



# **THE FIFTH WAVE**

BRIE-ETLA Collection of Articles

Timo Seppälä | Tomasz Mucha | Juri Mattila (eds.)

# Table of contents

**INTRODUCTION TO COLLECTION OF ARTICLES**

Beyond AI, Blockchain Systems, and Digital Platforms: Digitalization  
Unlocks Mass Hyper-Personalization and Mass Servitization ..... 5

Article 1 Estimating Firm Digitalization: A Method for Disaggregating Sector-level  
Digital Intensity to Firm-level.....25

Article 2 AI Diffusion Monitoring among S&P500 Companies: Empirical Results  
and Methodological Advancements.....49

Article 3 What Have We Learned About Machine Learning? A Meta Analysis .....81

Article 4 Blockchain-Based Deployment of Product-Centric Information Systems.....99

Article 5 Digital Protocols as Accounting and Incentivization Mechanisms in  
Anti-Rival Systems: Developing a Shareable Non-Fungible Token (sNFT) .....139

Article 6 The Little Engines That Could: Game Industry Platforms and  
the New Drivers of Digitalization.....161

Article 7 Platform-Dependent Entrepreneurs: Power Asymmetries, Risks, and  
Strategies in the Platform Economy .....173

Article 8 Supranationalism, Sino-American Technology Separation, and  
Semiconductors: First Observations.....211

---

## ARTICLE 4

# Blockchain-Based Deployment of Product-Centric Information Systems

Juri Mattila<sup>a, d</sup>

Timo Seppälä<sup>a, d</sup>

Pellervo Valkama<sup>b</sup>

Taneli Hukkinen<sup>c</sup>

Kary Främling<sup>d</sup>

Jan Holmström<sup>d</sup>

---

### Affiliations:

<sup>a</sup> ETLA Economic Research, Finland

<sup>b</sup> MyTownDoesGood, Finland

<sup>c</sup> BitPanda GmbH, Austria

<sup>d</sup> Aalto University, Finland

### Original publication:

Computers in Industry, Vol. 125(2) (2021), 103342.



## Abstract

Collecting and utilizing product life-cycle data is both difficult and expensive for products that move between different industrial settings at various points of the product life-cycle. Product-centric approaches that present effective solutions in tightly integrated environments have been problematic to deploy across multiple industries and over longer timespans. Addressing deployment costs, incentives, and governance, this paper explores a blockchain-based approach for the deployment of product-centric information systems. Through explorative design science and systematic combining, the deployment of a permissionless blockchain system for collecting product life-cycle data is conceptualized, demonstrated, and evaluated by experts. The purpose of the blockchain-based solution is to manage product data interactions, to maintain an accurate single state of product information, and to provide an economic incentive structure for the provision and the deployment of the solution. The evaluation by knowledgeable researchers and practitioners identifies the aspects limiting blockchain-based deployment of solutions in the current industrial landscape. Combining theory and practice, the paper lays the foundation for a blockchain-based approach to product information management, placing design priority on inter-industrial and self-sustained deployment.

## Keywords

Product-centric information management, Blockchain, Inter-industrial deployment, Platform sustainability

# 1 Introduction

Products in use—especially durable and capital goods—are valuable sources of information in many industrial settings (Aitken, Childerhouse, & Towill, 2003; Anderson & Zeithaml, 1984; Kärkkäinen, Holmström, Främling, & Artto, 2003; Rink & Swan, 1979). However, in settings where products move between systems and industrial settings at different points in the lifecycle, product data is rarely effectively collected and used (Lehtonen, Ala-Risku, & Holmström, 2012). Moreover, a combination of information asymmetries and a lack of incentives may even result in supply chain actors destroying data valuable to one another (Ala-risku, 2009).

The concept of *product-centric information management* (Kärkkäinen, Ala-Risku, & Främling, 2003; Meyer, Främling, & Holmström, 2009; Tang & Qian, 2008) was developed to enable multiple actors to share information on product individuals comprehensively over their lifecycle. While significant improvements have been observed in case studies (Bussmann & Sieverding, 2001; Främling, Holmström, Loukkola, Nyman, & Kaustell, 2013; Hribernik, Rabe, Schumacher, & Thoben, 2006; Lyly-Yrjänäinen, Holmström, Johansson, & Suomala, 2016; Rönkkö, Kärkkäinen, & Holmström, 2007), the deployment of product-centric information management as a sustained solution has been challenging. Deployment challenges include, e.g. high initial costs, scalability (Leitão, 2009; Tähtinen, 2018; Trentesaux, 2009), and unresolved conflicts of interest regarding platform control and governance (K. Främling, Harrison, Brusey, & Petrow, 2007). Establishing more integrated platform solutions for product data management has been similarly challenging (Naphade, Banavar, Harrison, Paraszczak, & Morris, 2011).

This conceptual paper explores blockchain-based deployment of a product-centric information system. The focus is on the use of blockchain-based functionality (Buterin, 2013; Hukkinen, Mattila, Smolander, Seppälä, & Goodden, 2019; Nakamoto, 2008; Poon & Buterin, 2017; Wood, 2013), such as protocols, crypto-mining payments, and smart contracts to initiate and sustain product data collection and use. The purpose is to conceptualize and demonstrate a solution, where the design priority is on the incentivization of actors to participate in providing item-level product lifecycle information, and reimbursing their efforts by using blockchain technology. This paper contributes to research on viable inter-industrial deployment (Alam & El Saddik, 2017; Naphade *et al.*, 2011) and self-sustained platforms (Blossey, Eisenhardt, & Hahn, 2019; De Filippi & Loveluck, 2016; Mattila & Seppälä, 2018).

## 2 Literature review

Storing and maintaining data on each product individual over its entire life cycle is not a trivial undertaking. The high initial investment has been identified as a reason for why integrated product data management systems have not been widely adopted by the industry (Leitão, 2009; Trentesaux, 2009). As an alternative, more loosely coupled peer-to-peer solutions have been proposed to share the burden (Främling, Kubler, & Buda, 2014; Kärkkäinen, Holmström, *et al.*, 2003; Kubler, Främling, & Derigent, 2015). However, while the use of a peer-to-peer approach reduces the investment cost of individual actors, it introduces a variety of new challenges for product centric information management, *e.g.* tracking and coordinating the global state of the system, attracting a critical mass of users, as well as facilitating authentication and trust in a decentralized manner (Petkovic & Jonker, 2007; Trentesaux, 2009).

### 2.1 Product-centric information and blockchain

In the field of product lifecycle management, earlier efforts towards using a peer-to-peer network have mainly been aimed at increasing the interoperability and openness of product data systems (Kubler *et al.*, 2017; Raggert, 2015). However, obtaining guarantees of the satisfactory performance of peer-to-peer networks has been found difficult; Due to the coordination constraints involved, evaluating the global state of a fully decentralized system—and thus predicting its behaviour—can be highly challenging (Trentesaux, 2009). Over the last decade or so, blockchain technology has provided a potential solution to this issue by enabling a single programmatic state to be maintained in peer-to-peer networks in an entirely decentralized fashion (Buterin, 2013; Hukkinen *et al.*, 2019; Poon & Buterin, 2017; Wood, 2013).

Consequently, in recent research literature, several conceptualizations have been drafted for using blockchain-related systems to improve the transparency and traceability (Azzi, Chamoun, & Sokhn, 2019; Caro, Ali, Vecchio, & Giaffreda, 2018; Cole, Stevenson, & Aitken, 2018; ElMessiry & Elmessiry, 2018; Galvez & Mejuto, 2018; Heber, 2017; Heber & Groll, 2018; H. M. Kim & Laskowski, 2018; Kshetri, 2018; Lu & Xu, 2017; Tian, 2016; Westerkamp, Victor, & Axel, 2018; Wu, Li, King, Miled, & Tazelaar, 2017), the sustainability (Bai & Sarkis, 2020; Kouhizadeh & Sarkis, 2018; Nayak & Dhaigude, 2019; Saberi, Kouhizadeh, Sarkis, & Shen, 2019), the cybersecurity and resilience (Banerjee, Lee, & Choo, 2018; Kshetri, 2017; Min, 2019; Papakostas, Newell, & Hargaden, 2019), and the integration and interoperability (Dai, Zheng, & Zhang, 2019; Gordon & Catalini, 2018; Huang, Wang, Yan, & Fang, 2020; Korpela, Hallikas, & Dahlberg, 2017; Miller, 2018; Repository, 2016; Ruta, Scioscia, Ieva, Capurso, & Sciascio, 2017) of supply chain and product data

management structures. Some conceptualizations have also been presented specifically for distributed workflow management with blockchain-based smart contracts (Bahga & Madiseti, 2016; Chen *et al.*, 2017; Evermann & Kim, 2019; Leiding, Memarmoshrefi, & Hogrefe, 2016; Leng, Jiang, Liu, Chen, & Liu, 2017; Yu *et al.*, 2018). Furthermore, other closely resembling themes have been touched upon in many adjacent research streams, *e.g.* focusing on the use of blockchain systems for data governance (Liang *et al.*, 2017; Turk & Klinc, 2017) and ownership management (Karafiloski, 2017; Toyoda, Mathiopoulos, Sasase, & Ohtsuki, 2017; Zhang & Wen, 2017).

Despite the vibrant streams of publications on the issue in recent years, little attention has been paid to the challenge of combining solution deployment and sustainability at the inter-industry level. For example, (Elmessiry, Elmessiry, & Elmessiry, n.d.; Lu *et al.*, 2019; Sternberg, Hofmann, & Roeck, 2020) address the problem of successfully deploying a blockchain architecture for increased transparency and trust in inter-organizational supply chains but do not consider inter-industrial, or system-of-systems, integration. Conversely, (Jiang, Fang, & Wang, 2019; Özyılmaz & Yurdakul, 2019; Tijan, Aksentijev, & Ivani, 2019) discuss using a blockchain-based architecture for creating an inter-industrial backend for the Internet of Things, but do not address the feasibility of solution deployment. (Katuwal, Pandey, Hennessey, & Lamichhane, 2018), on the other hand, briefly acknowledges the potential suitability of using a blockchain system as an incentivization mechanism to deploy a global health information exchange but does not address the solution sustainability aspect. Respectively, (Rajala, Hakanen, Mattila, Seppälä, & Westerlund, 2018) points out the need for self-reinforcing business models for sustainable systems-of-systems, but does not discuss the feasibility of solution deployment.

While potentially sharing a common manufacturing supply chain, product items do not usually follow one uniform chain of ownership throughout their individual lifecycles. Therefore, an inter-industrial perspective combining both effective deployment and self-sustainability is required in order to establish a prominent product-centric information solution, enabling transformational insight into individual product behaviour across national and industrial boundaries.

