Towards a Distributed Ledger System for Supply Chains

Petra Maria Asprion
University of Applied Sciences and Arts
Northwestern Switzerland. FHNW
petra.asprion@fhnw.ch

Philipp Hübner
Teraport, Munic
Germany
philipp.hübner@teraport.de

Pascal Moriggl
University of Applied Sciences and Arts
Northwestern Switzerland. FHNW
pascal.moriggl@fhnw.ch

Abstract

Interoperability and traceability of digital supply chains are becoming a major competitive factor. Businesses operating in supply chains need to share interoperable information and systematically track product and service deliveries. This research investigates a novel approach to model digital supply chains and operationalizes this through a "Distributed Ledger System" in combination with "Smart Contracts". Based on design science, relevance and rigor for a novel approach are derived. As resulting 'artifacts', exemplary supply chains using colored Petri-nets are modeled as a structured and automatable instance for the sketched 'Token-flow Supply Chains'. For the operation of our visionary scenario, a baseline concept with an associated architecture is drafted. We argue that the outlined approach and related artifacts are predestined to achieve a new quality of performance and innovation including bridging the current challenges for digital supply chains.

1 Introduction

A 'Distributed Ledger System' (DLS) consists essentially of an internet-based database of records or a public ledger [1] [2]. In the ledger, all transactions or digital events that have been executed can be stored and shared among participating parties [2] [3]. The exceptional and disruptive features of DLSs are that each transaction is verified by a consensus of the DLSs´ participating parties and that once entered, information can never be erased [1] [2].

The digital currency 'Bitcoin' is one of the most well-known application based on a DLS [2]. Whereas Bitcoin itself is highly controversial discussed, the underlying DLS works almost perfect and inspires researchers and practitioners to develop numerous applications [2].

Crosby et al. [2] predict that DLSs will trigger an economic revolution and are predestined for developing a globally open democratic and scalable digital economy; they see tremendous opportunities for DLS based applications. It is not surprising that the release of Bitcoin in 2009 [4] has triggered a wave of new developments beyond digital currencies using DLSs as the underlying technology [5]. One concept that fits perfectly in DLSs are 'Smart Contracts', which are executable programs containing a set of rules (contracts) under which participants agree to interact with each other [6]. Smart Contracts are the purest form of a decentralized automation and DLSs supports it perfectly with decentralized execution offerings [7].

Information technology (IT) systems provide sufficient support for the sharing of information in Supply Chains (SCs). However, SCs are often struggling with the centralized trust mechanism, caused by the participating parties' self-interests, the information asymmetry between them and not least, the lack of transparency across company borders [8].

The value of IT systems for SCs depends on their input information credibility and its traceable processing. Credible, traceable and timely information sharing among SC participating parties is the most critical enabler for efficient SCs [9]. According to Wu et al. [10], DLSs, or "a set of geographically delineated public ledgers with a suitable hand-off mechanism from one ledger to another" could address those problems.

1.1 Relevance

DLSs in combination with Smart Contracts are excellent to support SC-related demands like exchange tracking or reveal information across various SCs in order to provide real-time visibility. In addition, DLSs facilitate a decentralized verification of information; this can be used to trigger any events or transactions [11].

Triggers in the form of events or transactions may provide the mechanism to hand-off information across different DLSs within a SC. This 'handing-off' of information between SC participating parties is increasingly important and contributes to the information integration and transparency in SCs or SC networks [12]. In order to reduce SCs' complexity, narrow down and grasp SC-related challenges (section 3.3) it seems appropriate to set up a sufficient modeling formalism [13] with functionality to automate modeled (SC-related) elements. Therefore, the application of a DLS with Smart Contract functionality in combination with modeled SCs as a pre-condition could be a promising approach towards a DLS-based architecture for digital SCs.

URI: https://hdl.handle.net/10125/59902
ISBN: 978-0-9981331-2-6

1.2 Objectives

The potential of DLSs in combination with Smart Contracts is already recognized in research and practice, for example from [14] [15] [16]. It is predicted that Smart Contracts will gain a remarkable speed in the SC industry [15].

Our objective is to outline the relevance, opportunities and an exemplary operation model of digital SCs based on DLSs. To this end, we sketch how a modeling formalism can be used to create digital SCs which in turn can be operated on DLS-based 'Token-flow Supply Chains' (in the following abbreviated with 'token-flow SCs').

1.3 Methodology

Our research process started with three research questions: (1) how can we accelerate digital SC integration with DLSs (as outlined in section 1.1), (2) how will a modeling formalism support this integration and (3) how could a first rough prototypical architecture look like?

To answer the questions we used design science as a method [17], which considers both rigor and relevance of the results. According to [17] the development of an artifact, in our case, a prototypical concept is embedded into circles of empirical and theoretical validation and improvements. In the center of the three recommended cycles is the 'design cycle' considering the iterative design of the artifact. The other one, the 'relevance cycle' bridges the contextual environment of the research that needs to be considered with the design activities that are occurring in the 'design cycle'.

The next one, the 'rigor cycle' connects the 'design cycle' with the knowledge base, the scientific foundations. To converge the latter, we used an iterative approach for a systematic literature analysis based on the recommendations of [18] [19]. During the iterative steps of our analysis, we focused on scientific databases, mainly on Web of Science (all journals) and Google Scholar (top journals). We enriched our findings by adopting practical experience from whitepapers or blogs. The main queries we used were 'Distributed Ledger' and 'Blockchain' combined with 'Smart Contracts' plus 'Supply Chain' or 'Logistics' in combination with 'digital', 'management', 'challenges', 'issues', and 'architecture'.

