Automation in Construction 141 (2022) 104420

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Tracking prefabricated assets and compliance using quick response (QR) codes, blockchain and smart contract technology

Wouter van Groesen *, Pieter Pauwels

Department of the Built Environment Eindhoven University of Technology, P.O. Box 513, Eindhoven 5600 MB, The Netherlands

A R T I C L E I N F O

Keywords:
Asset tracking
Quick response (QR) codes
Blockchain
Distributed ledger technology
Smart contracts
Ethereum
Building information modelling

A B S T R A C T

Construction supply chains (CSC) are characterized by their converging structure and the hierarchical payment system which is inherently related. Due to the pursuit of diverging goals and mutual interdependencies, risks and trust problems between stakeholders occur. Contracts are used to govern these complex collaborations. To track compliance to contract terms, various activities are executed by involved stakeholders. These activities do not add any value to an end-product, cause intransparency, are costly and cause conflicts. Therefore, a combination of asset tracking and blockchain technology is investigated in this paper to address these problems. A plug-and-play framework of interacting applications and a workflow to operate them are proposed. Proof of this concept is provided by developing prototypical applications and by testing them in various practice-based scenarios. Simplified asset tracking throughout supply chains was realized and linked to smart contracts. This resulted in semi-automated compliance tracking and an immutable record of transactions. Therefore, this work provides more insight into combining asset tracking and blockchain for compliance checking in the construction industry. By doing so, further automation of construction processes, increased transparency between stakeholders and a reduction of conflicts can be enabled. However, it remains clear that many other variables impact on transparency and trust; and while technology can make improvements, considerable procedural security and control measures are needed in addition. Therefore, we focus on semi-automation of workflows using distributed ledger technology.

1. Introduction

Within the construction industry, complex multidisciplinary consortia of people collaborate towards the realization of a single project. These projects are required to be completed within a specified budget and within a pre-determined time frame. Due to the complexity of construction projects, the realization phase encompasses the involvement of many specialists and suppliers of products, components and sub-elements, to construct a building [1]. These sub-elements are manufactured and installed by a wide variety of stakeholders. Therefore, construction supply chains are characterized by their length, network structure as well as the inclusion of a large number of varying suppliers [2]. This complexity and a lack of transparency cause failure costs and conflicts between stakeholders [3–5].

1.1. Background

The relationships between two or multiple stakeholders are governed by contracts. Contracts often involve: (1) a description of the work and/or products which need to be performed/delivered by a contracted party, (2) the required quality of the work/material upon delivery, (3) a time frame in which the work/delivery needs to be completed, (4) a payment scheme which is based on the level of completion of the work/delivery and (5) general terms, conditions and liabilities related to the aforementioned points [6]. The construction industry is filled with legal disputes [7]. These disputes can result from a wide range of causes, such as ambiguity in the terms of the contract, late payments and late or under performing delivery of work [4].

Because of the complexity and large number of stakeholders involved in construction projects, the number of contracts and their complexity is high as well. These contracts are mostly managed internally in each of the stakeholder organizations. Furthermore, each stakeholder executes various activities to check whether other stakeholders comply to their contract terms and abide by their obligations. Examples of these activities are quality checks, planning monitoring and payment verification [4,6]. Because each stakeholder individually checks whether products

* Corresponding author.
E-mail address: w.v.groesen@tue.nl (W. van Groesen).

https://doi.org/10.1016/j.autcon.2022.104420
Received 8 September 2021; Received in revised form 5 May 2022; Accepted 6 June 2022
Available online 15 June 2022

(assets) or services are delivered on time, asset management and tracking is executed separately. Furthermore, these asset tracking activities and their registration are often executed via inefficient manual procedures. Because these activities and their registration lack transparency and are sensitive to errors, they cause communication gaps between stakeholders [2]. These communication gaps are the basis for trust problems and defects which result into conflicts and failure costs [3]. Maintaining these asset tracking systems within a fragmented supply chain is not considered to add any value for the final customer [8,9]. Therefore, these inefficient and decentralized activities have to be addressed to increase trust and reduce conflicts. Although these inefficient and decentralized activities are not the only base for lacking trust between stakeholders, addressing them would significantly contribute to reducing conflicts.

1.2. Asset tracking and blockchain

Current construction engineering management suffers numerous challenges regarding trust, information sharing, and process automation [10]. Physical Asset Tracking (PAT) and Digital Asset Management (DAM) thus have to be addressed to increase transparency and trust. Various technologies and workflows to aid towards this purpose are promising and existing [11]. Geospatial tracking technologies and related asset management systems are for example widely applied outside of the construction industry to replace manual asset tracking [12]. Increasing examples can be found of its application in construction use cases as well [13,14]. Within the construction industry, increased collaboration between stakeholders and exchange of information in BIM models enhances current practice [15]. The combination of BIM models with real-time data and their benefits are also demonstrated in other work [16]. Other emerging technologies which are indicated to contribute towards transparency and trust are blockchain and smart contracts [17–19].

Because the use of blockchain technology enables the construction of an distributed and immutable log as single source of truth, its application in complex collaborations is argued to be favourable [19,20]. Because transparency and traceability is created in a jointly managed ledger, its application reduces communication gaps and conflicts between stakeholders [11,21]. As an extension, smart contracts which run on the blockchain can be deployed to semi-automate the comparison between reality and contract terms. Furthermore, the need for intermediate stakeholders (banks) and delays are eliminated as such contracts instantaneously executes upon fulfillment of contract terms [6,22]. The combination of blockchain, smart contracts and asset tracking is not a new concept. Previous work has demonstrated the merits and drawbacks of combining them within and outside of the construction industry [11,23,24]. Although there are still hurdles for the joint application of these technologies, it is widely acknowledged that the construction industry is ready for just such a socio-technical shift [8,25].

Therefore, further exploration of these concepts, their combination and application in practice is required. We thus aim to address challenges such as trust and transparency to some extent in this work by combining and testing these technologies in an industry use case. In order to do so, we aim to semi-automate asset and compliance tracking activities for large prefabricated assets throughout the supply chain.

1.3. Research outline

First a broad literature review is conducted on asset tracking, management, blockchain and smart contracts. Existing research on the combination of these technologies is reviewed, a research gap is identified after which a conceptual model is presented in Section 2. The proposed conceptual model is expanded into a framework of interacting applications. Several individual static and dynamic prototypes are developed and are presented in Section 3. Mutual interaction between these prototypes is enabled to semi-automate compliance checking and payments. These prototypes and the interaction between them are later tested in four distinct industry-based scenarios as presented in Section 4. The article is concluded in Section 5 with an overview, conclusions, and future work.

2. Literature review

First we will discuss the proper definition of assets and the systems which deal with them in Section 2.1. In Section 2.2 we discuss how digital assets are managed in the built environment by means of BIM models. The possible technologies for tracking physical assets by means of sensors is discussed in Section 2.3. The current state and possibilities of distributed ledger technology and smart contracts are presented in Section 2.4. The literature review is concluded with the identification of a research gap and presentation of a research model in Section 2.5 and Section 2.6 respectively.