## 2.2 Blockchain systems and smart contracts

Blockchain technology is often described as a combination of information technology elements and methods enabling the creation of decentralized, distributed, and replicated digital ledgers. To this end, the technology employs *e.g.* peer-to-peer networking, public-key cryptography, digital tokens, multi-version concurrency control, and a cryptographically concatenated chain of data blocks used to store database modifications (Nakamoto, 2008).

For this paper, we define blockchain systems strictly as 1) open source and open access technology compositions; 2) comprising a non-hierarchical peer-to-peer networks without single points of failure or control; 3) which maintain consensus over cryptographically concatenated, shared and replicated append-only data structures; 4) according to deterministic self-contained consensus algorithms, void of external inputs such as validation by central authorities or off-chain signaling (Slootweg, 2016). In other words, we make a clear distinction between blockchain systems and the more loosely defined concept of distributed ledgers. A strict delineation of this kind is necessary, as the latter do not exhibit the same kinds of properties essential to solution deployment, as will be discussed later in this paper in Section 4.3.2.

In a computational sense, blockchain systems can be characterized as distributed state machines: peer-to-peer networks capable of maintaining a single programmatic state—or consensus—across the entire network and its shared data, without any single participant having authority over another. By employing Turing-complete programming languages, state-changing programs known as smart contracts can be created, stored and executed in the blockchain network to facilitate diverse distributed workflows (Buterin, 2013; “Ethereum Frontier Guide,” n.d.; Hukkinen *et al.*, 2019; Poon & Buterin, 2017; Wood, 2013).

Smart contracts can be described as programmatic containers for tokenized assets. Essentially, they are persistent computer programs which have the ability to autonomously govern assets and to execute transactions. Once assets are deposited into a smart contract’s address, they cannot be recuperated until the programming logic of the smart contract permits it. The logic of the smart contract itself is protected by the distributed blockchain network: any unauthorized attempt to tamper with its design is obvious, and easily discarded by other participants (Buterin, 2013; “Ethereum Frontier Guide,” n.d.; Hukkinen *et al.*, 2019; Poon & Buterin, 2017; Wood, 2013).

By default, the execution environment of blockchain-based smart contracts lifeless. In order to interact with the smart contract’s workflow in a state-changing manner, one must compensate the network on a per-operational basis for providing service. These compensations are also used to allocate request priority and to deter aberrant behaviour, such as requesting infinite computational loops. As each network interaction is bundled with its respective payment in this manner, any state-changing activities, such as database writes, are commonly referred to as ‘transactions’ in the blockchain vernacular (“Ethereum Frontier Guide,” n.d.).

For this paper, we define smart contracts as digital computer programs that: 1) are written in computer code and formulated using programming languages; 2) are stored, executed and enforced by a distributed and replicated blockchain network; 3) can receive, store, and transfer digital assets of value; and 4) can execute with varying outcomes according to their specified internal logic (Lauslahti, Mattila, Hukkinen, & Seppälä, 2018).



## 2.3 Problem summary

Deploying product-centric information management systems over the product life-cycle is cumbersome, regardless of the technical approach, as all parties involved in the product-life-cycle also need to participate in the information management solution. Attaining a critical mass for a digital platform often requires considerable initial investments. To deploy a solution, the participation of at least one market side must be first subsidized to attract other market sides onto the platform *via* indirect network effects (Armstrong, 2006; Caillaud & Jullien, 2003; Hagiu, 2014; Hagiu & Wright, 2015; Katz & Shapiro, 1994). Consequently, in order to compensate the high-risk venture of establishing a solution in the first place, the pricing models often involve significant economic rent, reducing the appeal of participation (Gawer, 2009; Hagiu, 2014; Tähtinen, 2018).

Thus, understandably, the question of control and ownership of a product-centric information system has been at the centre of attention in research and development (K. Främling *et al.*, 2007). Recently, however, the problem of control and ownership has increasingly become reframed as a broader question of viable inter-industry deployment, especially in the research domain of cyber-physical systems (Alam & El Saddik, 2017; Naphade *et al.*, 2011; Porter & Heppelmann, 2014).

In addition to the problems related to deployment, another set of problems arises from the complexity of dynamic multi-industrial environments. The problem with static workflow designs is that in today's economy, supply chain structures are often complex and prone to reconfigurations (Ali-Yrkkö, Mattila, & Seppälä, 2017; Rajala *et al.*, 2018). While at the industry level, the data integrations and the required reconfigurations may be manageable, at the inter-industrial level the complexity in this regard increases exponentially. Therefore, even if all the parties involved were fully motivated to co-operate to their best ability, product data regarding individual product items could still become fragmented due to the information asymmetries involved.

The third problematic dimension is related to the motivation to preserve the product data workflow. So far, neither centralized nor peer-to-peer-based solutions have been able to provide a satisfactory solution to the problem of adequately incentivizing solution sustainability beyond individual commercial interests. While centralized models have suffered from asymmetrical power structures and single-points of failure, peer-to-peer models so far have lacked proper governance models to foster sufficient network effects for the solution to perpetuate (Ahluwalia, 2016).

### 3 Methodology

The proposal for an improved design presented in this paper was developed and evaluated by using an explorative design science research approach. Design science is a research method well suited for situations where a practical problem and its solution can effectively be examined through the development of a design artefact, such as a computer program, a system model, or a conceptual practice (Holmström, Ketokivi, & Hameri, 2009; Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). The design science approach was selected because it enables a rigorous way of designing, building, and evaluating a conceptualization for a product-centric information management system.

The study also incorporates elements of the methodology of systematic combining where an emergent theoretical framework, the empirical fieldwork, practical demonstration, and outcome evaluation are developed in a simultaneous, iterative process (Dubois & Gadde, 2002, 2014). While systematic combining is particularly useful for proposing new approaches and ideas for conceptual research, the main focus of this study is in new practice design. It assumes an integrational approach, providing a cross-disciplinary evaluation of the applicability of blockchain technology to address the challenges of introducing product-centric information management in an inter-industrial setting.

A former case study is also exploited and modified to demonstrate some of the key aspects of the conceptualized design proposal (Eisenhardt, 1989). The demonstration was iteratively developed and contextualized to a relevant product item example and industry setting. The programming of this design artifact draws from the methodologies of computer science (Ayash, 2014).

Through an evaluation procedure, design science enables research objectives to be addressed and problematic areas to be charted and pinpointed at an early phase, without waiting for large-scale implementation. To evaluate the validity of the de-

Table 1. A description of the evaluation interviews

SUBJECT	1 <sup>st</sup> ROUND DURATION	2 <sup>nd</sup> ROUND DURATION	AGE	OCCUPATIONAL TITLE	AFFILIATION	EXPERIENCE IN PRODUCT DATA SYSTEMS (YEARS)	EXPERIENCE IN BLOCKCHAIN TECHNOLOGY (YEARS)
#1	51 min	45 min	39	chief technology officer	industry	11	4
#2	68 min	75 min	54	industrial internet facilitator	academic	25	4
#3	61 min	71 min	34	university lecturer	academic	8	0
#4	61 min	45 min	42	entrepreneur	business	20	4
#5	60 min	61 min	55	program manager	industry	25	0
#6	51 min	58 min	24	doctoral candidate	academic	5	2
#7	56 min	51 min	45	head of digitalization	regulator	0	4

sign proposal, and to provide further in-depth insights into the conceptualization, two rounds of seven qualitative interviews were conducted in a semi-structured manner. The interviews were not intended as a substitute for field testing of the design proposal, but for evaluating the key assumptions and concepts, as well as mapping the critical issues related to the implementability of the design. In other words, the aim was to involve the interviewees in exploring what aspects of the problem situation are important from the interviewee perspective, and how these concerns relate to their view and evaluation of the design proposal. A description of how the evaluation sessions were carried out is presented in Appendix A.

The interviewees were selected in an opportunistic fashion, based on their credentials and expertise, and their heuristically evaluated ability to provide the most valuable insights on the design proposal. The first round of evaluation interviews involved a generic system-level demonstration which was not contextualized to any particular product item or industrial setting. The follow-up interview round involved a more detailed and contextualized iteration of the design proposal with a specified product item, a conceptual data model of the product system architecture (not to be confused with a product data model), and an improved source code artefact with more elaborate incentivization and payment mechanisms. The follow-up interviews also involved a Delphi segment (Dalkey & Helmer, 1963) which allowed the interviewees to comment on the summarized key points from the first round of interviews and to readjust their views. The interview questions around which the interviews were framed is included in Appendix B.

## 4 Solution proposal and demonstration

### 4.1 Objectives for a solution

On the basis of the problem summary in Section 2.3, we determine that the main objective for a solution is a design for a product-centric information management system which can be deployed across many industries in terms of costs, coordination, and critical mass, and which can sustain its own existence independently. We postulate that in order to achieve such a design, the system should be able to satisfy the following conditions and specifications: Firstly, the design proposal should be able to a) *enable participation* of all the willing parties. In order to achieve this, the system should feature ahierarchical governance. Secondly, the proposal should be able to b) *prevent data and workflow fragmentation* in a dynamic environment. For this purpose, the system should be based on replicated and distributed architecture. Thirdly, the design proposal should be able to c) *ensure data and platform sustainability* over the complete lifespan of product individuals. For this reason, the system should involve an inherent incentivization mechanism.

## 4.2 Design principles

We address the research problem and our objectives for an improved design with an approach based on blockchain technology. The motivation for choosing this approach stems from the observation that permissionless open source blockchain systems exhibit a range of properties which conveniently line up with our objectives for a solution. Firstly, due to their ahierarchical governance structure, blockchain systems can be well-suited for enabling participation. Secondly, their blockchain data structure and consensus mechanism can be very effective in maintaining multi-version concurrency control in a decentralized fashion. And lastly, crypto-token-based incentivization mechanisms can be directly incorporated in their participation protocol. Furthermore, the chosen approach comes with a proven track record of several peer-to-peer networks already having been successfully deployed in the described manner in the past (e.g. Bitcoin, Ethereum).

In order to accomplish our objectives for a solution, the demonstration of the design proposal needs to show that blockchain systems can be used to involve new parties in the product data system. The demonstration also needs to demonstrate that blockchain systems can be used to include new information as a part of the product-centric information management system. Furthermore, the capability for facilitating adequate incentive structures also needs to be demonstrated.

In this paper, we demonstrate these abilities by employing a smart contract to facilitate a product individual's lifecycle journey. The smart contract was designed for Ethereum, as it represents a suitable deployment environment successfully established in a similar manner as conceptualized in this paper. The other option would have been to establish an entirely new blockchain network as a designated deployment environment for product-centric information management. While perhaps better suited for the actual purpose of the use case, this approach would be difficult to demonstrate in a similar capacity and therefore was not pursued in this paper.

