, Anurag Agarwal, and Li Da Xu. "The Internet of Things - survey of topics and trends." *Information S stems Frontiers* (Springer) 17 (2015): 261-274.
Wood, G. *Ethereum: a secure decentralised generalised transaction ledger - Homestead re ision.* 2015. http://gavwood.com/paper.pdf.
Zhang, Yu, and Jiangtao Wen. "The IoT electric business model: using blockhain technology for the Internet of Things." *Peer-to-peer Net orking and Applications* (Springer) 10, no. 4 (2017): 983-994.