The 'relevance cycle', which bridges research results to the research environment, was driven by our focus on business challenges/issues in context of SCs and derived from literature. Another proof of relevance - the application of the designed artifacts in an appropriate environment that provides empirical validation and further changes needs, will be considered in the next round of the cycle and will be part of our future research. The 'rigor cycle' is the key to accepting the research results in the academic community; it serves research contributions to the knowledge base. In our research, this will be the exemplary modeled SCs as an input for DLS-based token-flow SCs and an associated baseline architecture. Both provide the foundation for further research on the use of DLS to improve/accelerate SCs and releated processes.

The remainder of the paper is structured as follows. We discuss, based on literature, key concepts and characteristics of DLSs (chapter 2). After that, relevant concepts and integration perspectives of SCs are compiled (chapter 3). The need of modeling SCs and Petri-nets as modeling formalism/language is explained in chapter 4. Whereas chapter 5 outlines our 'vision' and provides a baseline concept for token-flow based SCs, which can be integrated in a DLS. Chapter 6 concludes and shows further research intentions.

2 Distributed Ledger Systems

In this chapter, we discuss the basic concepts of DLSs, Smart Contracts, and characteristics that are crucial for potential SC improvements.

2.1 Concepts

In 2008, the foundation for DLSs was laid with an article, published under the pseudonym Nakamoto [4], with a description of an Internet-based distributed ledger, the so-called 'Blockchain', to realize the digital currency 'Bitcoin'. The blockchain is simply a ledger that can be distributed and synchronized across multiple nodes. The latter can be any electronic device that is connected to the internet and as such has an Internet Protocol (IP) address. A node supports any network by maintaining a copy of the Blockchain and, in some cases, of the process transactions [4].

Every transaction that occurred could be stored in this ledger with the advantage of complete transparency for the participating parties [20]. The principal idea of Bitcoin [4] is the absence of a central trust instance, which was mandatory in previous digital currency systems. With regard to the discussed digital SCs, transparency is an important characteristic for the required real-time information whereas the absence of intermediary actors is also useful with regard to efficiency and cost reduction [9] [8].

Bitcoin-like DLSs support a consensus process: the 'proof of work' (PoW) [4] [20]. With PoW, specific network nodes, called 'miners', compete against each other to solve a cryptographic puzzle [21]. The fastest solver is allowed to verify a set of transactions and to create a so-called 'block': a data structure, which contains the verified transactions and, among other information, the hash value of the previous block. A generated block is submitted to the network and will be attached to the existing distributed ledger. The motivation to participate in the system of PoW is monetary: the solver gets a specific reward (in Bitcoin) for creating a new block.

DLSs can be divided into two essential paradigms: ‘permissionless’ or ‘permissioned’. There are still some others like ‘federated’, ‘public’ or ‘private’. The DLSs discussed so far can be characterized as permissionless, which means that they are publicly accessible; the structure is decentralized - without any central instance or intermediates. Two unique properties mark these DLSs: first ‘trustlessness’, which means that the participating parties do not need to trust a central control instance, but can only rely on the technical infrastructure and the ‘self-support’ of the system [22]. The latter is monetarily incentivized for the PoW work.

The opposite paradigm is permissioned DLSs; the participating parties are known and verified by a central instance with intermediary tasks and authorized to monitor the DLS and, if applicable, initiate roll back transactions, which would be impossible in a permissionless DLSs [20]. Permissioned DLS do not require an elaborate consensus process nor an economic incentive for the self-support. This allows simpler consensus protocols that validate transactions and the creation of more performant blocks and have lower energy consumption as PoW-based systems.

A compromise between permissionless and permissioned DLSs is a ‘federated DLS’ that operates under the leadership of certain participants supervised by a consensus process [22]. Compared to permissionless DLSs, a consortium DLS is hardly affected by high-energy consumption or even scalability issues [22].

2.2 Smart Contracts

Advanced DLSs provide functionality to deposit ‘autonomous’ software agents or programming codes, so-called Smart Contracts, that can be automatically enforced without asking a or relying on a trusted authority [6] [23] [24]. This way, a DLS acts as a virtual computing machine, transferring transaction system properties to the execution of software; thus, DLSs provide a perfect infrastructure for Smart Contracts [23].

The term Smart Contract is commonly used with two different orientations [25]: (1) Operational - as programming code typically (but not necessarily) executed on a DLS; the term ‘contract’ indicates that the programming code performs quasi-contractual conditions and control assets. (2) Legal – as subject terms of the contract(s), which can be described in software. A common platform for executing Smart Contracts is ‘Ethereum‘ [14], which includes a suitable (turing-complete) programming language. The latest – as code - can be executed on the DLSs via replication – each node executes the Smart Contract code stored on the ledger, which results as a negative consequence in a bad performance. Furthermore, Smart Contract code execution is guaranteed by mutually distrusting nodes [14] and the initial creator or any participant parties are not enabled to influence the content or the execution of the contract once it has been established [24].

2.3 Characteristics

To understand the difference between traditional databases and a DLS, it is worth discussing essential characteristics of DLSs. We chose three examples that we consider as important; the order reflects their importance from our point of view:

(1) Integrity: the use of cryptographic methods and consensus processes makes a permissionless DLS quasi-immutable. It is nearly impossible to alter the stored information without being detected. In a PoW consensus, attackers would need to control over 50% of the networks computing power [26]. The inherent immutability provides the high data integrity; transactions cannot be counterfeited or manipulated after being accepted by the consensus process and transmitted to the DLS. Although data integrity is a highly desired property, it can be a disadvantage because incorrect Smart Contracts cannot be reversed and must be accepted ‘as is’ by the participating parties. In addition, data protection regulations like the European General Data Protection Regulation (GDPR) could have an impact because of the personal right to be forgotten which needs to be clarified. In addition, it must not be concealed that due to their similarity to centralized systems or traditional databases, permissioned or consortium DLSs do not have a specific advantage regarding integrity since nodes in such systems can readily agree to alter existing information [26].