In transitioning from product class data to product-centric information management on individual product items, the number of required transactions can be expected to increase many-fold. Furthermore, as individual product items journey through their individual product lifecycles and paths of ownership, the number of information sources and different data system interactions can also be expected to increase heavily. In order to ensure that the data regarding all the product individuals is provided by all the relevant parties, data provision should be directly rewarded at the level of the participation protocol. For seamless inter-industrial functionality, the system should be constructed so that data exchange can happen spontaneously. In other words, no premeditated *ad hoc* data system integrations should be required between the participants, other than with the blockchain network itself. To this end, the demonstration also illustrates how these incentivization mechanisms can be facilitated by a blockchain-based system design. Furthermore, we also conceptualize,

how the provision and the development of the product-centric information system itself can be incentivized by a blockchain-based approach.

### 4.3 Demonstration of blockchain-based deployment: A loader crane for commercial vehicles

The demonstration of deployment concerns an illustrative product individual, a loader crane for commercial vehicles. These types of loader cranes are manufactured by companies such as Palfinger of Austria, and Hiab of Sweden. The loader crane is typically mounted on a new vehicle before delivery to the customer by the dealer. However, it may also be installed on a vehicle at a later time by the OEM of the loader crane. When the vehicle reaches the end of its life-cycle, the loader crane can be remounted to a different vehicle. This way, the life-cycle of the crane exceeds that of the vehicles to which it is mounted. Over its life-cycle, the loader has many different owners. Furthermore, not only can it be mounted to different vehicles, it can also be repurposed and refurbished by other organizations than the OEM. Product individual data on the loader crane needs to be collected in many countries due to safety regulations.

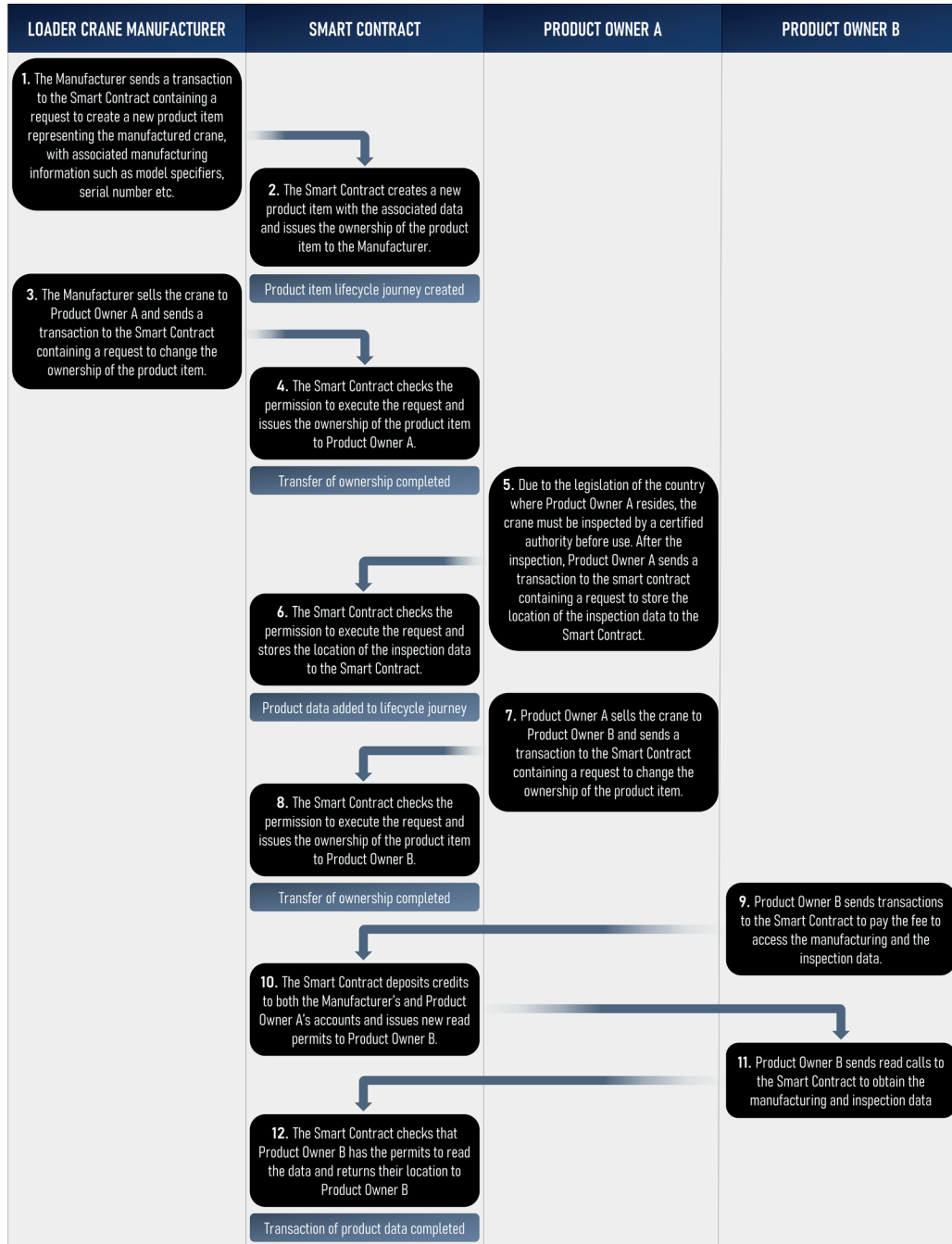
#### 4.3.1 Participation protocol overview

To demonstrate the conceptualization drafted according to our specified design principles, we present an example protocol of a manufacturer deploying product-centric information management over the product life-cycle of a loader crane (see Figure 1). We demonstrate how the relevant contractual and incentive functionalities in each step are defined in the source code that forms the smart contract in Appendix A. The complete and functional source code for the demonstration can also be found at (Valkama, 2020).

The participation protocol of the demonstration begins with the reception of a new loader crane order by the manufacturer. At this stage, we assume that the smart contract facilitating the workflow for the product life-cycle journey is already deployed in the environment consisting of *e.g.* vehicle manufacturers, loader crane OEMs, truck dealers, trucking firms, and service and maintenance companies. In this conceptualized implementation, after the crane has been manufactured, the manufacturer sends a transaction to the smart contract, requesting that a new product item life cycle journey representing the physical crane is established in the blockchain and its ownership assigned to the manufacturer. In addition, the request contains manufacturing information such as crane model specifiers and a serial number to be stored on the product item (1).



**Figure 1** Participation protocol for blockchain-based deployment of product-centric information management over the life-cycle of a loader crane



After this step has been executed by the smart contract (2), the manufacturer can now control the product item in the product data system. As the current owner of the product item, it is possible for the manufacturer to store additional data to the lifecycle journey or query the data already stored without any extra fee.

Upon the sale of the crane to a vehicle manufacturer the crane manufacturer initiates a new transaction in the smart contract in order to transfer the ownership of the product item to the new owner (3). Consequently, the smart contract checks for the permission to perform the request and updates the lifecycle journey accordingly (4).

Over the life-cycle of the loader crane, a multitude of information relevant to different parties is accumulated and can be linked to the smart contract. In the example scenario, once the vehicle manufacturer receives the crane from the loader crane manufacturer, the crane is required to pass an individual inspection performed by a certified authority before it can be installed and used on a vehicle. After the inspection, the vehicle manufacturer sends a transaction to the smart contract in order to store the location pointing to the inspection data (5). Upon receiving the request, the smart contract ensures that the sender of the request is the current owner of the product item and then stores the datum to the smart contract (6).

Once the crane has been mounted onto a vehicle, the vehicle manufacturer delivers the assembly to a truck dealer to fulfil a pre-existing purchase order on the vehicle. Upon the delivery, the vehicle manufacturer sends a transaction to the smart contract in order to transfer the ownership of the product item to the truck dealer (7). The smart contract once again checks for the required permissions and then executes the transfer of the ownership (8).

Before putting the vehicle out for sale, the truck dealer must complete the vehicle registration process and provide documents to the registration authority which prove the vehicle's suitability for its intended use. In order to do this, the truck dealer requires all the relevant information regarding the vehicle's life-cycle journey. To obtain this information, the dealer first sends transactions to the smart contract to pay for the access to the manufacturing and the inspection data from the smart contract (9). Upon receiving the payment transactions, the smart contract deposits credits to the accounts of both the loader crane manufacturer and the vehicle manufacturer for the data they have contributed earlier. Subsequently, the smart contract grants the truck dealer access to the data (10). After the payment transactions have been successfully completed, the truck dealer sends queries to the smart contract to read the relevant data (11). Finally, the smart contract checks that the truck dealer has the valid access and returns the requested data (12). The truck dealer can now proceed with the registration of the vehicle.

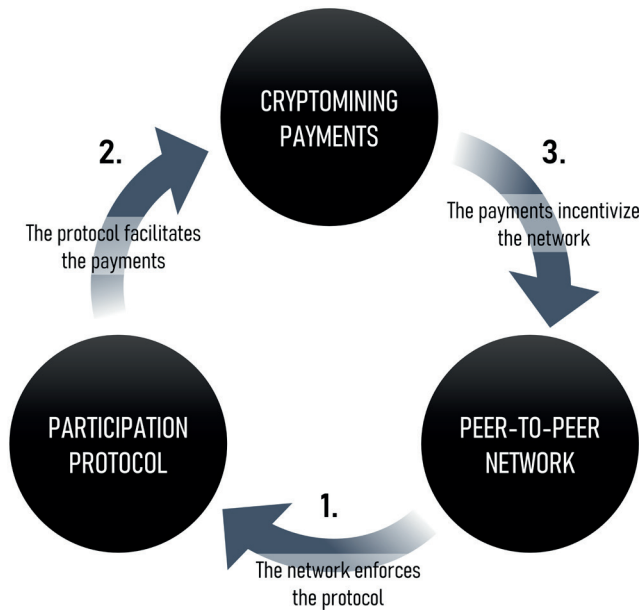
### 4.3.2 Incentivizing the provision of the product-centric information system

The successful deployment of an inter-industrial product-centric information system, such as the one outlined for the loader cranes, is intricately linked to the concept of *network effects*. In economics, a direct network effect occurs when the value to an agent from using a product, a service or a system depends on the extent of its use by other similar agents. Indirect network effects, in turn, occur when such an increase affects the users of a different product, service or system (Armstrong, 2006; Caillaud & Jullien, 2003; Katz & Shapiro, 1994).

Blockchain-based solutions incorporate a mechanism for a positive feedback loop of indirect network effects to incentivize solution deployment. In essence, the blockchain-based operations described in Appendix A begin by drafting a participation protocol—an elaborate set of rules of engagement to which the participants must adhere in order to be acknowledged by the peer-to-peer network. The actor who initially seeks to create the solution for loader cranes starts the deployment by formulating and publishing the participation protocol. Blockchain systems make use of this participation protocol by inherently embedding financial incentive structures for platform collaboration directly into the protocol itself.