2.1. Assets and asset systems

An asset is defined by ISO (2014:55000) as: *"an item, thing or entity which is of potential or actual value to an organization"* [26]. The value of such an asset may vary during its lifecycle as well as between organizations. The value of an asset can include the consideration of risks and liabilities. Therefore, the value can be positive or negative at different stages of an asset's life. Assets as well as their value can be subdivided between tangible or intangible and financial or non-financial asset types. Asset types are a group of assets which share common distinguishable characteristics which define them as a group (e.g. digital and physical). Physical assets are considered to be tangible assets and can refer to equipment, materials, properties or human resources held by a company. Digital assets are intangible non-physical assets which can refer to agreements, use rights and intellectual property [26–28].

Furthermore, according to ISO (2014:55000), an asset system consists of: *"assets which interact or are interrelated as they share common properties"* [26]. Digital assets (e.g. model of a wall) can be interrelated to physical assets (the physical wall) and thus are considered to be an asset system. Critical assets and critical asset systems are considered to be assets which are of significant impact on the achievement of goals for an organization. Critical assets can be: (1) safety-critical, (2) environmental-critical as well as (3) performance-critical and can be related to legal requirements and obligations in contracts [15,29].

Within a built environment context, prefabricated construction elements can be considered as physical assets, while their designs can be considered as digital assets. As they are interrelated in terms of their common identity, they can be considered as an asset system. If both assets are able to communicate in real time, the digital asset is a digital twin. Prefabricated elements generally are of high value as their creation requires substantial investments by multiple stakeholders. During the construction phase of a project, activities related to prefabricated elements can be of high importance due to their presence on the critical path of an execution planning. Prefabricated elements can therefore be considered as performance-critical assets during the construction phase.

2.2. Digital asset management

The usability of a BIM model in varying phases is expressed by the level of model development (LOMD) as depicted in Fig. 1. Digital assets

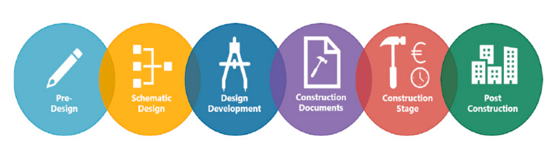

Fig. 1. BIM levels of model development (LOMD).

which are represented in BIM models as an information system are transferred into physical reality during the construction phase [30]. Therefore, the transfer from “as-planned” LOMD 400 (including planning and cost data) to “as-built” LOMD 500 is initiated during this phase. As physical asset data is processed into the information system, the “as-built” state is represented in the digital environment. Such an as-built model can be used both during and after the construction phase for performance and lifecycle management [31]. To enable successful performance and lifecycle management, however, the information system has to include physical asset data which is regularly updated [32,16]. To advance towards an updated “as-built” model and thus a (near) real-time update of a digital environment with data from the physical environment, sensor technology has to be implemented. BIM tools can be considered as a possibility to advance the as-planned to the as-built state [33]. However, implementational barriers persist in combining IoT data with BIM models [24]. Although such a model does not necessarily need to be realized through BIM tools, enabling the ability to consume BIM data sets in various formats is preferable for various reasons (e.g. obligations related to planning, cost and contract data) [32]. Alternative technologies which enable the achievement of the same goal (e.g. DLT or databases) are available and promising [16,31,32,34].

2.3. Physical asset tracking

To automate the measurement of progress and thereafter compliance to contract obligations in terms of planning, individual (prefabricated) elements can be identified through geospatial tracking technologies [9,12]. Geospatial tracking technologies enable interactions with physical assets through communication with tags or sensors (geo-tagging). These tags/sensors are coupled to the identity of individual assets and, therefore, individual locations can be identified (either through scanning devices or the tags themselves). Manipulation of the status of individual assets can therefore occur through either location or software in order to communicate progress [35,36]. Based on on/off-site progress in comparison to the desired progress embedded in contract obligations, compliance/non-compliance can be determined [6,11,37]. Geospatial asset tracking technologies which are widely applied in industry consist of: (1) Quick Response (QR) codes, (2) Radio-Frequency Identification (RFID), (3) Ultra-Wideband (UWB) and (4) Global Positioning Systems (GPS).

An alternative to geospatial tracking technology for the measurement of on-site progress and therefore, compliance, is the deployment of imaging technologies [12,38]. Imaging technology is used for capturing digital images in order to generate 3D information regarding various objects on construction sites. This 3D information is used for construction progress analysis by comparison with a 3D representation of the desired construction progress [9,38]. As demonstrated in the work of El-Omari and Moselhi (2008), imaging technologies capture the “as-built” state/situation of multiple assets in one frame. After a frame is compared to a 3D “as-planned” situation, the identity of individual assets as well as discrepancies in terms of construction progress can be determined [9,35,38,39]. Based on this comparison, on-site progress can be determined and compliance/non-compliance can be communicated as well. Imaging technologies consist of: (1) photogrammetry, (2) 3D laser scanning, (3) videogrammetry and (4) range images [9,12,38].

Geospatial technologies are more suitable for application throughout a supply chain as they can collect data both on- and off-site [10,13,39]. Because most of the geospatial technologies require the operation of a scanning procedure, semi-automation of data acquisition can be realized [12,40]. As shown by Yavaraj and Sangeetha (2016), GPS can enable full automation as no manual scanning procedures are required to acquire data [41]. In terms of time efficiency of these procedures, QR scanners are restricted to the acquisition of data from a single physical asset, but do not require special equipment and hence, investments. RFID and UWB can acquire data from multiple assets simultaneously, but do require additional equipment (e.g. a RFID scanner). GPS does not require additional equipment (besides sensors) and can collect data related to multiple assets as well.

Although geospatial technologies are suited for the acquisition of data throughout supply chain stages, it is undesirable and infeasible to apply this technology to all construction components (e.g. a tag/sensor for every brick). Therefore, geospatial technology is best suited for tracking large prefabricated elements throughout the supply chain by conducting (semi) automated procedures [14,35]. Kopsida et al (2015) indicates that the installation and maintenance of RFID tags requires a lot of time which is undesirable [12]. Maintenance of these tags can be an issue if they are applied for longer periods (e.g. throughout a buildings lifespan). When applied for a short period (e.g. during the execution phase only), the need for maintenance is considered unlikely [13,40].

In terms of application of imaging technologies, the usability is restricted to the capturing of on-site conditions [9]. Therefore, only the execution status of construction components can be registered. Data acquisition by imaging technologies can be automated if no human intervention is required (e.g. setting up the scanner) [12]. Furthermore, investments in terms of resources to install and maintain tags are not required [40]. Because scanning technology does not require the presence of tags in order to verify the identity of a construction component, the technology is more viable for measuring construction progress (4D) [33]. As discussed previously, it is infeasible and undesirable to equip every brick with a sensor in order to measure progress through geospatial technologies.

In order to advance towards (semi) automated compliance tracking, the monitoring of progress in the physical world has to be conducted both on-site and off-site to cover the entirety of the supply chain (digital twinning) [16,32]. Therefore, geospatial technologies have to be considered for large and valuable construction components (prefabricated elements). However, if we do not restrict ourselves to progress measurement of prefabricated components, but want to enable full construction progress measuring in the future, the additional use of imaging technology is definitively required [9,33].