(2) Decentralization: the risk of a central point of failure (in comparison to traditional databases) is eliminated; if one network node breaks off, other participants will keep the system running. This makes a DLS highly fault-tolerant and resilient [3]. Furthermore, no (third-party) intermediates are necessary; this reduces the related (third-party) risks and offers space for cost savings. Both are relevant features in the context of digital SCs challenges.

(3) Transparency: is a huge benefit of DLSs; they enable transactions to be tracked and evaluated for a wide range of questions. However, from a critical view, for permissionless DLSs, the assurance of personal privacy is still not solved and remains serious issue [20].

Overall, DLSs offer promising characteristics; the highlighted ones above give an impression about the manifold application possibilities.

But, there are also critical characteristics that should be considered [1]: to name some, for example DLSs latency and scalability under-performs significantly compared to traditional databases; this means the execution of Smart Contracts on DLSs could be extremely slow and therefore expensive. Besides, there are potential security threats with respect to permissionless DLSs as well as already mentioned potential areas of conflicts between transparency and personal privacy.

3 Supply Chains

In this chapter, we discuss basic concepts, the architecture of SCs, and scenarios where DLSs could provide promising advantages.

3.1 Concepts

A SC "consists of all stages involved, directly or indirectly, in fulfilling a customer request" [27]. It includes all actors from pre-suppliers up to end customers. Multiple exchanges occur between involved actors, like an exchange of raw materials, products, information, and payments [28].

Fundamental SC structures are either 'convergent', 'divergent', 'conjoined' or 'network' [29]. Convergent stands for assembling sub-products to a final product; divergent means that one supply turns into different sub-products; conjoined is the combination of a convergent and a divergent SC structure; the network structure does not fall into any of the preceding three structures [29].

Information and material in these SC structures can flow in different operational directions. Figure 1 shows a tiered SC according to a conjoint SC structure with distribution. Widespread internet access and the availability of mobile devices both have a substantial impact on information flows between SC actors and their trust in the underlying data [30].

The integration of different information layers (e.g., network-, organization-, physical-, financial-) into one value-creating ecosystem is aspired to and transforms traditional SCs into digital SCs [31]. The information flow of products along SCs is comparable with a discrete event system: the system has discrete states and is event-driven, whereas the asynchronous discrete events drive the evolution of the states over time. The SC environment determines the state of the SC object [32].

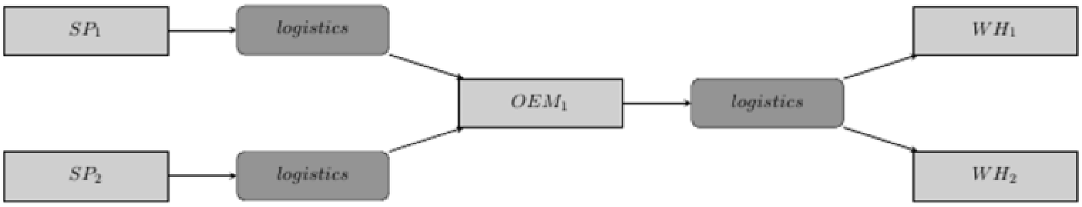


Figure 1. Conjoint Supply Chain Structure

An Information System (IS) for digital SCs should be able to analyze SC states, provide current information to all network participants and facilitate decision-making.

The underlying architecture should be developed to be able to manage for example data sources from sensors, integrate third-party data or accumulate information in near time [33]. The IS should efficiently integrate suppliers, manufacturers, warehouses, and stores to produce goods and distribute them in the desired quantities, to the correct locations in time to minimize system costs while meeting service level requirements [34].

One concept to depict and track physical products in the digital world is 'digital twins'. These are digital replica of physical assets. A digital twin in our experimental scenario is a serialized, distinguishable product, which represents the asset that passes the discrete states when defined events trigger the state evolution [30]. The state of a digital twin is always depending on asynchronous discrete events. Technically, instead of a product, a system asset could be for example a supplier, a third-party logistics company, a truck, an order or a storage.

3.2 Architecture

The architecture of a digital SC consists of hard- and software, which enable systems or participating parties to collect and exchange data. A hardware example could be an on-/offline product sensor or more broadly a 'Cyber-Physical System' (CPS) reporting for example temperature data to the relevant actors during transportation. Software examples are automated decisions or execution software based on temperature sensor logs that trigger decisions, actions or information.

With the growing impenetrability and complexity of SC networks, such information provides the actors involved the confidence they need. Thus, both the data provenance, as well as data integrity can be largely guaranteed [35].

For the application of DLSs in SC networks there are both supporting and critical opinions: Kaijin et al. [34] propose for example a double SC structure to solve identified issues of decentralized agricultural business resources in China; the proposed architecture relies on a single Smart Contract balancing the transactions between suppliers, SC network, Blockchain-network, and demanders. Rodriguez et al. describe a SC network based on an architecture with three layers, namely user, network, and sensors. The architecture utilizes user-generated data, enriched through sensors, and stored in a network (cloud). This architecture facilitates solutions for SC functions like automated system feedbacks, demand predictions, routing optimization, production planning or machine failure predictions. These SCs could be their own participating party as (one) role in a SC network [30]. Wüst and Gervais [1] throw a rather critical look at the DLSs application for SCs: in their model, it is central that if there is a constantly online trusted third party, the user will have to choose the trusted third party over a distributed-ledger system [1]. The authors state that the requirements for a demand-driven SC are more in line with the already known benefits of a DLS, but for supplier-driven SCs, the benefit comparing a DLS to a traditional, centralized database cannot clearly be distinguished. However, the immutable ledger records provide a solution for a reasonable audit trail.