The protocol is open, both allowing new actors to join, as well as the introduction of other types of products than loader cranes. Figure 2 illustrates the positive

**Figure 2** The growth-fostering positive feedback loop of network effects in blockchain systems



feedback loop of network effects in blockchain-based deployment. The blockchain system involves a set of rules to which all participants must adhere in order to be acknowledged as members of the network. By contributing computational work, as instructed by the rules of the system, the network enforces a single state of the participation protocol (1). The participation protocol handles each product individual's lifecycle journey and the interactions with it, including the payment transactions for providing product data (2). As each payment also includes a compensation to the network operators for providing service, this incentivization attracts more participants to provide data and to operate the network (3). As the network grows larger, contributing even more computational work (1), the participation protocol grows more robust, making the data and the respective payments in the system more valuable (2). This, again, strengthens the incentives to participate (3), and so on (Athey, Parashkevov, Sarukkai, & Xia, 2016; Athey & Roberts, 2001; Catalini & Gans, 2016; Mattila & Seppälä, 2018).

#### 4.3.3 Incentivizing the provision of product data

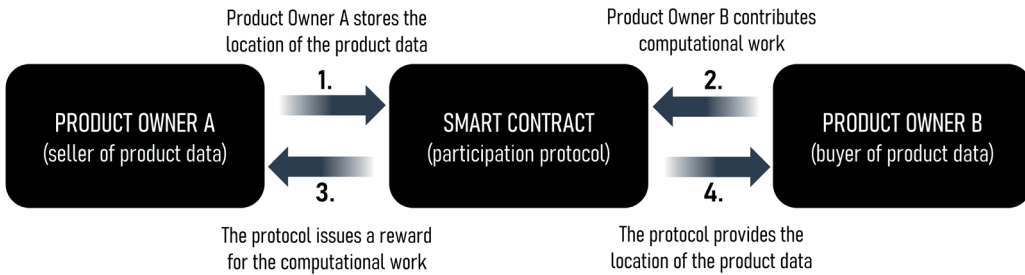
A product datum regarding an individual loader crane can be of very low value to the transacting participants in itself. Therefore, it can be difficult to facilitate the corresponding payments globally in a dynamic environment by any traditional means. Furthermore, in order to maintain the decentralized quality which makes the solution appealing to all parties, the payment processing should also be executed in the same decentralized manner.

While blockchain systems can be used for direct payment processing, they do not scale well in terms of transaction throughput capacity. Therefore, directly facilitating payment transactions through smart contract workflows can quickly become infeasible in large numbers (Hukkinen *et al.*, 2019). Blockchain systems do, however, enable an alternative microtransaction mechanism through the use of crypto-mining payments.

Crypto-mining payments are based on the fact that blockchain systems, require constant inputs of computational work to maintain their single state. Normally, providing this work entitles its contributors to rewards in the form of cryptographic tokens of value in order to incentivize participation. The rewarding is carried out *via* an inflationary tax on the entire network by issuing a small number of new tokens to the recipient of the reward, thus adding tokens into the token supply of the network and depreciating the value of each individual token in the process (Mattila & Seppälä, 2018).

In crypto-mining payments, the cost of the computational work contributed to the network and its respective reward are disentangled from one another to facilitate a payment transaction (see Figure 3). Once the seller has provided the item of

**Figure 3** The mechanism of a crypto-mining transaction, as conceptualized in the participation protocol



sale to the smart contract (1), the buyer contributes computational work to maintain the network's concurrency control, expending electricity which effectively constitutes the payment (2). The smart contract then allocates the respective mining reward issued by the network to the seller (3). Finally, the item of sale is delivered to the buyer (4). In essence, in crypto-mining payments, the act of making a payment always simultaneously contributes to the provision of the payment processing platform itself (Pearson, 2018; R  th, Zimmermann, Wolsing, & Hohlfeld, 2018).

## 5 Evaluation

### 5.1 Technical design

The interviewees unanimously considered the loader crane a good product example and an appropriate industrial setting for the conceptualized design proposal. Two of the interviewees commented (#2,3), however, that while the conceptualization seems well-suited for the loader crane—*i.e.* a product of mid-range complexity—in reality product-centric information management must be extended to far simpler products and sub-components than the crane; In such cases, tracking the material and component identities and incentivizing collaboration could become more challenging *via* the conceptualized design, according to the two interviewees. Mostly the interviewees agreed (#1,4,5,6,7), however, that in a full implementation, the participation protocol could be expanded to facilitate the real-world complexity of a product individual's life-cycle.

The final iteration of the participation protocol was considered a sound design and logically coherent by all of the interviewees. One of the interviewees felt (#4), however, that a better possible way of configuring the participation protocol would have been to assign the loader crane product individual with its own unique identity in an equivalent manner to the manufacturer and the owners, and to use the smart



contract's workflow only as a transaction link layer for the identities, the data, and the associated payments: *"This, I think, would have been more in line with the current Industry 4.0 digital twin mentality. The added benefit here would be that this participation protocol could guarantee the identities of the agents and product individuals when interacting through this kind of a link layer."*

As a noteworthy point for further development, one of the interviewees also remarked (#7) on the design proposal's low threshold for extensive field testing: *"One good thing about this conceptualization is that it wouldn't be a huge effort to try this in practice. It's a classic example of a problem that is so complex that it's difficult to anticipate what would happen, so the easiest way to find out would be to simply try it out. And since the concept itself mainly deals with metadata, the risks for the participants would also be quite low."*

## 5.2 Enabling participation

In Section 4.1, we postulated that in order to achieve our design objectives, the design proposal should feature ahierarchical governance to enable full participation by all the willing parties. To reflect this design principle, the solution proposal was based on a peer-to-peer blockchain architecture with no centralized authority or any designated individual or group responsible for the solution provision.

The distributed design approach was considered a good and sensible starting point for enabling open participation by all interviewees. Interviewees mostly agreed (#1,4,5,6,7) that successfully establishing an inter-industrial infrastructure at scale will require some new type of an approach. While a caveat offered (#1,6) that starting in the right place does not necessarily mean arriving at a functional solution, the proposed design was generally seen (#1,4,5,6) as a step in the right direction in the design principles. As described by interviewee #4: *"If we think about the loader crane industry, this kind of a systemic approach and the entire platform-building way of thinking is still quite alien to them. However, I think this is the only way to enable vast collaboration between different agents around a single product individual's lifecycle. I don't think any other approach would work at such a high level of scope."*

The interviewees also largely agreed (#1,4,5,6,7) that the conceptualized open source, open access, and blockchain-based deployment would significantly reduce the costs of solution deployment and lower the barriers of entry into the product data market. The interviewees mostly agreed (#1,4,5,6,7) that the open access design and the role flexibility in solution provision should make participation more inviting, as its less constrictive nature means that participants are free to pursue business opportunities without restrictions by the solution provider. For inter-industrial deployment, this prospect was also considered pivotal (#1,4,6,7) because of the excessive difficulty of any solution provider anticipating all the use cases and business mod-

els in which potential participants are interested in an inter-industrial setting. However, arguments were also made (#1,4,6,7) that certain functions could still end up requiring centralized services to be offered on top of the system, involving additional fees for the users; For example, the identities of the users and the product items could turn out difficult to onboard in a completely decentralized fashion.

While the open access to become a provider for the solution architecture was also considered (#2,3,4,6) beneficial for the trustworthiness of the system, one interviewee had (#7) reservations in this regard: *“With this kind of deployment, the network could end up being operated by parties not really involved in the supply chain structures at all. Of course, then you are faced with administrative questions, such as can these parties be trusted and is it really sensible that just literally anyone can start operating the data network. Or do we, after all, want to retain a little bit more control in the hands of those who actually use the data and the system?”*

Some concerns were also raised regarding the scalability of the conceptualized design. These concerns were mainly related to three key points. The first point of concern mentioned (#1,2,5,6) by the interviewees was the possibility of runaway costs due to system inefficiencies as the system is scaled up. This consideration stemmed from the technical properties of the conceptualized solution architecture (e.g. the requirement of constant inputs of computational work).

Another point of concern brought up (#1,5,6) regarding scalability had to do with the practical difficulty which often arises in the finer details of scaling up proofs-of-concept and other conceptual solutions. Building conventional IT solutions is a safer practice with a lot more history and experience on avoiding the potential pitfalls. A novel permissionless blockchain-based approach at scale is likely to produce a variety of unforeseeable problems and security issues, such as uncharted attack vectors, which need not have been considered in more traditional approaches.

Lastly, the third scalability-related point of concern mentioned by one interviewee (#2) was the presence of “walled gardens”—the purposeful lack of interoperability maintained by some industry actors as their competitive strategy. Some interviewees felt (#4,6), however, that this kind of a mindset was becoming less common and would be phased out by the market within the next 5–10 years; While customers have not been willing to pay extra for smart product features, market competition is making the smart product approach increasingly a necessity in maintaining a competitive product.

### 5.3 Preventing data and workflow fragmentation

As our second design objective we stipulated that the system should be based on replicated and distributed architecture in order to prevent data and workflow fragmentation in a dynamic network.

Contemporary solutions to product information management have often involved building case-specific *ad hoc* integrations between the data systems of the vendor and the client. Many of the interviewees expressed (#3,4,5,6) the opinion that due to the difficulty of indexing such *ad hoc* solutions in current configurations, the conceptualized design proposal could help locate the source of product data with greater ease. As explained by interviewee #6: *“When a new system comes along, an integration is built to each pre-existing system. And so the number of APIs absolutely skyrockets, and the system doesn’t scale. And at the end of it all, the PLM people are left wondering where the master data is coming from, which systems are integrated with what, and so on. This conceptualization could provide a standard way of transferring the product data between all the various systems.”*

The conceptualized design proposal was purposefully left agnostic in terms of the product data format and meta data standards. The interviewees largely considered (#1,2,3,4,6) this a valid decision, pointing out that specifying a universal standard suitable for the needs of all actors in a cross-industrial context would be exceedingly difficult.

Defining machine-readable formats and relevant meta data standards was, however, considered (#1,2,4,5,6) one of the most important aspects for any shared inter-industrial or even intra-industrial use to be possible. For example, as pointed out by one of the interviewees (#1): *“You want the information fields to have enough flexibility to be able to cover anything, like a potential repurposing of the product, but at the same time, you need enough rigidity to pick up the elements that are important for the loader crane. You need to have the different loader crane manufacturers input similar data in comparable form. That structure is really important.”*

Some of the interviewees elaborated (#1,3,4,5,7) that determining such data ontologies was a task best left for the markets and the soft law efforts of each specific industry. As expressed by interviewee #3: *“At the end of the day, everything hinges on what kinds of product data models are demanded by the customers. This way, companies could be forced to switch over to using different kinds of models.”*

In the demonstration’s participation protocol, the product data is not stored in the blockchain, as such an approach would hardly be technically feasible. This aspect aroused both positive and negative considerations. The most obvious concern was the fact that the product data still needs to be stored somewhere. While the conceptualization does not describe in detail how the product data could be stored, the interviewees were (#1,4,6,7) open to the exploration of InterPlanetary File System-style solutions. InterPlanetary File System (IPFS) is an open-access peer-to-peer network designed to store data by using content-based addressing. In other words, a given address always points to the same content, thereby preventing data fragmentation within the network<sup>1</sup>.