2.4. Distributed ledger technology

DLT can be defined as a consensus of replicated, shared, and synchronized digital data, geographically spread across multiple sites, countries, or institutions where there is no single entity in control [42]. Nawari et al. (2019) defines distributed ledger technology as: *“a digitized, decentralized public ledger of data, assets and all pertinent transactions that have been executed and shared among participants in the network”* [19]. DLTs consist of a chain of blocks which contain information and are encrypted through a cryptographic hash function provided by an algorithm. Transactions representing value are grouped within these blocks, are verified and validated through a consensus mechanism within a distributed peer-to-peer network [18]. Distributed Ledger Technology (DLT) thus enables opportunities in which data is stored securely, transparently, immutably and distributed. DLT technology is currently used most intensively within industries which rely heavily on financial transactions and the exchange of information [25]. Due to the immutability of the DLT, transactions within the system are secure and can aid to reduce lack of trust [11,17,43]. Nakamoto (2019) adds that DLT provides certainty marked by complete consensus, provenance, finality, and immutability [44]. Successful implementation of DLT technology is considered to profoundly change not only services and products, but also the way in which work is structured. The functionalities of a DLT system allow for: (1) the protection of participants, (2) the reduction of trust problems, (3) the removal of information barriers and (4) the avoidance of lawsuit costs [14].

2.4.1. Smart contracts

The term smart contract was first developed and defined by the cryptographer Nick Szabo in 1997 [45]. Smart contracts are defined as:

"a computerized transaction protocol that executes the terms of a contract". The general objectives of smart contract designs are to satisfy common contractual conditions, minimize expectations and minimize the need for trusted intermediaries [45]. Mason (2017) states that smart contracts can be defined as *"contracts that are fully executable without human intervention"* [6] or *"self-enforcing, monitoring external inputs from trusted sources in order to settle according to the contract's stipulations"* [46]. Smart contracts are considered to be a layer on top of DLT. Due to their capabilities of expressing requirements into computer readable scripts which automatically execute actions upon fulfilment of pre-set conditions, the application of such a technology might yield significant benefits if combined with asset tracking and management [11,17,43]. A combination of DLT (ledger technology), BIM (digital execution plan) and smart contracts (computer readable scrips) can be deployed for automation of some aspects of traditional contract clauses. Mason (2017) discusses that the key legal document itself, the contract, has the potential to become automated although some challenges (e.g. quality assurance) are identified [6]. Shojaei et al (2019) supports this argument and argues that quality-related compliance processes still require human intervention to some extent [7]. Although various efforts have been observed focusing on (e.g. automated quality assurance processes), the merits of the elimination of all human interference is undesirable [6,47]. Consequently, capitalization on all benefits of distributed ledgers might be difficult to achieve in the construction industry. Inefficient manual compliance activities related to product delivery/construction progress, financial obligations (e.g. payments) as well as quality-related compliance processes, can be (partially) automated. Implementation of Internet of Things (IoT) based sensor technology enables tracking of physical assets [43]. Hence, IoT encompasses possibilities for further automation as a supplement to integrated DLT, smart contracts and BIM-based processes [7,11,48].

2.4.2. Ethereum blockchain

Smart contracts are generally embedded on a blockchain and operate based on data which is present on the blockchain on which it resides. Not all blockchain networks are equipped to incorporate smart contract functionalities such as the renown Bitcoin blockchain. One of the major blockchains which does enable smart contract functionalities is the Ethereum blockchain [49–51]. Besides the hardware node on which an Ethereum client runs, the infrastructure of the Ethereum (ETH) blockchain network is divided into six layers. These layers consist of: (1) infrastructure, (2) data, (3) consensus, (4) network, (5) execution and (6) application. These layers are depicted in Fig. 2.

The infrastructure layer consists of nodes (one for each supply chain party) organized in a peer-to-peer (P2P) network. All nodes share a single distributed and decentralized ledger. Each node is related to an externally owned account (EOA) with a unique account address. Data related to each individual address (e.g. balance) is stored in separate account state tries (tree-like structures), which are merged in a single

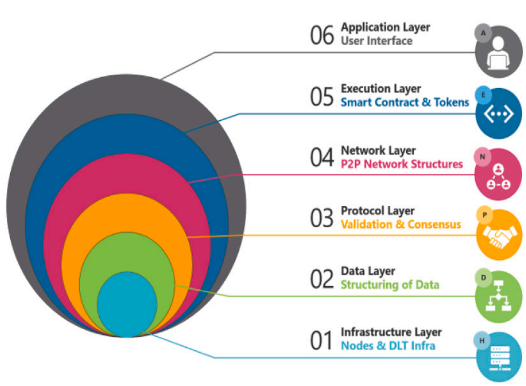

Fig. 2. Ethereum blockchain layers.

world state trie [50–52]. Subsequently, the world state trie at a specific point in time is stored in blocks. These blocks are chained by means of a hashing algorithm which provides immutability and security to the shared ledger. Hence, a single source of truth is created. As consensus between nodes needs to be reached in order to append blocks to the shared ledger, a consensus algorithm is utilized in the protocol layer. The Proof of Authority (PoA) consensus algorithm is best suited for built environment collaborations, as power is equalized between nodes independently of node computing power or stake sizes.

The PoA system thus differs from other consensus algorithms such as Proof of Work (PoW) or Proof of Stake (PoS) [51]. This is mainly due to the presence of validating nodes which do not stake value but reputation instead. Thus, the network is secured by pre-defined nodes of which the identity can be known to other nodes in the system. Because of the lack of anonymity and the absence of rewards for miners, the PoA algorithm is better suited to the needs of private consortium-based networks [53]. Considering the network layer, the P2P network structure which is best applicable in the built environment is the consortium-based permission model [21,54]. This structure allows all nodes to view transactions and commit transactions to blocks. This can contribute towards more transparency and potentially to the development of trust [55]. Writing capabilities of transactions are restricted to nodes which have a stake in a transaction. Therefore, scalability of the network as well as counteraction of mistakes and misuse is ensured [49,50,56].

2.4.3. Application layer

The final layer of the ETH blockchain infrastructure is the application layer, where users interact with the distributed ledger. The application layer is connected to the execution layer on which transactions with smart contracts are executed. The smart contracts which can compare the planned to the real state by means of computations resides within the execution layer. Smart contracts are developed, compiled and deployed subsequently on the blockchain by an EOA. EOAs are able to interact (input values) by means of pre-signed transactions with contract functions [51]. These transactions can be scheduled for execution in the future by means of the Ethereum Alarm Clock.[1] Based on these inputs and the smart contract computation code, (semi) automated result calculation is realized (payments and notifications).

2.5. Research gap

Other research has been conducted on blockchain and smart contract technology for asset tracking in construction. Lee et al. (2021) propose a combination between digital twins and blockchain technology to ensure traceable data communication [57]. Within their research, a blockchain network is created on Microsoft Azure and interaction with a digital twin in Unity is established. Hypothetical PAT data is communicated into a digital twin in near real-time, after which data is recorded on the chain. This enables the creation of a single source of truth between collaborating stakeholders, more effective progress tracking and transparency. Although semi-automated payments are not realized, a single source of truth can enable this functionality [57–59]. The potentials of applying smart contracts to integrate material and financial flows throughout the supply chain are widely recognized [54,60]. Its actual application in working examples of these combined technologies however is still scarce [10,23,25].