3.3 Challenges

We know that macro trends, such as digitalization and globalization increase the complexity in SCs; others result from striving for efficiency and lower operating costs. Business models like lean manufacturing or just-in-time also place high demands and expectations on digital SCs. Therefore, we will discuss a few selected challenges that may be mitigated or abolished by using our token-flow based SCs based on DLSs.

The homogeneity of data and technology is crucial to integrate data from multiple sources for the effectiveness of digital SCs. There are substantial hurdles to overcome to achieve this because of the incompatibility of ISs and data formats: heterogeneity of data semantics and obsolesce of technology [35].

SC visibility is crucial in order to enable actors to predict demands and optimize supply as well as inventory levels. It is important, to predict/find anomalies, avert them and be able to enhance supplier accountability for example to ensure on-time delivery. Reasons for the weaknesses of invisibility are complex, but often triggered by non-interconnected systems and processes, low data accuracy, quality, or harmonization [36].

A disruption at some point in the SC is not limited to its occurrence, but it causes ‘ripple effects’. Competitive business models require that companies need to deliver products at highly regular times while the demand can quickly increase or decrease. In turn, sudden demand changes trigger ‘bullwhip effects’, which are related to these requirements and similar to ripple effects; a small change in demand can cause increasing swings in inventory and afterwards in the SC network. These swings can cause higher costs or longer delivery times [37].

Overall, the challenges in digital SCs are similar to those that occur in the context of ‘Industry 4.0’, a term given to the automation and data exchange trend in manufacturing [38]. Currently, there is ongoing research focused on the pillars of Industry 4.0 (e.g., CPS, Internet of Things); researchers predict that future products and services will be flexibly connected via the Internet and/or via DLSs to guarantee various requirements such as consistency and connectivity [39]. Our visionary token-flow SCs are drafted to support such future scenarios focused on SCs.

4 Modeling Supply Chains

This chapter introduces SC modeling and explains extended Petri-nets – as an appropriate modeling formalism, which can be used, instantiated and executed by DLSs.

4.1 Concepts

For SCs, two modeling concepts are common: (1) discrete event systems and (2) differential equation systems (DESs) [13]. Models based on the first concept deal with operational level events (e.g., warehouse management, material handling, logistics routing) while DESs are sufficient to abstract SCs to continuous models. The latter are mainly relevant for strategic (long-term) decision-making [40], whereas the first concept is appropriate for simulating detailed relations of SCs.

4.2 Colored Petri-nets

Petri-nets are appropriate to model SCs [13] - they enable the automation-based formalization of discrete event systems and the simulation of concurrent/asynchronous processes [41].

Petri-nets can be described as a directed bipartite graph consisting of two kinds of nodes: place and transition. Arcs connect transitions with places or vice versa. Arcs are weighted with positive integers. Places may contain markings. A transition T fires if an input place of T contains at least w(p,t) markings, where w(p,t) is the weight of the arc from p to t.

The firing removes w(p,t) tokens from each input place p of t and adds w(p,t) to each output place p. Petri-nets can be extended with distinct markings to so-called Colored (CP)-nets. Markings are elements of sets. Arcs and places are associated with these sets; transitions can destroy and produce markings from different sets.

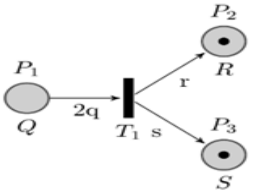


Figure 2. Colored Petri-net (CP-net)

Figure 2 shows a CP-net. The places P1, P2 and P3 extended with the disjunct sets Q, R, and S. Markings of P1 are q∈Q, P2 are r∈ R and P3 are s ∈S. If two Q-markings are at P1, T1 fires and puts one R marking on P2 and one S marking on P3.

The limitation of Petri-nets regarding the distinctness of markings can be bypassed by using CP-nets. The latter can be used for both formalizing the relations in SCs and as a technological basis for a DLS. In this respect, a distinctive marking can be implemented as a transferable token within a DLS. Therefore, CP-nets can be used to create aggregated models of SCs, which in turn can act as input for DLS-based SCs.

A simplified approach to model SCs is to represent them as directed graphs. For example, a graph node stands for a SC stakeholder, while edges depict the flow of goods and information between actors (in the edge direction). Financial flows can be modeled as reversed edges between nodes.

We call this graph model 'SC network graph' because it models the structure of a SC network and the direction of the different flows. This graph with its characteristics could be easily extended to CP-nets, which in turn allow both richer modeling and instantiation on DLSs.

For this, stakeholder-nodes of the SC network graphs need to be replaced by CP-net places, whereas directed edges of the SC network graph will be replaced with two arcs and a transition. The respective in-between product, material or financial transaction are represented by differently colored markings and arc annotations. A change in the marking configuration of a CP-net place can be referred to an SC event: the outgoing or ingoing of goods to respective SC actors.

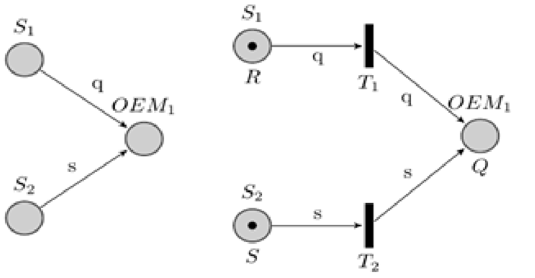


Figure 3. SC network graph and CP-net

Figure 3 shows an exemplary SC network graph with a corresponding CP-net. In the graph on the left side, the suppliers S1 and S2 are connected via the material flows q and s with the stockholder OEM1.