As a positive side, not storing the product data into the blockchain database was seen (#2,4) to enable further access control by each data provider at their end as

they see fit. One noteworthy possibility enabled by this aspect, as pointed out (#4) by one of the interviewees, would be the facilitation of product-centric data products. Differing from data-driven applications, such as software solutions using API-based data for analytics, data products are independent, self-adapting entities which combine data inputs with analytical tools and models to produce new outputs of broadly applicable refined data (J. Kim & Bengfort, 2016). Currently, the API-driven solutions utilized in contemporary approaches are insufficient to construct and manage data products effectively. The conceptualized design proposal could offer a way to record and track the product and user identities, ownership relations, and the relevant data ontologies in a more constructive manner.

## 5.4 Ensuring data and solution sustainability

As the third objective in our design approach, we stated that the system should include an incentivization structure in order to ensure data and solution sustainability over the complete lifespan of product individuals.

One potential problem in this aspect which was pointed out (#1,3,7) is that designing universal incentive structures can be overwhelmingly difficult. For example, if actors were directly compensated for performing transactions of data into product items' life cycle journey, this could lead to the said actors purposefully bloating the system. Similarly, if a generic part of lesser quality is used in maintenance, adding this information to the product data could reduce the resale value of the product. Therefore, the owner may not be inclined to do so, regardless of the incentives embedded in the participation protocol.

While many of the interviewees felt (#2,3,7) that the problems stemming from humans cutting corners cannot be mitigated by incentives embedded in the participation protocol, the resulting market mechanism could alleviate the problem, as explained (#1) by one interviewee: *“If there are 100 fields which should be inputted for the loader crane, is there an incentive to update the fields that are the most popular and have the most valuable use cases? When the system has the incentive mechanism you have conceptualized, I think it will happen organically. When you leave it to a market mechanism, the market will find out which data is more valuable.”*

Another point raised (#2,3,4,6,7) by many of the interviewees regarding the participation protocol was that the system cannot necessarily be perpetuated with internal token incentives alone. Some external motivation for preserving the product data is required outside of the system itself. The interviewees estimated (#1,4,5,6) that the stakeholders in the loader crane's lifecycle would be willing to pay in the order of magnitude of tens to hundreds of euros for relevant data on their product items to be made available upon request, depending on the specific circumstances. This was seen to be motivated by e.g. opportunities of increased sales and modernization,

regulatory compliance, and reverse logistics at the end of the product lifecycle. Heuristically, the amounts were considered (#1,4,5,6) sufficient to enable the sustained facilitation of the curated workflow, as proposed by the design.

The crypto-mining payments conceptualized in the design proposal provoked a mixed reception. On the one hand, the idea was widely considered intriguing. The notion that every payment transaction also simultaneously contributes to the provision of the underlying payment processing architecture was largely seen (#1,2,4,5,6) as an interesting prospect for fostering positive network effects and producing a positive scaling effect for the deployment of the network. Also, the implications for machine-to-machine payments and the idea that smart devices equipped with some CPU capacity and an internet connection could autonomously pay other devices directly for the curation of their own product data throughout their lifecycles mostly aroused (#2,3,4,5,6) interest.

On the other hand, a majority of the interviewees was concerned (#1,4,6,7) that implementing such a payment model would create an extra layer of unnecessary complexity and token price stability issues, potentially requiring some kind of a middleman to mitigate. Also, in regard to the prospect of M2M payments, it was pointed out (#1,2,3,6) that currently, the vast majority of industrial internet devices in use do not have the required smart capacity to carry out such payments. In the words of interviewee #6: *“Usually the software in products like loader cranes is quite specialized and proprietary, so I imagine adding the capability for crypto-mining payments would be quite a painful endeavour in a larger scale.”*

Due to these considerations, mostly the interviewees largely agreed (#2,3,4,6,7) that while an interesting prospect in its own right, crypto-mining payments would not be feasible as the only possible payment option in the present configuration of industrial systems.

## 6 Discussion

Several limitations apply which should be acknowledged when interpreting this exploratory study and its findings. Firstly, this study did not explore the integration of the demonstrated design proposal with other IT systems. Secondly, the study did not consider the details of viable product data formats in product-centric information management or the heterogeneity of real-world product data in general. Thirdly, the study did not address the question of how the actor and product identities could be onboarded in a fully decentralized fashion.

The applied semi-structured interview approach is limited in comparison to the more extensive field testing needed for empirical findings and design iterations in accordance with the design science process. The purpose of the loader crane demonstration and its evaluation was not to capture the complexity of a real product lifecycle.



cle, however, but to illustrate how a blockchain-based deployment of a product-centric solution could be configured to facilitate the necessary core functionalities for handling the product data, the agent identities, and the incentivization mechanisms required for a full scale implementation. Aiming at a solution that can be deployed across different environments over a long period of time, we seek to contribute to the research on viable inter-industrial deployment (Alam & El Saddik, 2017; Naphade *et al.*, 2011) and self-sustained platforms (Blossey *et al.*, 2019; De Filippi & Loveluck, 2016; Mattila & Seppälä, 2018).

While the use of a blockchain-based system offers a different set of abilities than more conventional approaches, some general problematic aspects regarding its utilization remain which were also not addressed in this paper. For example, while the participation protocol can algorithmically manage the solution provision and the product data workflow, the governance of more strategic development goals remains an open question in the research of blockchain systems (Mattila & Seppälä, 2018). Also, some criticism has also been presented regarding the alleged decentralized nature of blockchain systems in the first place (Walch, 2019).

The proposed approach enables anyone to freely enter the system in any market role and to produce open innovations for all areas and functions of the system. This approach, we anticipate, would create power dynamics where all participants are—not necessarily *de facto* equally powerful—but at least algorithmically equipotent and equally privileged by default. In such a system configuration, no participant would have an obligation to participate in the development, provision, or financing of the system architecture and its auxiliary services, but respectively, no participatory role or function would be off-limits to any participant willing to engage in its provision.

The proposed design presented in this paper extends product data management beyond standard systems. In our proposed design, many such systems are linked in a controlled way, with the product individual as the focal and organizing entity. Even when different actors use their own solutions for product life cycle management information, this information is purposefully collected and distributed between these many systems and actors. Our proposed solution makes it possible to incentivize the collection and distribution of high-quality and high-value product lifecycle information for many different types of product data residing in different systems. This is achieved through a mechanism for different entities to initiate and reward this controlled linking. For example, for a composite product with different modules, the product design and manufacturing information is located in the different PLM systems of the OEMs (*e.g.* Windchill, Teamcenter). The asset and performance data is located in the current and previous owners' operational systems (*e.g.* IBM Maximo, Avantis EAM), and service delivery in the systems of different service providers maintaining and supporting the systems (*e.g.* SAP, Odoo). With the proposed solution, an OEM or a product owner can incentivize other parties to collect and share data on product individuals.

The results of this study suggest that while significant challenges for implementation exist in the current industrial landscape, the applicability of blockchain technology to the problem of product-centric information management has so far been perceived narrowly in academia, largely overlooking its potential significance to sustained inter-industrial deployment. This observation supports the earlier findings of (Blossey *et al.*, 2019) where the authors state that the “[supply chain] applications of blockchain technology mostly focus on efficiency improvements and risk mitigation from a single-firm perspective. – However, this perspective largely omits the institutional innovation potential of blockchains reorganizing supply chains for collaborative ecosystem-based value creation.”

The insights provided by this study regarding the incentivized deployment of blockchain solutions for product-centric information management may also help the deployment of similar distributed data sharing solutions intended for other purposes and other sectors of society. The conceptualization delineated in this paper may be especially helpful in cases where the aim is to establish auxiliary services and solutions for business processes that are not core to any of the participants involved. Furthermore, the conceptualized design could also enable an approach where data products on product individuals were manufactured to order, and the curated workflow of the participation protocol served as an index on where the data product could be requested. If successful in its deployment, due to its agnostic data ontology, the system could also be expanded to house a variety of all kinds of data products. Also, the technique could be utilized to manage data in other contexts than product data management, *e.g.* direct from design manufacturing.

## 7 Conclusion

Our study offers a new network-effect-driven perspective on how inter-industrial data sharing solutions could be established and maintained through a blockchain-based approach, including system development, deployment, and payment processing. In most contemporary design proposals for product-centric information management, the deployment and workflow structures of digital interactions are unilaterally controlled by the service provider who is also providing the underlying technical architecture. By disentangling the solution provision from the control of the data and the workflow, hindrances in the integrational development of inter-industrial digitalization could potentially be alleviated, thus enabling more widespread adoption. Further studies are encouraged for the inter-industrial perspective to product-centric information management, with a design focus on sustained solution deployment.

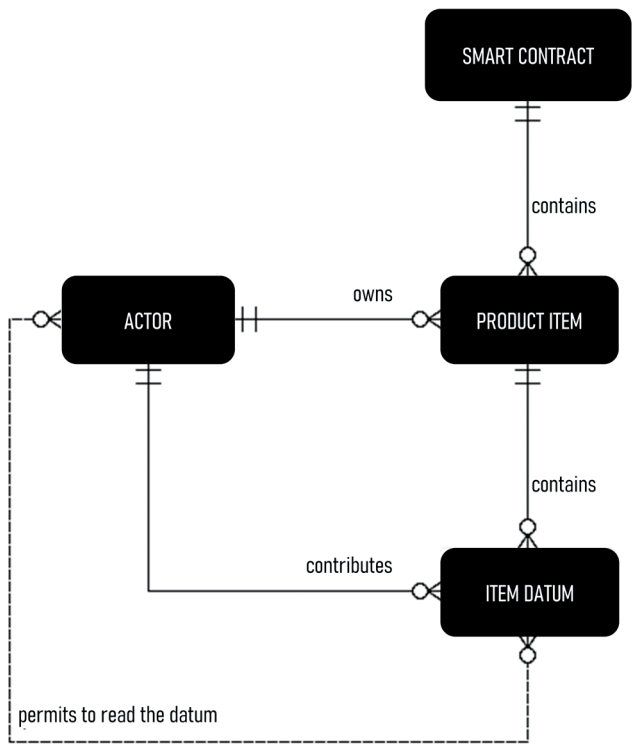
# Appendix A: Protocol for blockchain-based deployment

In the following sections, we will present data model, and the different operations that allow the deployment of the loader crane according to the scenario described above. The complete and functional source code for the demonstration can also be found at (Valkama, 2020).