Most research combines smart contracts with asset tracking aiming at on-site progress monitoring (between contractors and client). Implementation throughout the entire supply chain (including manufacturers and sub-contractors) is not always considered [11,21,22]. Integration of both on-site and off-site asset tracking is however essential to obtain the benefits incurred by the use of blockchain. Furthermore, these asset tracking techniques are mostly hypothetically integrated by artificial

[1] https://www.ethereum-alarm-clock.com/

creation of an as-built model [5,20,22,57,61]. Although this provides us with more insight into combining PAT and DLT, these insights have to be verified by means of actual application. The application of smart contracts and asset tracking techniques in construction supply chain use cases is limited and thus, have to be addressed.

2.6. Research model

We aim to obtain a better understanding of how tracking techniques, blockchain and smart contracts can be jointly deployed for asset/compliance tracking throughout the supply chain. In order to do so, we intend to combine QR code tracking outputs (as-built state) with the as-planned state in smart contracts. Therefore, digital (asset delivery) plans, DLT, smart contracts and physical asset tracking systems will be combined as depicted in Fig. 3.

To enable automated compliance tracking and payments, there needs to be (near) real-time communication between the digital and the physical. Contract obligations (planning and costs) have to be included in an information system to establish the as-planned state (e.g. LOMD 400 BIM model). As physical activities are conducted, data on the status of physical assets needs to be acquired, communicated and processed into the digital environment [16,31,32]. Within the digital environment, automated data analyses need to determine whether assets in the physical environment are either compliant or non-compliant to the as-planned state of the twin elements in the digital environment. Based on the outcome of this comparison, the digital environment is able to either reward (financially) or communicate non-compliance, to enable corrective actions in the physical environment. Geospatial technologies enable (semi) automation of data acquisition in the physical environment and processing it into the digital environments. Semi-automated comparisons between the planned and actual state, including feedback as a consequence (e.g. payments) are less apparent. Mason (2019) states that smart contracts can be: (1) independent (e.g. related to a single physical asset), (2) equipped with a clear functionality (e.g. checking of location and planning compliance) and (3) executed based on protocols (e.g. financial transactions between stakeholders) [37]. Distributed ledger technology (DLT) and smart contracts can thus aid towards this purpose. Insight into smart contract transactions (e.g. through a dashboard) is required to enable industry stakeholders to anticipate. To test the rigidity of such a system and to gain a better understanding of its operation in practice, applications have to be tested in industry use-cases.

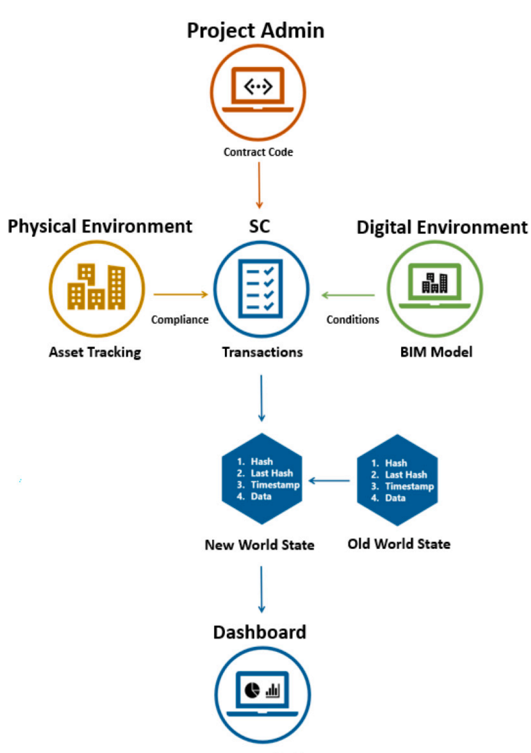

Fig. 3. Proposed smart contracting framework.

3. Smart contract(ing) framework

Based on the conducted literature review in Section 2, we propose to combine PAT, DAM and DLT. In a simplified framework which is presented in Fig. 3, we propose such a connection to enable leveraging of the advantages of blockchain technology. The combination of these environments by means of applications enables industry partners to use various (existing or novel) asset tracking/management systems, and to connect them to smart contracts and blockchain technology. The intended purpose of applying these applications and DLT together, is to reduce the number of manual compliance checking activities and to create an immutable ledger of actions/transactions. The main goal of the presented research is thus to provide initial proof of work. The framework for the combination of applications to achieve these goals, is depicted in Fig. 4.

To achieve semi-automated compliance checking by smart contracts, the following functionalities need to be met: (1) the migration of data onto and from the blockchain shall be enabled in asset tracking/management applications, (2) an immutable log of activities (asset states) shall be generated on the blockchain, (3) smart contract shall be able to interpret the results of a comparison between asset states and contract stipulations, (4) semi-automated distribution of tokens (money) to stakeholders shall be realized and (5) the combined deployment of these applications shall be rigid enough to cope with multiple industry-based scenario's.

The presented framework is first divided into several individual components which are developed into individual static prototypes (visual mock-ups), after which dynamic prototypes are developed (functional). These prototypes are verified through alpha testing within the development team that includes university and industry partners. After applications are sufficiently tested, individual dynamic prototypes are further developed. These development efforts focus on the mutual interaction between applications to enable semi-automation of compliance checking activities and payments. Overall, the agile Incremental Prototype Development (IPD) methodology is followed. This method is often used in projects that include a lot of software development. IPD aims for the design and development of so called 'artifacts' or objects which aid in solving problems in practice [62,63]. Each of the components in this framework is developed in this research, and will be explained one by one in the following sections. The development of a physical asset tracking application to acquire current asset states, is discussed in Section 3.1. In Section 3.2, the input of contract parameters in a BIM model and their extraction is discussed. The centralization of current asset states and contract parameters into a database is discussed in Section 3.3. The link between details in the database and smart contracts by means of an oracle is presented in Section 3.4. The development and utilization of smart contracts and the Ethereum blockchain are discussed in Section 3.5 and Section 3.6 respectively. EOA accounts to enable transactions with smart contracts are retained on Metamask.[2] which is discussed in Section 3.7. In Section 3.8 the development of a dashboard WebApp to enable monitoring smart contract results is presented.

Finally, the developed prototypes and the interaction between them are tested in four distinct scenarios, to verify their functionality in Section 4. These practical scenarios are developed in collaboration with a large contractor in the Netherlands. This industry partner proposes