The CP-net on the right side extends this with colored markings of the sets Q and S and the transitions T1 and T2. The approach results in a highly aggregated digital SC model indicating material flows and the ingoing and outgoing of products.

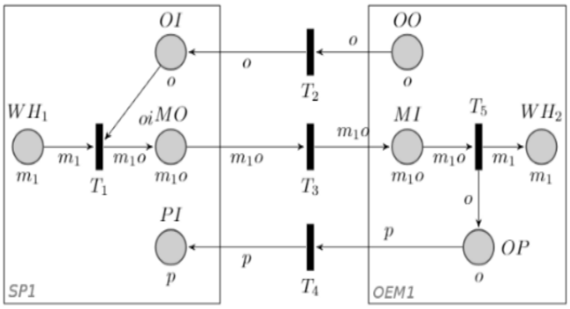


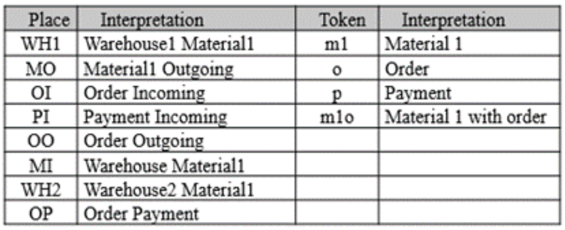
Figure 4. SC order processing

Figure 4 represents a CP-net with an inbound SC between SP1 and OEM1. The model shows an order processing between two actors (OEM1, SP1).

CP-nets are capable to model elements and events of SCs on a fine-grained level. SC processes can be modeled with sub-CP-nets representing more complex material flows (e.g., assembling, diversification, bundling of products). CP-net markings can be used to represent material flows within SCs and financial flows as well. For example, the arrival of goods can be linked with a payment flow in the opposite direction. Furthermore, control events like order(s) incomings/outgoings can be trace- and auditable connected.

Initially, a marking at place OO indicates an order of material m1. An o marking at OI signals an incoming order to SP1. This triggers the delivery of one m1 via transition t1, which combines m1 with the dedicated order to a token m1o. After the delivery (place MI), the transition T3 separates the order from the material and initiates the payment (place OP). The interpretation of places and markings with regard to SCs are depicted in table 1.

Table 1. Places and markings



Place	Interpretation	Token	Interpretation
WH1	Warehouse1 Material1	m1	Material 1
MO	Material1 Outgoing	o	Order
OI	Order Incoming	p	Payment
PI	Payment Incoming	m1o	Material 1 with order
OO	Order Outgoing		
MI	Warehouse Material1		
WH2	Warehouse2 Material1		
OP	Order Payment		

5 Token-flow Supply Chains

This chapter sketches our 'vision' and is an associated baseline concept of how SCs based on CP-nets can be implemented into and monitored by a DLS.

5.1 Vision

We assume that CP-net models are well suited as the foundation for organizing efficient SCs by using DLSs. For this, we sketch a DLS based token-flow based system, which will be established between participating parties (SC actors).

The DLS is the underlying infrastructure for the execution of the CP-net based SCs as the digital correspondence of the physical SCs. The DLS provides the functionality to execute multiple and separated or interconnected CP-net based SCs simultaneously.

To execute CP-net based SC models on a DLS, places, and transitions will be instantiated by dedicated Smart Contracts. This way, CP-net elements (places, transitions) are addressable nodes within the DLS where both can receive and send tokens. These tokens are distinguishable [39] and correspond to colored markings in a CP-net.

A token is either a digital twin token or an 'event trigger'. A digital twin token is a representative for a physically existing good – the DLS equivalent of a digital twin. As an event trigger, we denote any external or internal event, which is not directly associated with a physical object; therefore, it can be destroyed or created in a well-defined procedure via transitions.

Figure 4 shows these two kinds of markings: a marking in *O1* indicates an order event, while a marking in *WH1* indicates a product within a warehouse. Tokens can be enriched with additional information and documents. This way, SC-related documents can be managed implicitly by the ownership of tokens [42].

In addition, the financial flow can be integrated into the token-flow SC. For that, a trustworthy digital currency, maybe Bitcoin (?) could be incorporated. This way, the exchange between material flow and financial transactions can be managed.

A cyber-physical integration can be provided via smart property abilities: digital twin tokens are linked to clearly distinguishable physical devices using cryptographic algorithms. A device controls itself with a certain set of sensors or actors and connected to the network via DLS transactions [43].

The outlined token-flow SCs, will be able to generate various benefits for companies operating in SCs. In the following, we list some of them to show the relevance of our approach.

(1) **Information Distribution**: the main reason for bullwhip and ripple effects is the distortion of information. Information about point-of-sales orders is delayed or will not be sent to the relevant SC shareholders. Token-flow SCs will provide complete SC transparency and therefore prevent bullwhip or ripple effects.

(2) **Continuous Provenance**: all SC-related events stored on the ledger. This makes every event transparent and guarantees product or data provenance by real-time traceability; every position within the SC network can be proven through the analysis of token transformations.

(3) **Integrated Automation**: SC events can trigger processes directly with DLS transactions. For example, a CP-net place associated with any payment of an order can be triggered depending on the arrival event of the corresponding goods. This event automation is completely transparent for all SC actors.

(4) **Integrated Financial Flow**: due to the application of a stable digital currency system, financial flow management can be integrated into the DLS. Payments can be linked directly to material-flow related events (without latencies of traditional banking systems).