## A.1 Product system design

The conceptual data model of the conceptualized system is illustrated in Figure A. The product system contains a collection of product items which are owned by actors such as manufacturers or dealers. The product items each contain a collection of item datums. Consequently, each datum added to a system has an originating actor who is thus considered as the contributor of the datum. Only the contributor of a datum can read the particular datum without cost while all other actors in the sys-

**Figure A** The conceptual data model of the product system modelled as an Entity-Relationship (ER) diagram



tem are subject to a fee to be able to access it. The actors who have paid the fee are represented in the figure as having the permit to read a datum.

The implementation of the conceptual data model in Solidity, the language used to describe smart contracts in the Ethereum blockchain platform, is shown below:

#### Product system model (Solidity code)

```
struct ItemDatum {
    address payable contributor;
    mapping(address => uint) permits;
    string datum;
}

struct ProductItem {
    uint itemId;
    address owner;
    mapping(string => ItemDatum) data;
}

uint itemCount;
mapping(uint => ProductItem) private items;
```

The actors in the system are represented simply as Ethereum addresses in the smart contract. This establishes a unique identity to each actor and allows for authentication and access control of the smart contract operations in the Ethereum platform. Furthermore, a simple associative array style data structure of string keys and (datum) values was chosen to represent the product item data. As per the objectives, this imposes minimal restrictions on how to structure and model the product item data, thus enabling different industries to develop their own standards. The requirement of using only textual formats for data also allows for better interoperability across systems and actors. Furthermore, the requirement also discourages polluting the product system with *e.g.* proprietary binary files that are of no use on a larger scale when considering the entire life cycle of a product item and the larger systemic perspective.

The next sections will cover the different operations that are required to implement the semantics of the smart contract, as described in the example scenario. In addition, JavaScript example code of how the smart contract could be called from the client side will be shown.

## A.2 Creating a product item life cycle journey

Just as every loader crane in the physical realm goes through a journey of events over its life cycle, respectively, the life cycle of each corresponding product item object in the smart contract can be structured in the same manner. All the product items begin their life cycle journey in the smart contract when a manufacturer sends a trans-

action to the smart contract, requesting the creation of a new product item with the supplied manufacturing data:

#### Client side (JavaScript code)

```
createProductItem("4950", {serialNumber: 4950, modelSpecifier: "KPV"});
```

Upon receiving the request sent by the client, the smart contract stores a new product item to the blockchain with the manufacturing data and the sender of the transaction (the manufacturer) as its initial owner. Additionally, the smart contract sends an event, that can be subscribed to by clients, signalling the creation of a new product item:

#### Smart contract (Solidity code)

```
function createProductItem(
    string memory correlationId
    string memory _manufacturingData
) public returns (uint itemId) {
    uint newItemId = ++itemCount;
    ProductItem memory newProduct = ProductItem({
        itemId: newItemId,
        owner: msg.sender
    });
    items[newItemId] = newProduct;

    setItemDatum(newItemId, "manufacturingData", _manufacturingData);

    emit ProductItemCreated(newItemId, correlationId, msg.sender);

    return newItemId;
}
```

### A.3 Transferring the ownership of a product item

When the ownership of a physical loader crane is transferred, the product item in the smart contract must also undergo a transfer of ownership so that the new owner can control the product item. The ownership transfer process is initiated by the current owner by sending a transferral request transaction from the client side to the smart contract, with the product item identifier and the Ethereum address of the new owner as parameters:

#### Client side (JavaScript code)

```
transferOwnership(4950, "0x485B48DB7e8c65E76178a4C080a7099A5780aA86");
```

Before executing the transfer of the ownership, the smart contract checks that the sender address of the transaction is the same as the address of the owner of the



product item. If the sender is not the same as the owner, an error is returned, and the transaction is aborted. After ensuring that the sender is the owner of the product item, the new owner is assigned to the product item and the transaction completes successfully:

#### Smart contract (Solidity code)

```
modifier onlyOwner(uint productId) {
    require(
        msg.sender == products[productId].owner,
        "Operation permitted only by owner"
    );
    _;
}

function transferOwnership(
    uint _itemId,
    address _newOwner
) public onlyOwner(_itemId) {
    items[_itemId].owner = _newOwner;
}
```

## A.4 Assigning new data to a product item

As a loader crane journeys through its individual life cycle, it goes through a unique sequence of transformative events. Respectively, the information contained in the product item must be updated to reflect these changes accordingly. To associate new data to the product item, the owner sends a transaction to the smart contract, using the product item identifier, the key identifying a particular datum, and the datum itself as parameters:

#### Client side (JavaScript code)

```
setItemDatum(4950, "latestInspection", {date: "2020-04-21", result: "ipfs://..."});
```

Upon receiving the request, the smart contract first checks that the sender address of the transaction is the same as the current owner and then updates the product item, associating the datum by its key. Additionally, the address of the sender is stored along the new datum so that the smart contract will later be able to identify the actor who has contributed the particular datum to the system:

#### Smart contract (Solidity code)

```
function setItemDatum(
    uint _itemId,
    string memory _key,
    string memory _datum
) public onlyOwner(_itemId) {
    items[_itemId].data[_key] = ItemDatum(msg.sender, _datum);
}
```

## A.5 Paying to access product item data

If an actor wants to access a particular datum but is not its contributor, the actor must first pay a fee to obtain a right to access the datum. To this end, a transaction is sent from the client side with the product item identifier, the datum key and the payment amount as parameters:

### Client side (JavaScript code)

```
payDatumFee(4950, "latestInspection", {value: "10000000000000000"});
```

Upon receiving the payment request, the smart contract first checks that the sender of the transaction is not the contributor of the datum. If the contributor and the sender are the same, the transaction is aborted. Otherwise, the smart contract will deposit the paid fee to the Ethereum address of the contributor and then issue access to the sender while also associating the timestamp of the current blockchain block with the permit:

### Smart contract (Solidity code)

```
modifier onlyNotContributor(uint _itemId, string memory _key) {
    require(
        items[_itemId].data[_key].contributor != msg.sender,
        "Only applicable to actors who are not contributors of the datum"
    );
}

function payDatumFee(
    uint _itemId,
    string memory _key
) public payable onlyNotContributor(_itemId, _key) {
    require(msg.value == 10000000000000000, "Costs 0.001 eth");
    items[_itemId].data[_key].contributor.transfer(msg.value);
    items[_itemId].data[_key].permits[msg.sender] = block.timestamp;
}
```

## A.6 Querying product item data

The product item data may be queried at various stages of the product item's life cycle by various different owners. Furthermore, queries can also be made by others actors with access to the smart contract deployment, such as public authorities or third-party integration systems. However, only the original contributor of a particular datum may access it without a cost, whereas other actors must pay a query fee to obtain access. To query data from a product item, a read query is sent from the client side with the product item identifier and the datum identifier as parameters:

**Client side (JavaScript code)**

```
getItemDatum(4950, "latestInspection");
```

Upon receiving the query request, the smart contract first checks whether the sender of the transaction is different than the contributor of the datum requested. If the sender and the contributor are the same, the requested datum is returned immediately to the sender. Instead, if the sender and the contributor differ from one another, the smart contract will check whether the sender has access associated with the datum, and in case access has not expired, the datum will be returned:

**Smart contract (Solidity code)**

```
function getItemDatum(
    uint _itemId,
    string memory _key
) public view returns (string memory datum) {
    if (msg.sender != items[_itemId].data[_key].contributor) {
        uint permitTimestamp = items[_itemId].data[_key].permits[msg.sender];
        require(permitTimestamp + leaseTimeSeconds >= block.timestamp, "No permit");
    }
    return items[_itemId].data[_key].datum;
}
```

## Appendix B: Interview guide

TOPIC	KEY QUESTIONS
	<b>Warm-up</b> (1 <sup>st</sup> and 2 <sup>nd</sup> round )
BASIC INFORMATION	<ul style="list-style-type: none"> <li>Name, age, occupation?</li> </ul>
EXPERIENCE	<ul style="list-style-type: none"> <li>In number of years, how would you describe your experience in:               <ul style="list-style-type: none"> <li>product data systems?</li> <li>blockchain technology?</li> </ul> </li> </ul>
CLARITY	<ul style="list-style-type: none"> <li>Do you have any questions about the concept?</li> </ul>
SENTIMENT	<ul style="list-style-type: none"> <li>Other initial thoughts about the concept?</li> </ul>
	<b>Technical design</b> (2 <sup>nd</sup> round only)
PRODUCT EXAMPLE	<ul style="list-style-type: none"> <li>How do you feel about the loader crane product item and the industry setting specified for this demonstration?</li> </ul>
PARTICIPATION PROTOCOL	<ul style="list-style-type: none"> <li>What do you think about the technical design of the participation protocol?</li> <li>Does it make sense to you?</li> <li>Is there something that jumps out as good or bad?</li> <li>Is there something that hasn't been considered?</li> <li>Is there something you would want to change about its design?</li> </ul>
FEASIBILITY	<ul style="list-style-type: none"> <li>How do you see the practical implementability of this design?</li> <li>How do you feel about its ability to scale and to facilitate the complexity and heterogeneity of real-world product data?</li> </ul>
OTHER	<ul style="list-style-type: none"> <li>Is there anything else you would like to comment about the technical design?</li> </ul>
	<b>Conceptualization</b> (1 <sup>st</sup> round: without product & industry context; 2 <sup>nd</sup> round: with said context)
TECHNOLOGY	<ul style="list-style-type: none"> <li>What potential benefits and problems do you see with the use of a blockchain smart contract to facilitate the workflow of a product life-cycle journey?</li> </ul>
GOVERNANCE	<ul style="list-style-type: none"> <li>The conceptualized PCIM platform has no centralized authority or platform provider. What benefits do you see following from this design principle? What about problems?</li> </ul>
DATA FORMAT	<ul style="list-style-type: none"> <li>The concept does not specify any particular product data format. What are your thoughts on this? Benefits? Problems?</li> </ul>
DEPLOYMENT	<ul style="list-style-type: none"> <li>What do you think about viability of the suggested method of platform deployment through an incentivized open-source participation protocol?</li> <li>Could you also comment the cross-industrial aspect?               <ul style="list-style-type: none"> <li>What kinds of problems might the concept solve in establishing a PCIM system?</li> <li>What kinds of problems might the concept not solve in establishing a PCIM system?</li> </ul> </li> </ul>

PAYMENTS	<ul style="list-style-type: none"> <li>What do you think about the suggested method of incentivizing the provision of platform data through crypto-mining payments?             <ul style="list-style-type: none"> <li>The crypto-mining payment approach would, in principle, enable intelligent product items would be able to pay for the maintenance of their own product data with electricity and CPU power. What are your thoughts on this prospect? What are the benefits and the problems?</li> </ul> </li> </ul>
LONGEVITY	<ul style="list-style-type: none"> <li>The concept suggests that due to the incentivization mechanism, the conceptualized PCIM platform could outlive product individuals and even the companies that manufactured them. What benefits and problems do you see with this idea?</li> </ul>
VERSATILITY	<ul style="list-style-type: none"> <li>Due to the open-access nature of blockchain systems, the concept should be able to maintain the product data workflow intact even in the case of dynamic supply chain structures. What benefits do you see to this approach? What about problems?</li> </ul>
SHORTCOMINGS	<ul style="list-style-type: none"> <li>What do you consider the weakest aspect of the concept?</li> <li>Are there considerations which the concept fails to take into account?</li> </ul>
	<b>Delphi (2<sup>nd</sup> round only)</b>
DELPHI	<ul style="list-style-type: none"> <li>Do any of these summarized key points in this list jump out to you as something you want to comment? For example, is there something in particular you strongly agree or disagree with?</li> </ul>

## Endnote

- <sup>1</sup> For additional information, see <<https://docs.ipfs.io/introduction/>>. Accessed on 21st of January 2020.