[2] https://metamask.io/

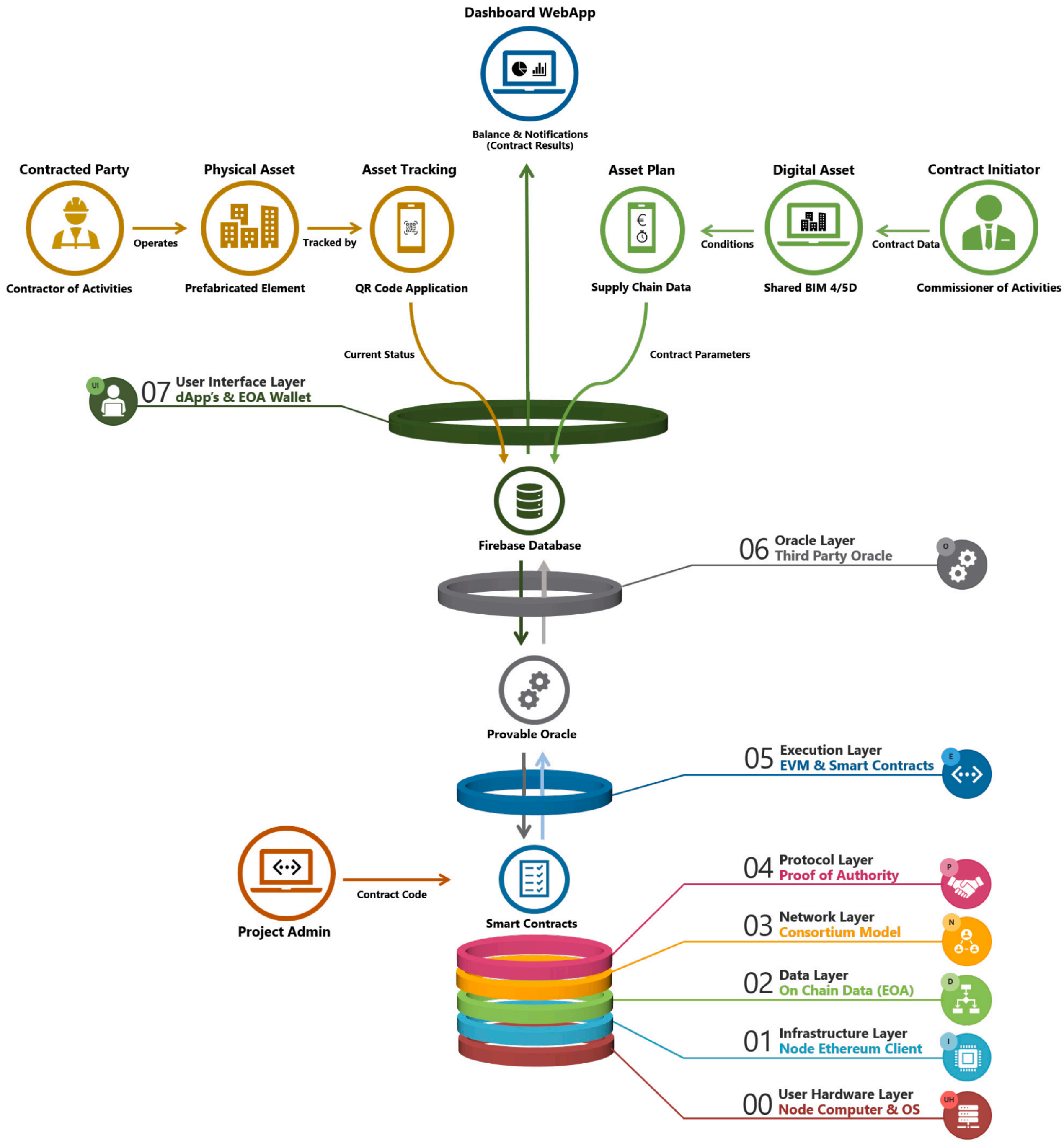

Fig. 4. Smart contracting application framework.

scenarios based on their experience with an asset tracking pilot. During alpha and beta testing rounds, the created artifacts are tested by means of these scenarios. During these tests, the satisfaction of pre-determined constraints/requirements are monitored and recorded. Thus, structured experiments are executed in both cases.

3.1. Physical asset tracking

To enable construction stakeholders to acquire and manipulate physical asset data, a fully functioning mobile QR-code application is developed. The development of a QR-code application instead of RFID application is preferred due to a shorter development time. A static UI storyboard of the application is created which is verified with industry partners. After sufficient verification and various design iterations, the QR application is coded using the MIT App Inventor application.[3] The dynamic application enables capturing physical asset states by means of scanning asset GUIDs embedded in QR-codes. The available states in the application consist of: (1) manufactured, (2) delivered, (3) ready for assembly, (4) assembled, (5) ready for verification and (6) verified.

The PAT application is designed to be used by: (1)manufacturers, (2) sub-contractors, (3) main contractors and (4) clients. Users are able to: (1) login onto the application, (2) select the desired asset state (displayed states dependent on login credentials), (3) scan the assets QR-code and (4) review the new status of the asset. If non-compliance to quality-related obligations is identified by a stakeholder: (1) a rejection

[3] https://appinventor.mit.edu/

status is selected, (2) the quality defect is captured by means of a camera and (3) a generated message is shared with other stakeholders. When the aforementioned procedure is successfully completed, the asset data is stored on a cloud database. The UI of the developed PAT application and described functionalities are depicted in Fig. 5.

3.2. Digital asset management

Besides physical asset data (current state of assets), smart contracts require related contract obligations or rules to function. To enable a comparison between these rules and reality, the smart contract has to be provided with parameters by the contract owner. These parameters are also required to enable the transfer of tokens (money) from one stakeholder onto another. To ensure a link with current industry practice, a jointly managed Revit model was used in which these parameters are centralized by supply chain stakeholders. These parameters consist of (1) a target (planned) date for every asset, (2) the value of the obligated payment after completion of specific asset states and (3) the EOA addresses and wallets of involved stakeholders on the blockchain which enables token transfers. After processing these parameters into the Revit model, schedules are generated and converted into JSON format. These JSON files are subsequently used to insert the contract parameters into a cloud database. The used Revit model and included contract parameters are presented in Fig. 6.

3.3. Firebase cloud databases

After the status of physical assets is successfully updated with the QR-code application and contract parameters are extracted from a Revit model, data is transferred to a cloud database. Besides contract parameters, the physical asset data which is stored in this database consist of (1) an assets GUID, (2) the date/time of a status mutation and (3) the location of an asset. In the cloud database, the data is structured in (1) a history log which contains a record of all previous asset states, (2) a status log which only contains the current asset state and (3) an obligation log which contains all contract parameters. Because a smart contract oracle can be directly linked with the API of Google Firebase,[4] this platform is used to accumulate the data. The described routine via a Firebase database is furthermore chosen to eventually enable dynamic web-based updates between a BIM model and the distributed ledger in real time. Because a manual workflow invalidates the purpose of using a distributed ledger, such a link can avoid manual file exports and imports. An example of the exported JSON structure containing contract parameters is depicted in Fig. 7.

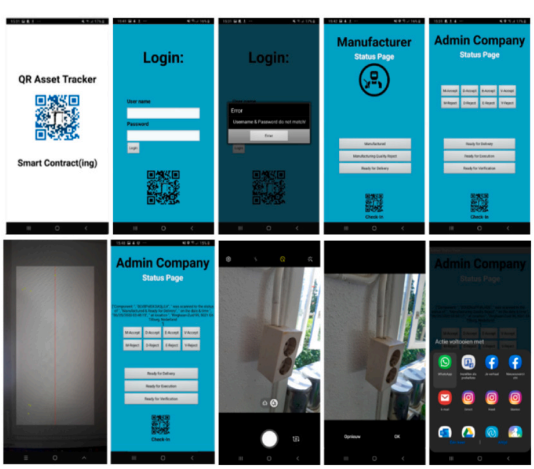

Fig. 5. Physical asset tracking application.

Supply Chain Schedule (as-planned)

Property	Value
Client	0x813620c3b12264FF46D88a3F9F...
DeliveryDate	18-05-2020
DeliveryPrice	100
ExecutionDate	20-05-2020
ExecutionPrice	125
MainContractor	0xebA00DEb7bCe7A1Ff24A97E76...
Manufacturer	0xa5A2ac899e51aE106509Ef997D...
Sub-Contractor	0x64eb7a27590989769179342f0F...
Type IfcGUID	0LV8Pid0X3IA3jJLVDPidY
VerificationDate	25-05-2020
VerificationPrice	150

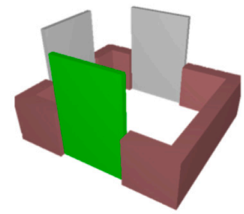

Fig. 6. Revit model and contract parameters.