(5) **Integrated Document Management**: digital twins can be enhanced with documents (e.g., quality certificates, customs clearance) - an integrated management for SC documents corresponding to the physical processes can be established.

(6) **Risk Management**: the state of the SC is continuously synchronized and distributed by a DLS; this guarantees 'real-time' transparency. Furthermore, every (critical) event can be simulated (e.g., sudden demand changes) directly on the instantiated SC model and therefore risks can be quickly mitigated.

5.2 Baseline Concept

To operate our vision, an architecture needs to be drafted for the DLS. We used the design science method [17] to draft a 'baseline' architecture with five layers. The concept is explicitly created for the CP-net based SCs (section 4.3). Similar concepts dedicated to DLS or Blockchain architectures can be found in [44] and served us as an inspiration.

The five-layers architecture shown in figure 5 requires an underlining DLS, which is geared to execute Smart Contracts (see section 2.2) and includes a distributed database layer.

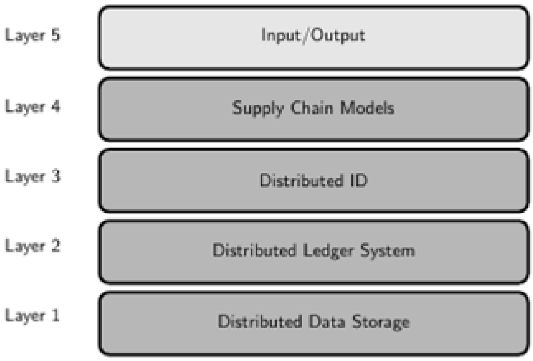


Figure 5. Layers of a token-flow SC

Layer 1, 'Distributed Data Storage' (DDS) refers to technologies/protocols (e.g., Kademila) providing data access without a centralized instance [34]. DDS is mainly used for 'peer-to-peer' file sharing [8] and can be linked with DLS transactions from layer 2. SC actors can create and encode documents for selected receivers and save them into the DDS. Documents are related to digital twin tokens, thus document management is realized implicitly by token ownership.

Layer 2, 'Distributed Ledger System' (DLS) refers to the underlying technical infrastructure for token-flow SCs; it is required that the DLS can execute Smart Contracts efficiently and with a low-latent transaction processing. In addition, the DLS needs to be able to create transactional privacy with cryptographic techniques (e.g., stealth addresses, ring signatures) [45].

Layer 3, 'Distributed ID' refers to a decentralized identification (ID) service; it is an application of decentralized ledger technology providing authentication services for DLSs. For the token-flow SCs, it is used to identify shareholders and thereby implement the management of rights for the dynamic creation of SC model elements [46].

Layer 4, 'SC Models' refers to the CP-net modeled SCs and instantiate them. The following Smart Contracts should be implemented: (1) 'SCNet', (2) 'SCMarkings', (3) 'SCPlaces', (4) 'SCTransitions'.

SCNet (1) is a contract instancing the closed CP-net - each SCNet (1) encapsulates a SCMarking (2) instance, a set of SCPlaces (3) and SCTransitions (4) to realize a SCNet (1). SCNet (1) itself is responsible for managing user access privileges to different parts and dynamic changes of the CP-net by adding or removing places/transitions. SCMarking (2) is a contract needed to define unique identifiable and undividable tokens. It acts similar to [39], but for an enclosed set of network SCPlaces (3) and SCTransitions (4). SCPlaces (3) and SCTransitions (4) stand for single CP-net graph nodes; each node has a unique address for receiving and sending tokens. Places and transitions represent the corresponding CP-net functionality and they are dedicated to managing the token transactions feasibility (accept and decline tokens) and to create respectively destroy tokens.

Layer 5, 'Input/Output' provides access to layer 4 in three ways, (1) Oracle events (2) cyber/physical (CPS) linkage and (3) analysis of the transaction ledger. Oracle events are places in the SC model, which can create marking tokens; the creation of the token is triggered by an external event (e.g., payment confirmation, goods arrival). Each external intervention or control is realized by Oracle events. In [23] the concept of Smart Contacts is extended to so-called 'smart properties' or 'proplets'; the idea behind is to enhance physical objects with embedded property rights; the proplets should be able to monitor smaller devices (e.g., sensors or effectors). Smart properties or proplets can be implemented by using DLS as a virtual part of a CPS.

Deniaud et al. [47] realize smart properties or proplets with hardware-bound cryptographic keys and an appropriate protocol. A similar cyber-physical-linkage approach should be utilized for token-flow SCs. Therefore, digital twin tokens are bound to a physical device; the device owns a DLS address to accept and send transactions. This way, actors can track events or be informed when specific events occur. Also, parts of the SCs can interact autonomously as a container equipped with sensors can interact with a transport ship.

6 Conclusion and Outlook

To answer our first research question – how digital SC integration can be accelerated with DLS – we proposed to model digital SCs with CP-nets, which allows instantiating SC models on a DLS. As a suitable DLS, we have identified those that possess properties, such as Ethereum, which can execute code for Smart Contracts.

For our approach, we selected the promising approach of DLSs as the underlying infrastructure because of their specific characteristics described in section 2.2 and 2.3. By using design science [17] as our guiding steps, we carried out a systematic literature review in order to evaluate the status of digital SCs and, in particular, to identify the challenges that may be solved by using DLSs.

The outlined challenges or deficits of the current DLS (section 2.3, 3.3) gave us some verification of the relevance of our research. Using a step by step approach [48], we modeled exemplary SCs based on CP-nets to sketch 'virtual twin' scenarios (section 4.2); this supports our second research question of how a modeling formalism can be used to transmit and operate SCs or SC elements via DLSs.