## References

- Ahluwalia, G. (2016). Device Democracy. The Next Generation IoT Technologies Using the Blockchain. Samsung Developer Connection. Retrieved from <https://youtu.be/hwaBM-kQeqc>
- Aitken, J., Childerhouse, P., & Towill, D. (2003). The impact of product life cycle on supply chain strategy. *International Journal of Production Economics*, 85(1), 127–140.
- Ala-risku, T. (2009). *Installed Base Information: Ensuring Customer Value and Profitability after the Sale*. Helsinki University of Technology.
- Alam, K. M., & El Saddik, A. (2017). C2PS : A Digital Twin Architecture Reference Model for the Cloud-based Cyber-Physical Systems. *Journal of LaTeX Class Files*, 14(8), 1–13.
- Ali-Yrkkö, J., Mattila, J., & Seppälä, T. (2017). *Estonia in Global Value Chains*. Helsinki. Retrieved from <https://pub.etla.fi/ETLA-Raportit-Reports-69.p>
- Anderson, C., & Zeithaml, C. (1984). Stage of the Product Life Cycle, Business Strategy, and Business Performance. *The Academy of Management*, 27(1), 5–24.
- Armstrong, M. (2006). Competition in Two-sided Markets. *The RAND Journal of Economics*, 37(3), 668–691.
- Athey, S., Parashkevov, I., Sarukkai, C., & Xia, J. (2016). *Bitcoin Pricing, Adoption, and Usage: Theory and Evidence*.
- Athey, S., & Roberts, J. (2001). Organizational Design: Decision Rights and Incentive Contracts. *American Economic Review*, 91(2), 200–205.
- Ayash, E. M. M. (2014). *Research Methodologies in Computer Science and Information Systems*.
- Azzi, R., Chamoun, R. K., & Sokhn, M. (2019). The power of a blockchain-based supply chain. *Computers & Industrial Engineering*, 135, 582–592.
- Bahga, A., & Madiseti, V. K. (2016). Blockchain Platform for Industrial Internet of Things. *Journal of Software Engineering and Applications*, 9(10), 533–546.
- Bai, C., & Sarkis, J. (2020). A supply chain transparency and sustainability technology appraisal model for blockchain technology. *International Journal of Production Research*, 58(7), 2142–2162.
- Banerjee, M., Lee, J., & Choo, K.-K. R. (2018). A blockchain future to Internet of Things security: A position paper. *Digital Communications and Networks*, 4(3), 149–160.
- Blossey, G., Eisenhardt, J., & Hahn, G. J. (2019). Blockchain Technology in Supply Chain Management: An Application Perspective. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences* (Vol. 6, pp. 6885–6893).
- Bussmann, S., & Sieverding, J. (2001). Holonic control of an engine assembly plant: an industrial evaluation. *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.*

- No.01CH37236), 5, 3151–3156.
- Buterin, B. V. (2013). *Ethereum White Paper – A Next Generation Smart Contract and Decentralized Application Platform*.
- Caillaud, B., & Jullien, B. (2003). Chicken & Egg: Competition among Intermediation Service Providers. *The RAND Journal of Economics*, 34(2), 309–328.
- Caro, M. P., Ali, M. S., Vecchio, M., & Giaffreda, R. (2018). Blockchain-based Traceability in Agri-Food Supply Chain Management: A Practical Implementation. In *IoT Vertical and Topical Summit on Agriculture – Tuscany (IOT Tuscany)* (pp. 1–4). Tuscany.
- Catalini, C., & Gans, J. (2016). MIT Sloan School of Management *Some Simple Economics of the Blockchain* (No. 5191–16). MIT Sloan Research Paper.
- Chen, S., Shi, R., Ren, Z., Yan, J., Shi, Y., & Zhang, J. (2017). A Blockchain-based Supply Chain Quality Management Framework. In: *The 14th IEEE International Conference on e-Business Engineering*. Shanghai, China.
- Cole, R., Stevenson, M., & Aitken, J. (2018). Blockchain Technology: Implications for Operations and Supply Chain Management. *Supply Chain Management: An International Journal*, 24(4), 469–483.
- Dai, H., Zheng, Z., & Zhang, Y. (2019). Blockchain for Internet of Things: A Survey. *IEEE Internet of Things Journal*, 6(5), 8076–8094.
- Dalkey, N., & Helmer, O. (1963). An Experimental Application of the Delphi Method to the Use of Experts. *Management Science*, 9(3), 458–467.
- De Filippi, P., & Loveluck, B. (2016). The Invisible Politics of Bitcoin: Governance Crisis of a Decentralised Infrastructure. *Internet Policy Review*, 5(3), 1–28.
- Dubois, A., & Gadde, L.-E. (2002). Systematic combining: an abductive approach to case research. *Journal of Business Research*, 55(7), 553–560.
- Dubois, A., & Gadde, L. (2014). Systematic combining – A decade later. *Journal of Business Research*, 67(6), 1277–1284.
- Eisenhardt, K. M. (1989). Agency Theory: An Assessment and Review. *Academy of Management Review*, 14(1), 57–74.
- Elmessiry, M., & Elmessiry, A. (2018). Blockchain Framework for Textile Supply Chain Management. In: Shiping Chen, H. Wang, & L.-J. Zhang (Eds.), *Blockchain – ICBC 2018* (pp. 213–227). Cham: Springer International Publishing.
- Elmessiry, M., Elmessiry, A., & Elmessiry, M. (n.d.). Dual Token Blockchain Economy. The Garment Use Case. In: J. Joshi, S. Nepal, Q. Zhang, & L.-J. Zhang (Eds.), *Blockchain – ICBC 2019. Lecture Notes in Computer Science. Conference paper*. (Vol. 11521, pp. 157–170). Springer International Publishing.
- Ethereum Frontier Guide. (n.d.). GitBook. Retrieved from <https://legacy.gitbook.com/book/ethereum/frontier-guide>
- Evermann, J., & Kim, H. M. (2019). Workflow Management on the Blockchain – Implications and Recommendations. *Computing Research Repository*, 1–27.

- Retrieved from <http://arxiv.org/abs/1904.01004>
- Främling, K., Harrison, M., Brusey, J., & Petrow, J. (2007). Requirements on unique identifiers for managing product lifecycle information: Comparison of alternative approaches. *International Journal of Computer Integrated Manufacturing*, 20(7), 715–726.
- Främling, Kary, Holmström, J., Loukkola, J., Nyman, J., & Kaustell, A. (2013). Sustainable PLM through Intelligent Products. *Engineering Applications of Artificial Intelligence*, 26(2), 789–799.
- Främling, Kary, Kubler, S., & Buda, A. (2014). Universal messaging standards for the IoT from a lifecycle management perspective. *IEEE Internet of Things Journal*, 1(4), 319–327.
- Galvez, J. F., & Mejuto, J. C. (2018). Future challenges on the use of blockchain for food traceability analysis. *Trends in Analytical Chemistry*, 107, 222–232.
- Gawer, A. (2009). *Platforms, Markets and Innovation*. Cheltenham, UK: Edward Elgar Publishing Inc.
- Gordon, W. J., & Catalini, C. (2018). Blockchain Technology for Healthcare : Facilitating the Transition. *Computational and Structural Biotechnology Journal*, 16, 224–230.
- Hagiu, A. (2014). Strategic Decisions for Multisided Platforms. *MIT Sloan Management Review*, 55(2), 71–80. Retrieved from <http://sloanreview.mit.edu/article/strategic-decisions-for-multisided-platforms/>
- Hagiu, A., & Wright, J. (2015). Multi-sided Platforms. *International Journal of Industrial Organization*, 43, 162–174.
- Heber, D. (2017). Towards a Digital Twin: How the Blockchain Can Foster E/E-Traceability in Consideration of Model-Based Systems Engineering. In: *21st International Conference on Engineering Design* (Vol. 3, pp. 321–330).
- Heber, D., & Groll, M. (2018). A Meta-Model to Connect Model-based Systems Engineering with Product Data Management by Dint of the Blockchain. In: *International Conference on Intelligent Systems* (pp. 280–287).
- Holmström, J., Ketokivi, M., & Hameri, A.-P. (2009). Bridging Practice and Theory: A Design Science Approach. *Decision Sciences*, (February).
- Hribernik, K. A., Rabe, L., Schumacher, J., & Thoben, K. (2006). The Product Avatar as a Product-Instance-Centric Information Management Concept. *International Journal of Product Lifecycle Management*, 1(4), 367–379.
- Huang, S., Wang, G., Yan, Y., & Fang, X. (2020). Blockchain-based data management for digital twin of product. *Journal of Manufacturing Systems*, 54(January), 361–371.
- Hukkinen, T., Mattila, J., Smolander, K., Seppälä, T., & Goodden, T. (2019). Skimping on Gas – Reducing Ethereum Transaction Costs in a Blockchain Electricity Market Application. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences* (Vol. 6, pp. 6875–6884). Retrieved from <https://>