3.4. Oracle

Due to technical factors which impede the import of data external to the blockchain environment, an oracle service is utilized (layer 6 in Fig. 4). An oracle enables the extraction of data from online databases [56]. To extract data from Google Firebase, the Provable Oracle Service.[5] is used. The Provable Oracle Service and Firebase URLs are embedded into a separate smart contracts that are coded and compiled in the Ethereum Remix environment.[6] The Ethereum Remix environment is a platform on which users can code smart contracts and interact with various test networks. To enable the retrieval of data attributed to a specific asset, the assets GUID is appended to the Firebase URL in the smart contract. Functions which are embedded in the smart contract successfully enable the retrieval and import of desired data into the DLT environment.

3.5. Smart contract

Besides a smart contract which retrieves data from Google Firebase, a second smart contract is developed which contains the capabilities in its code required to compare the "as-planned" state of assets to the "as-built" state. To do so, a client stakeholder first extracts contract obligations from the oracle smart contract and imports them into the second. The client stakeholder then also deposits a number of tokens (money), in correspondence with the contracts value. A manufacturer or contractor stakeholder also interacts with the functions in this smart contract by importing current asset state data. Based on the semi-automated comparison between both states, the smart contract identifies: (1)


```
{
   "0LV8Pid0X3IA3jJLVDPidY": {
      "DeliveryDate": "18-5-2020",
      "ExecutionDate": "20-5-2020",
      "VerificationDate": "25-5-2020",
      "DeliveryPrice": "100",
      "ExecutionPrice": "125",
      "VerificationPrice": "150",
      "Manufacturer": "0xa5A2ac899e51aE106509Ef997DdF6236BF6bde99",
      "Sub-Contractor": "0x64eb7a27590989769179342f0F7dD1845f84F711",
      "MainContractor": "0xebA00DEb7bCe7A1Ff24A97E7620d8948408EC8AA",
      "Client": "0x813620c3b12264FF46D88a3F9F4911F2Dc937d7a"
   },
   "9ae98218-0358-4959-8d8d-f7c575337397": {
      "DeliveryDate": "19-05-2020",
      "ExecutionDate": "24-05-2021",
      "VerificationDate": "26-05-2022",
      "DeliveryPrice": "110",
      "ExecutionPrice": "135",
      "VerificationPrice": "165",
      "Manufacturer": "0xa5A2ac899e51aE106509Ef997DdF6236BF6bde99",
      "Sub-Contractor": "0x64eb7a27590989769179342f0F7dD1845f84F711",
      "MainContractor": "0xebA00DEb7bCe7A1Ff24A97E7620d8948408EC8AA",
      "Client": "0x813620c3b12264FF46D88a3F9F4911F2Dc937d7a"
   },
```


Fig. 7. Contract parameters json format.

[4] https://firebase.google.com/

[5] https://provable.xyz/

[6] https://remix.ethereum.org/

compliance, (2) quality issues or (3) planning-related non-compliance. If compliance is identified, the smart contract: (1) rewards a specific stakeholder with deposited tokens and (2) emits a notification of success. If non-compliance or quality issues are identified, the smart contract does not transfer tokens and emits an error message. This error message includes: (1) the GUID of the asset, (2) involved stakeholder EOAs, (3) type of defect and (4) contract value. Based on these error messages, stakeholders are able to change contract parameters to ensure later fulfillment of the contract.

3.6. Distributed ledger

Because smart contracts are developed in the Ethereum Remix environment and deployed on the Ethereum Goërli test net,[7] transactions with smart contracts by EOAs are registered on an operational blockchain. Although various test net environments are available, the Goërli test net is selected due to its similarities with the main Ethereum blockchain network. Because smart contracts are deployed on an operational blockchain, (1) transactions with these contracts, (2) computation results and (3) smart contract notifications are immutably stored on the blockchain. Because of this, developed smart contracts and the interactions with these contracts are eligible for review on the Goërli Etherscan webpage.[8] Events related to one of the used smart contracts are visible in the *"events"* tab.

3.7. Metamask

When stakeholders interact with the functions in a smart contract, transactions need to be signed to verify the identity of a stakeholder. Furthermore, stakeholders need to be able to deposit or receive tokens from the smart contract which is stored in a Smart Contract Account on the blockchain. To enable stakeholders to do so, stakeholders need to possess an Externally Owned Acount (EOA). An EOA is linked to an unique account on the blockchain and its own crypto wallet. The creation and operation of an EOA normally requires the operation of an Ethereum node client on a computer. This node enables the user to participate in a blockchain network. Due to the requirement of multiple nodes to execute experiments, nodes for each stakeholder are required. To reduce complexity, EOAs are created for each stakeholder in the Ethereum Remix environment and linked to the Metamask application.[9] The Metamask application functions as an indirect node and provides users with an EOA, wallet and user interface without requiring them to install or run individual nodes locally. In Metamask, stakeholders are able to: (1) interact with smart contract functions, (2) sign transactions and (3) receive/deposit tokens into a smart contract.

3.8. Dashboard webapp

Although transactions are recorded on a blockchain, data recovery from blockchains is far from ideal. Because it is difficult to review or extract information contained on the blockchain, a dashboard is required. Such a dashboard can be intuitively used by construction stakeholders in regular contruction projects. Such a dashboard needs to provide insight into the results of smart contract execution. With this dashboard stakeholders are able to manage their projects because it enables them to review current asset states, contract parameters and token transfers. Although dashboard applications are available for the Ethereum Main Net,[10] they are not available for test nets. Therefore, a dashboard WebApp.[11] is constructed which consists of: (1) a front-end user interface, (2) a back-end which contains computation capabilities similar to the developed smart contracts and (3) a controller which enables interaction between the UI and back-end. In Firebase, asset data in each log is exported into JSON files. These JSON files are subsequently uploaded via the user interface of the dashboard WebApp. Based on the data contained in these files the dashboard displays: (1) the history log of asset states, (2) all current element states, (3) an overview of token transfers and (4) a diagram displaying the number of non-compliant assets. Due to the comparison between the *"as-planned"* and *"as-built"* state of a project in the back-end, stakeholders are able to quickly review the operations, finances and contracts in the front-end.

4. Testing and validation

While testing and validation of individual components with industry partners already took place during the development of all components in Section 3, the interaction between these components needs to be tested as well. It is required to test whether the constructed framework presented in Fig. 4, functions in accordance to the functionalities described in Section 3. These functionalities are listed hereafter.

1. Asset tracking and management application shall be able to migrate data onto and from the blockchain
2. Applications shall be usable in multiple practice-based scenarios
3. An immutable log of activities and payments shall be created on the blockchain by smart contracts
4. Semi-automated distribution of tokens (money) to stakeholders shall be realized

4.1. Preparation

Because one of these functionalities involves the applicability in practice, multiple practice-based scenarios are developed. These scenarios are drafted based on PAT pilot data provided by the VolkerWessels company and are described hereafter.