In chapter 5, we answered the third research question by developing a rough prototypical concept, the token-flow SC, for instancing CP-net models as input/instance for DLSs (section 5.1).

To complete our idea, we sketched a baseline architecture to operate and execute the token-flow SCs (section 5.2).

According to [17], the proof of a designed artifact needs to be carried out in an appropriate environment (in our case e.g., real-life SCs in a company) to achieve empirical validation. This is a pending next step, which we declare as the next research task.

For further research, many open questions and challenges need to be answered: for example, the DLS underlying technical implementations tend to result in significantly huge nets. With regard to our intention, a potential solution could be an enlarged system to instantiate hierarchies of CP-net based SCs so that the CP-net marking tokens can be forwarded to a hierarchy node, which represents, in turn, a SC sub-system – this must be explored in the near future.

The nearest next fields of research closely related to our presented vision goes towards the DLS for SCs. Our next research will go first in the refinement of the CP-net based SCs with an introduction of Petri-net hierarchies. In addition, we will work on an approach for smoothly rolling back DLS transactions, which is necessary e.g., to deal with errors. Furthermore, we plan to develop a working prototype to evaluate the concept and architecture within real world environments.

7 References

[1] K. Wüst and A. Gervais, "Do you need a blockchain?", in *International Association for Cryptologic Research Archive, IACR*, https://eprint.iacr.org/2017/375.pdf, 2017.

[2] M. Crosby, Nachiappan, P. Pattanayak, S. Verma and V. Kalyanamaran, "BlockChain Technology: Beyond Bitcoin", 2016. [Online]. http://scet.berkeley.edu/wp-content/uploads/AIR-2016-Blockchain.pdf.

[3] Bank for International Settlement, "Distributed ledger technology in payment, clearing and settlement", 2017. [Online]. www.bis.org/cpmi/publ/d157.pdf.

[4] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", 2008. [Online]. http://bitcoin.org/bitcoin.pdf.

[5] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. Kroll and E. Felten, "SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies", in *IEEE Symposium on Security and Privacy*, San Jose, USA, 2015. pp. 104-121.

[6] N. Szabo, "The Idea of Smart Contracts", 1997. [Online]. https://tinyurl.com/Szabo.

[7] BlockchainHub, "Smart Contracts", 2018. [Online]. https://blockchainhub.net/smartcontracts.

[8] S. Chen, R. Shi, Z. Ren, J. Yan, Y. Shi and J. Zhang, "A Blockchain-based Supply Chain Quality Management Framework", in *14th IEEE International Conference on e-Business Engineering, ICEBE 2017*, Shanghai, China, 2017. pp. 172-176.

[9] Ö. Özer and Y. Zheng, "Establishing Trust and Trustworthiness for Supply Chain Information Sharing", in *Ha A., Tang C. (eds) Handbook of Information Exchange in Supply Chain Management. Springer Series in Supply Chain Management 5*. Springer, Cham, pp 287-312.

[10] H. Wu, Z. Li, B. King, Z. Miled, J. Wassick and J. Tazelaar, "A Distributed Ledger for Supply Chain Physical Distribution Visibility", in *37th IEEE International Conference on Distributed Computing Systems ICDCSW*, Atlanta, USA. www.mdpi.com/2078-2489/8/4/137/pdf, 2017.

[11] K. Korpela, J. Hallikas and T. Dahlberg, "Digital Supply Chain Transformation towards Blockchain integration", in *Presentation HICSS 50 DSC*, Hawaii, 2017.

[12] H.L. Lee, S. Whang (2004), "E-Business and Supply Chain Integration", in *The Practice of Supply Chain Management: Where Theory and Application Converge. International Series in Operations Research & Management Science 62*, Springer, Boston, MA.

[13] X. Zhang, L. Qiang and T. Wu, "Petri-net based applications for supply chain management: An overview", in *IEEE International Conference on Industrial Engineering and Engineering Management 49(13), IEEEM2011*, Singapore. 2011. pp. 3939-3961.

[14] V. Buterin, "Ethereum: a next generation smart contract and decentralized application platform", 2013. [Online]. https://github.com/ethereum/wiki/wiki/White-Paper.

[15] Deloitte, "Getting smart about smart contracts", 2016. in *CFO Insights*, [Online]. www2.deloitte.com/content/dam/Deloitte/us/Documents/finance/us-cfo-insights-getting-smart-contracts.pdf.

[16] A. Third and J. Domingue, "Linked Data Indexing of Distributed Ledgers", in *Proceedings of the 26th International Conference on World Wide Web Companion, WWW 2017*, Perth, Australia. https://dl.acm.org/citation.cfm?id=3053895, 2017. pp. 1431-1436.

[17] A. Hevner, S. March, J. Park and S. Ram, "Design Science in Information Systems Research", in *MIS Quarterly 28(1)*, 2004. pp. 75-105.

[18] D. Tranfield, D. Denyer and P. Smart, "Towards a methodology for developing evidence-informed management knowledge by means of systematic review", in *British Journal of Management 14(3)*, 2003. pp. 207-222.

[19] S. Boell and D. Cecez-Kecmanovic, "A hermeneutic approach for conducting literature reviews and literature searches", in *Communications of the Association for Information Systems 34(1)*, 2014, pp. 257-286. pp. 257-286.

[20] F. Tschorsch and B. Scheurmann, "Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies", in *IEEE Communications Surveys & Tutorials 18(3)*, 2016.

[21] J. Brito and A. Castillo, "Bitcoin: A primer for policy makers", 2014. in *Technology Policy 29(4)*, Mercatus Center, George Mason University.