- hdl.handle.net/10125/60123
- Jiang, T., Fang, H., & Wang, H. (2019). Blockchain-based Internet of Vehicles: Distributed Network Architecture and Performance Analysis. *IEEE Internet of Things Journal*, PP(6), 3.
- Karafiloski, E. (2017). Blockchain Solutions for Big Data Challenges A Literature Review, (July), 6–8.
- Kärkkäinen, M., Ala-Risku, T., & Främling, K. (2003). The product centric approach: A solution to supply network information management problems? *Computers in Industry*, 52(2), 147–159.
- Kärkkäinen, M., Holmström, J., Främling, K., & Artto, K. (2003). Intelligent products: A step towards a more effective project delivery chain. *Computers in Industry*, 50(2), 141–151.
- Katuwal, G. J., Pandey, S., Hennessey, M., & Lamichhane, B. (2018). Applications of Blockchain in Healthcare: Current Landscape & Challenges, 1–17. Retrieved from <https://arxiv.org/abs/1812.02776>
- Katz, M. L., & Shapiro, C. (1994). Systems Competition and Network Effects. *Journal of Economic Perspectives*, 8(2), 93–115.
- Kim, H. M., & Laskowski, M. (2018). Toward an ontology-driven blockchain design for supply-chain provenance. *Intelligent Systems in Accounting Finance and Management*, 25(1), 18–27.
- Kim, J., & Bengfort, B. (2016). *Data Analytics with Hadoop: An Introduction for Data Scientists*. Sebastopol, CA, USA.: O'Reilly Media. Retrieved from <https://www.oreilly.com/library/view/data-analytics-with/9781491913734/ch01.html>
- Korpela, K., Hallikas, J., & Dahlberg, T. (2017). Digital Supply Chain Transformation toward Blockchain Integration. In: *Proceedings of the 50th Hawaii International Conference on System Sciences* (pp. 4182–4191).
- Kouhizadeh, M., & Sarkis, J. (2018). Blockchain Practices, Potentials, and Perspectives in Greening Supply Chains. *Sustainability*, 10(10), 3652.
- Kshetri, N. (2017). Can Blockchain Strengthen the Internet of Things? *IT Professional*, 19(August), 68–72.
- Kshetri, N. (2018). International Journal of Information Management 1 Blockchain's roles in meeting key supply chain management objectives. *International Journal of Information Management*, 39(3), 80–89.
- Kubler, S., Främling, K., & Derigent, W. (2015). P2P Data synchronization for product lifecycle management. *Computers in Industry*, 66, 82–98.
- Kubler, S., Robert, J., Hefnawy, A., Främling, K., Cherifi, C., & Bouras, A. (2017). Open IoT Ecosystem for Sporting Event Management. *IEEE Access*, 5, 7064–7079.
- Lauslahti, K., Mattila, J., Hukkinen, T., & Seppälä, T. (2018). Expanding the Platform: Smart Contracts as Boundary Resources. In: *Springer Translational Systems Sciences series, Volume titled as Collaborative Value Co-creation in the*

- Platform Economy* (pp. 65–90). Springer Singapore. Retrieved from [https://link.springer.com/chapter/10.1007/978-981-10-8956-5\\_4](https://link.springer.com/chapter/10.1007/978-981-10-8956-5_4)
- Lehtonen, O., Ala-Risku, T., & Holmström, J. (2012). Enhancing field-service delivery: The role of information. *Journal of Quality in Maintenance Engineering*, 18(2), 125–140.
- Leiding, B., Memarmoshrefi, P., & Hogrefe, D. (2016). Self-managed and blockchain-based vehicular ad-hoc networks. *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct – UbiComp '16*, 137–140.
- Leitão, P. (2009). Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence*, 22(7), 979–991.
- Leng, J., Jiang, P., Liu, J., Chen, Q., & Liu, Q. (2017). Self-organizing of Product-oriented Prosumers Manufacturing Community Driven by the Blockchain Technique. *Journal of Guangdong University of Technology*, 34(5), 1–9.
- Liang, X., Shetty, S., Tosh, D., Kamhoua, C., Kwiat, K., & Njilla, L. (2017). ProvChain: A Blockchain-based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability. In: *17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing* (pp. 468–477).
- Lu, Q., & Xu, X. (2017). Adaptable Blockchain-Based Systems. *IEEE Software*, 34(6), 21–27.
- Lu, Q., Xu, X., Liu, Y., Weber, I., Zhu, L., & Zhang, W. (2019). uBaaS: A unified blockchain as a service platform. *Future Generation Computer Systems*, 101, 564–575.
- Lyly-Yrjänäinen, J., Holmström, J., Johansson, M. I., & Suomala, P. (2016). Effects of combining product-centric control and direct digital manufacturing: The case of preparing customized hose assembly kits. *Computers in Industry*, 82, 82–94.
- Mattila, J., & Seppälä, T. (2018). Distributed Governance in Multi-Sided Platforms – A Conceptual Framework from Case: Bitcoin. *Springer Translational Systems Sciences Series, Volume Titled as Collaborative Value Co-Creation in the Platform Economy*.
- Meyer, G. G., Främling, K., & Holmström, J. (2009). Intelligent Products: A survey. *Computers in Industry*, 60(3), 137–148.
- Miller, D. (2018). Blockchain and the Internet of Things in the Industrial Sector. *IT Professional*, 20(3), 15–18.
- Min, H. (2019). Blockchain technology for enhancing supply chain resilience. *Business Horizons*, 62(1), 35–45.
- Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. *Www.Bitcoin.Org*, 9.
- Naphade, M., Banavar, G., Harrison, C., Paraszczak, J., & Morris, R. (2011). Smarter Cities and Their Innovation Challenges. *Computer*, 32–39.
- Nayak, G., & Dhaigude, A. S. (2019). A conceptual model of sustainable supply

- chain management in small and medium enterprises using blockchain technology. *Cogent Economics & Finance*, 7(1), 1–23.
- Özyılmaz, K. R., & Yurdakul, A. (2019). Designing a Blockchain-Based IoT With Ethereum, Swarm, and LoRa. *IEEE Consumer Electronics Magazine*, 8(2), 28–34.
- Papakostas, N., Newell, A., & Hargaden, V. (2019). A novel paradigm for managing the product development process utilising blockchain technology principles. *CIRP Annals – Manufacturing Technology*, 68(1), 137–140.
- Pearson, J. (2018). Mining Cryptocurrency to Pay for Journalism Is Not the Worst Idea I've Ever Heard. *Motherboard*. Retrieved from [https://www.vice.com/en\\_us/article/3k7avk/mining-cryptocurrency-to-pay-for-journalism-salon-coinhive-monero-not-the-worst-idea](https://www.vice.com/en_us/article/3k7avk/mining-cryptocurrency-to-pay-for-journalism-salon-coinhive-monero-not-the-worst-idea)
- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2008). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), 45–77.
- Petkovic, M., & Jonker, W. (2007). *Security, Privacy, and Trust in Modern Data Management. Web Services: Concepts, Architectures and Applications*. Berlin: Springer Berlin Heidelberg.
- Poon, J., & Buterin, V. (2017). *Plasma: Scalable Autonomous Smart Contracts*.
- Porter, M., & Heppelmann, J. (2014). How Smart, Connected Products are Transforming Competition. *Harvard Business Review*.
- Raggert, D. (2015). The Web of Things: Challenges and Opportunities. *Computer*, 48(5), 26–32.
- Rajala, R., Hakanen, E., Mattila, J., Seppälä, T., & Westerlund, M. (2018). How Do Intelligent Goods Shape Closed-Loop Systems? *California Management Review*, 60(3), 20–44.
- Repository, I. (2016). Blockchain ready manufacturing supply chain using distributed ledger, 0–10.
- Rink, D., & Swan, J. (1979). Product Life Cycle Research : A Literature Review. *Journal of Business Research*, 7(3), 219–242.
- Rönkkö, M., Kärkkäinen, M., & Holmström, J. (2007). Benefits of an item-centric enterprise-data model in logistics services: A case study. *Computers in Industry*, 58(8–9), 814–822.
- Ruta, M., Scioscia, F., Ieva, S., Capurso, G., & Sciascio, E. Di. (2017). Poster Abstract: Supply Chain Object Discovery with Semantic-enhanced Blockchain. In: *15th ACM Conference on Embedded Networked Sensor Systems (SenSys'17)* (pp. 1–2). Delft, The Netherlands A.
- Rüth, J., Zimmermann, T., Wolsing, K., & Hohlfeld, O. (2018). Digging into Browser-based Crypto Mining. In: *Proceedings of the Internet Measurement Conference 2018* (pp. 70–76). Boston, MA.
- Saberi, S., Kouhizadeh, M., Sarkis, J., & Shen, L. (2019). Blockchain technology and its relationships to sustainable supply chain management, 7543.



- Slootweg, S. (2016). Is my blockchain a blockchain? Retrieved from <https://gist.github.com/joepeie91/e49d2bdc9dfec4adc9da8a8434fd029b>
- Sternberg, H. S., Hofmann, E., & Roeck, D. (2020). The Struggle is Real : Insights from a Supply Chain Blockchain. *Journal of Business Logistics*, 1–17.
- Tähtinen, T. (2018). *How to Turn IIoT Software Solutions into Platforms: A Case Study of Three Existing Solutions*. Aalto University. Retrieved from <http://urn.fi/URN:NBN:fi:aalto-201809034820>
- Tang, D., & Qian, X. (2008). Product lifecycle management for automotive development focusing on supplier integration. *Computers in Industry*, 59(2–3), 288–295.
- Tian, F. (2016). An Agri-food Supply Chain Traceability System for China Based on RFID & Blockchain Technology. *2016 13th International Conference on Service Systems and Service Management (ICSSSM)*, 1–6.
- Tijan, E., Aksentijevic, S., & Ivani, K. (2019). Blockchain Technology Implementation in Logistics. *Sustainability*, 11(4), 1–13.
- Toyoda, K., Mathiopoulos, P. T., Sasase, I., & Ohtsuki, T. (2017). A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain. *IEEE Access*, 5, 17465–17477.
- Trentesaux, D. (2009). Distributed control of production systems. *Engineering Applications of Artificial Intelligence*, 22(7), 971–978.
- Turk, Ž., & Klinc, R. (2017). Potentials of Blockchain Technology for Construction Management. *Procedia Engineering*, 196, 638–645.
- Valkama, P. (2020). Blockchain Product System. Retrieved from <https://github.com/peltchu/blockchain-product-system>
- Walch, A. (2019). Deconstructing ‘Decentralization’: Exploring the Core Claim of Crypto Systems. In: C. Brummer (Ed.), *Cryptoassets: Legal, Regulatory, and Monetary Perspectives* (pp. 39–68). New York, NY: Oxford University Press.
- Westerkamp, M., Victor, F., & Axel, K. (2018). Blockchain-based Supply Chain Traceability: Token Recipes model Manufacturing Processes. In: *2018 IEEE Confs on Internet of Things, Green Computing and Communications, Cyber, Physical and Social Computing, Smart Data, Blockchain, Computer and Information Technology, Congress on Cybermatics* (pp. 1595–1602).
- Wood, G. (2013). *Ethereum: A Secure Decentralised Generalised Transaction Ledger*.
- Wu, H., Li, Z., King, B., Miled, Z. Ben, & Tazelaar, J. (2017). A Distributed Ledger for Supply Chain Physical Distribution Visibility. *Information*, 8(4).
- Yu, B., Wright, J., Nepal, S., Zhu, L., Liu, J., & Ranjan, R. (2018). TrustChain: Establishing Trust in the IoT-based Applications Ecosystem Using Blockchain. *IEEE Cloud Computing*, 5(4), 12–23.
- Zhang, Y., & Wen, J. (2017). The IoT electric business model: Using blockchain technology for the internet of things. *Peer-to-Peer Networking and Applications*, 10, 983–994.