1. Full compliance and thus successful delivery, execution and verification of an asset
2. Non-compliance due to violation of planning-related contract parameters
3. Non-compliance due to violation of quality-related contract parameters
4. Incorrect use of the PAT application and thus a missing status

To test each scenario, a fictive design is constructed which contains four prefabricated assets as depicted in Fig. 6. Each of these assets is attributed to an individual scenario and is equipped with a QR-code containing a GUID. Furthermore, a smart contract is developed and deployed for each of these assets. In order to test whether desired results are yielded in each scenario, pre-conditions and post-conditions are drafted. These conditions include the expected token balance of each stakeholder before and after the execution of all scenarios.

4.2. Execution

Interactions with smart contracts during test execution are depicted in Fig. 8 and will be described hereafter.

1. In the pre-conditions the client EOA possesses all tokens (money) and transacts them into the smart contract for each scenario

[7] https://goerli.net/
[8] https://goerli.etherscan.io/address/0x185ca833440b4a71341fd0b3b3acecb2f420a83e
[9] https://Metamask.io
[10] https://https://ethereum.org/en/
[11] https://www.smartcontracting.xyz

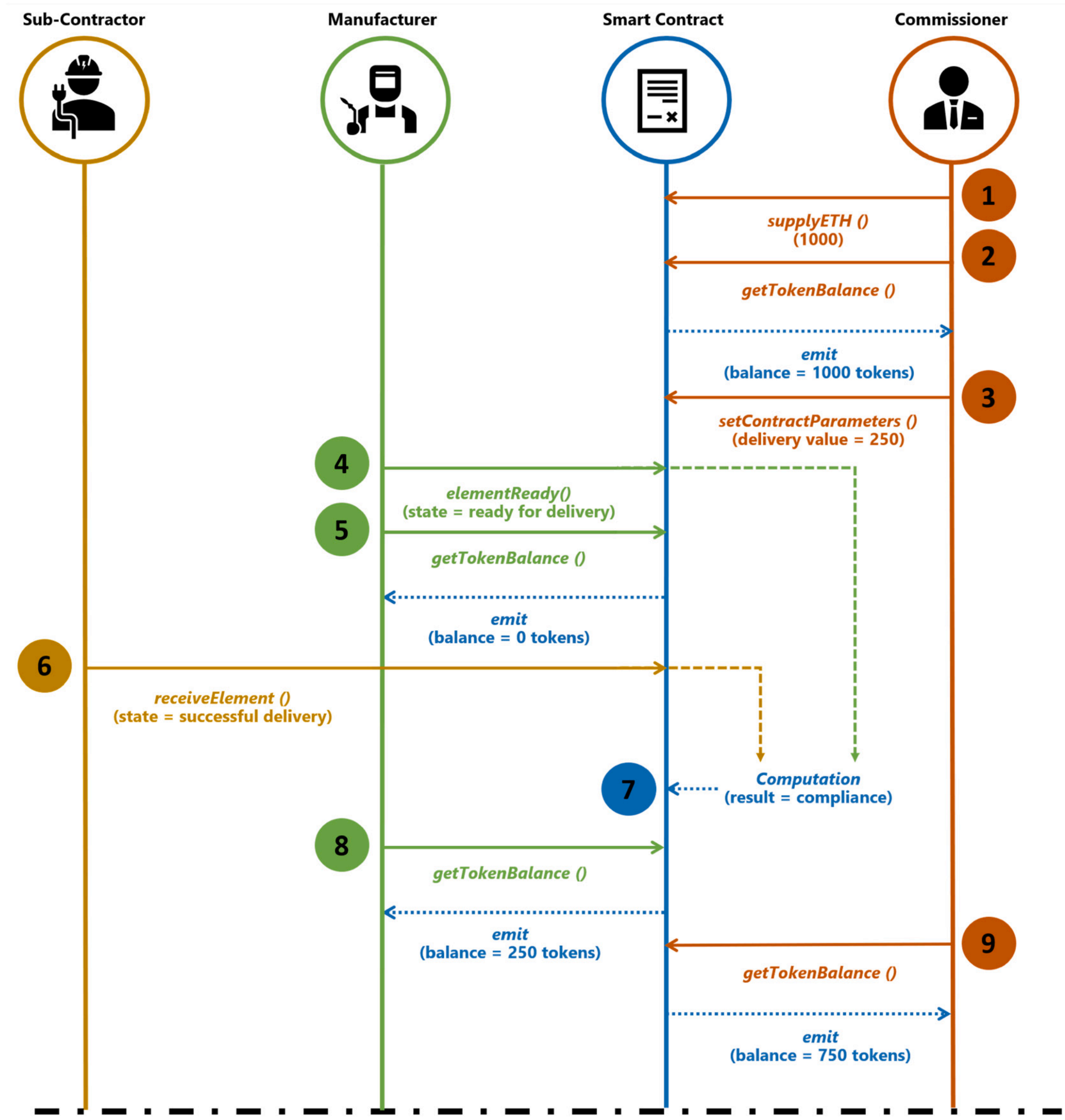

Fig. 8. Smart contract interactions in use cases.

2. The balance of the client EOA is checked to verify whether the correct number of tokens is deposited into the smart contract
3. Contract parameters are inserted into the smart contracts using the client EOA
4. A manufacturing stakeholder communicates completion of a task by updating an assets state in the PAT application
5. The token balance of the manufacturing stakeholder is 0 because delivery is not completed yet
6. An asset is transported to the construction site after which the asset state is updated by a verifying stakeholder
7. Based on inserted contract parameters and the actual state of an asset, the smart contract communicates compliance or non-compliance
8. If compliance is identified, the manufacturing stakeholder is rewarded with the appropriate number of tokens. If non-compliance is identified, such a token transfer is not made
9. The rewarded tokens are deducted from the clients balance in the smart contract until all production steps are completed

Finally the Firebase records are uploaded on the dashboard WebApp to review the dashboard's functionality in each scenario.

4.3. Results and validation

Based on the described workflow, four tests each related to an individual scenario are executed. For each test, the results are compared to the stated requirements to check whether they are met in each scenario. The execution of the executed tests in all environments, their results, suggestions for optimization and a brief reflection are provided hereafter.

4.3.1. Framework

A centralized BIM model and cloud database is used in the tested scenarios for contract parameters and asset tracking data. However, this is not appropriate if these models or databases are not distributed. Furthermore, an admin is included within the proposed framework, commissioned with the development, deployment and execution of smart contracts. This admin also monitors data validity when data is transferred between applications. This admin has to be replaced in the future by the involved stakeholders themselves when appropriate Decentralized Application (DApps) have been developed. In this instance, all data is incorporated into the blockchain and data validity is ensured by its embedded mechanisms. Data acquisition and for example quality assurance procedures will however be difficult if not impossible to decentralize.

4.3.2. Digital asset management

In the tested framework, the presence of an "as-planned" LOMD 400 BIM model containing all required supply chain data is used. However, a BIM model that contains this data in practice is not identified. Therefore, the presumed conditions in terms of DAM do not fully align with the situation in practice. Yet, differences are slight and therefore are expected to be surmountable in practice.

To enable the combination of DAM with PAT and DLT, required contract parameters need to be imported into the smart contract. To test such a workflow, these parameters first need to be processed into Revit. In order to do so, a model containing four elements (one for each scenario) is created. Next, the required data for each phase is manually processed in Revit which include a price, a deadline date and the blockchain adresses of both parties. Because these parameters need to be exported to the online database and thus, need to be converted to a readable format, a schedule export is converted into a JSON file and uploaded to Firebase. By using the Provable Oracle Service, the required parameters to set contract obligations are retrievable by smart contracts in the Remix Environment. For each of the proposed scenarios, the described workflow was followed and parameters were processed accordingly as depicted in Fig. 9.