[22] R. Beck, C. Becker, J. Lindman and M. Rossi, "Opportunities and Risks of Blockchain Technologies", Dagstuhl Reports, Saarbrücken, 2017.

[23] N. Szabo, "Formalizing and Securing Relationships on Public Networks", 1997. [Online]. https://tinyurl.com/szabo1997network.

[24] L. Luu, D. Chu, H. Olickel, P. Saxena and A. Hobor, "Making smart contracts smarter", in *International Association for Cryptologic Research Archive, IACR*, http://eprint.iacr.org/2016/633, 2016.

[25] J. Stark, "Making sense of blockchain smart contracts", 2016. [Online]. www.coindesk.com/making-sense-smart-contracts/, 2016.

[26] J. Xu (2016). "Are blockchains immune to all malicious attacks?", in *Financial Innovation 2(25)*, Springer, Berlin, Heidelberg.

[27] T. Kostis and C. Papadopoulos, "A design model and a production-distribution and inventory planning model of a multi-node,multi-product supply chain network", in *Escalafón Planta Técnica, SSMSO'15*, Volos, Greece, 2015. pp. 229-236.

[28] K. Arshinder, A. Kanda and S. Deshmukh (2011) “A Review on Supply Chain Coordination: Coordination Mechanisms, Managing Uncertainty and Research Directions”, in *Supply Chain Coordination under Uncertainty*, Springer, Heidelberg, pp. 39-82.

[29] B. Beamon and V. Chen, “Performance Analysis of Conjoined Supply Chains”, in *International Journal of Production Research 39(14)*, 2001. pp. 3195-3218.

[30] J. Rodriguez, M. Blanco and K. Gonzalez (2018) “Architecture Proposal in the Industry 4.0 context”, in *Rocha A., Guarda* T. (eds) *Proceedings of the International Conference on Information Technology & Systems, ICITS 2018*. pp. 677-687.

[31] G. Smith, W. Baker, K. Watson and J. Pokorski II, “A critical balance: Collaboration and security in the IT-enabled supply chain”, in *International Journal of Production Research 45(11)*, 2007. pp. 2595-2613.

[32] E. Glaessgen and D. Stargel, “The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles”, in *53rd Structures, Structural Dynamics, and Materials Conference*, Hawaii, 2012.

[33] B. Maccarthy, C. Blome, J. Olhager, J. Srai and X. Zhao, “Supply chain evolution – theory, concepts and science”, in *International Journal of Operations and Production Management 36(12)*, 2016. pp. 1696-1718.

[34] L. Kaijun, J. Ya, J. Linbo, C. Han and I. Van Nieuwenhuyse, “Research on agricultural supply chain system with double chain architecture based on blockchain technology”, in *Future Generation Computer Systems 86*, 2018. pp. 641-649.

[35] Q. Xu, H. Feng and R. G. Qiu, “Heterogeneous information integration for supply chain systems”, in *International Conference on Systems, Man and Cybernetics 1, IEEE SMC*, 2005. pp. 97-102.

[36] M. Cooper, D. Lambert and J. Pagh, “Supply Chain Management: More Than a New Name for Logistics”, in *The International Journal of Logistics Management 8(1)*, 1997. pp. 1-14.

[37] R. Angkiriwang, N. Pujawan and B. Santosa, “Managing uncertainty through supply chain flexibility: reactive vs. proactive approaches”, in *Production & Manufacturing Research 2(1)*, 2014. pp. 50-70.

[38] E. Hofmann and M. Rüsch, “Industry 4.0 and the current status as well as future prospects on logistics”, in *Computers in Industry 89*, 2017. pp. 23-34.

[39] O. Esteban, “Distinguishable Assets Registry (Contract for NFTs)”, 2018. [Online]. https://github.com/ethereum/EIPs/issues/821

[40] H. Makatsoris and Y. Chang, “Supply chain modeling using simulation”, in *International Journal of Simulation 2(1)*, 2011. pp. 24-30.

[41] N. Adam, V. Atluri and W. Huang, “Modeling and analysis of workflows using Petri nets”, in *Journal of Intelligent Information System 10(2)*, 1998. pp. 131-158.

[42] V. L. Lemieux, “Trusting records: is Blockchain technology the answer?”, in *Records Management Journal 26(2)*, 2015. pp. 110-139.

[43] M. Sallaba, D. Siegel and S. Becker, “IoT powered by Blockchain How Blockchains facilitate the application of digital twins in IoT”, in *Deloitte Issue*, 2018. [Online]. https://www2.deloitte.com/content/dam/Deloitte/de/Documents/Innovation/IoT-powered-by-Blockchain-Deloitte.pdf.

[44] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat and L. Njila, “ProvChain: A Blockchain-based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability”, in *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, IEEE/ACM CCGRID* , Madrid, Spain, 2017. pp. 468-477.

[45] N. Van Saberhagen, “CryptoNote 2.0”, 2013. [Online]. https://cryptonote.org/whitepaper.pdf.

[46] J. Fang, C. Yan and C. Yan, “Centralized identity authentication research based on management application platform”, in *1st International Conference on Information Science and Engineering, ICISE 2009*, Nanjing, China, 2009. pp. 2292-2295.

[47] S. Deniaud, P. Descamps, V. Hilaire, O. Lamotte and S. Rodriguez, “An Analysis and Prototyping Approach for Cyber-Physical Systems”, in *Procedia Computer Science 56(1)*, 2015. pp. 520-525.

[48] K. Peffers, T. Tuunanen, M. Rothenberger and S. Chatterjee, “A Design Science Research Methodology for Information Systems Research”, in *Journal of Management Information Systems 24(3)*, 2007. pp. 45-77.