To increase the efficiency of this process, contract parameters can be directly inserted into Firebase manually. Furthermore, a plugin in BIM software or a DLT application (DApp) for the same purpose increase efficiency as well. The export of data from a BIM model in JSON format (e.g. IFCJSON) equally benefit efficiency and relate best to current built environment practice. JSON is used in this case because most, if not all, current web developments rely on this format. This includes all Blockchain-related development environments that we use.

4.3.3. Physical asset tracking

Besides the input of contract obligations into the smart contract, compliance needs to be communicated as well. To test the use of the developed PAT application (QR), first a QR code containing the asset's GUID is generated for the four fictive elements (one for each scenario). To simulate each of the aforementioned scenarios, these QR code are scanned and the appropriate status is selected. By doing so, a online log of activities is automatically created in the Firebase database as depicted in Fig. 10. Similarly to the contract obligations, the required parameters for smart contract execution are thus retrievable in the Remix environment by means of the Oracle Service as presented in Fig. 11.

After execution of all scenarios with the developed PAT application (QR) and demonstration to the industry partner, the efficiency of data acquisition and communication of assets states between parties is observed to be improved compared to current practice. Furthermore, the execution of this workflow results in centralization of data in a cloud database and enables the use of this data for smart contract execution.

Fig. 9. Firebase obligation log.

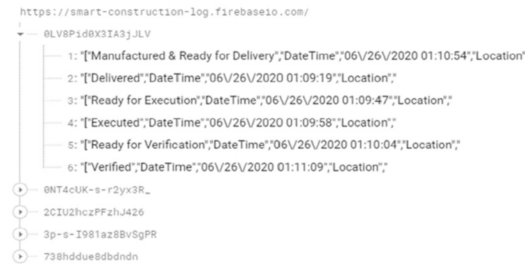

Fig. 10. Firebase activity log.

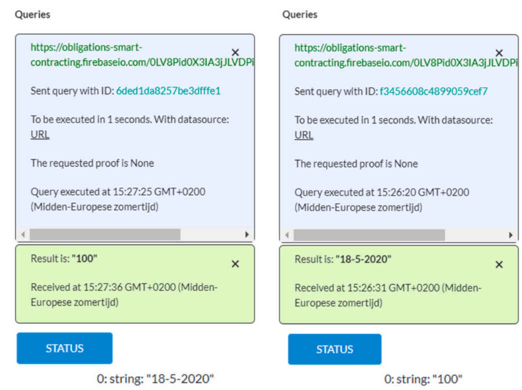

Fig. 11. Oracle query in smart contract.

Therefore, the migration of data to the Remix environment using the developed applications was successfully demonstrated. Further optimization of the developed PAT application can increment these observed positive results. To identify aspects which are suitable for optimization, potential limitations, applicability of the application in practice as well as the validity of yielded results, further verification in practice has to be sought. This can be done by extending the use of the PAT application throughout construction projects.

4.3.4. Ethereum distributed ledger

The generation and import of required parameters for smart contract execution is successfully demonstrated in the Remix environment. Because both the obligation parameters (as-planned) and the compliance parameters (as-built) are communicated to the smart contract, a transaction can be executed. For each scenario, the appropriate transactions are executed in the Remix environment which should either result in the transfer of tokens between parties or an error message depending on the executed scenario. After execution of all scenarios, the smart contract feedback in Remix is checked to identify whether tokens were transferred or retained. An example of a successful transaction is provided in Fig. 12.

Observation of the transaction feedback demonstrated that the smart contact responded as expected in each scenario. This is also reflected by checking the token balance for each fictive stakeholder after a transaction is executed. Because the used Remix environment is coupled to the Gërli Testnet, all transactions with the smart contracts for each scenario are recorded on an instance of the Ethereum Blockchain.[12] Because these transactions are recorded on a blockchain, the use of Etherscan enables us to review the transaction log and inspect transactions as depicted in Fig. 13.

Although semi-automation is achieved, several other components

[12] https://goerli.etherscan.io/address/0x62237076519c70133ac434ab76d362f5bbc997b8

```
"from": "0x4a52959d833Dcc461c2408A277d831Bb6000749B",
"topic": "0x26eb4a415baf730c62c6011b03653f5b8960baf7ad391bd18725e6caa4bfa4c9",
"event": "Payment",
"args": {
        "0": "Payment sent",
        "1": "0x813620c3b12264FF46D88a3F9F4911F2Dc937d7a",
        "2": "0x64eb7a2759098976917934 2f0F7dD1845f84F711",
        "3": "200",
        "_message": "Payment sent",
        "_from": "0x813620c3b12264FF46D88a3F9F4911F2Dc937d7a",
        "_to": "0x64eb7a27590989769179342f0F7dD1845f84F711",
        "_amount": "200",
        "length": 4
}


"from": "0x4a52959d833Dcc461c2408A277d831Bb6000749B",
"topic": "0x953dec42c76a036f9a102dc78d7144cfa7a2c7ef58220ef12da6c75fa2fcfac0",
"event": "Success",
"args": {
        "0": "Element executed",
        "1": "3P$I981AZ8BVSGPRCT9GZT",
        "2": "6",
        "3": "1593046344",
        "4": "0xebA00DEb7bCe7A1Ff24A97E7620d8948408EC8AA",
        "_message": "Element executed",
        "GUID": "3P$I981AZ8BVSGPRCT9GZT",
        "_locationData": "6",
        "_timeStamp": "1593046344",
        "_sender": "0xebA00DEb7bCe7A1Ff24A97E7620d8948408EC8AA",
        "length": 5
}
```


Fig. 12. Execution of successfully transaction in remix.

and actions to trigger smart contract execution are identified to be suitable for semi-automation in the future. During test execution several input values are manually provided in the Ethereum Remix environment to increase flexibility. These interactions can be partially automated by scheduling them in the future with the Ethereum Alarm Clock. The application of a consortium-based and permission-oriented network model with a PoA consensus algorithm proves to be most suitable for built environment collaborations. Instead of using a permissioned blockchain, smart contracts are deployed on the Goërli test net which is an open permissionless network. The results which are yielded from tests on the Goërli testnet, are not expected to change when tested on the preferred customized blockchain network.

4.3.5. *Dashboard webapp*

Although the results of smart contract execution for individual contracts are eligible for review on Etherscan, the interface and workflow to do so is perceived to be unfriendly to users. To provide potential stakeholders with a comprehensible overview of all smart contract results, a dashboard is preferred. On such a dashboard, a log of asset states and token transactions needs to be displayed. To enable simulation of dashboard functionalities in the future, a dashboard WebApp is developed and deployed on an online domain. Because the developed dashboard depends on the manual import of JSON files, these files are exported from Firebase for each scenario. Next, these files are imported into the dashboard's interface after which the information presented on the dashboard is compared to the transaction history on the blockchain. The executed test with the dashboard environment indicate that the dashboard can cope with both successful and unsuccessful transactions. An example of the populated dashboard after execution of the first scenario is presented in Fig. 14. Optimizations can focus on: (1) the establishment of a direct link with the Firebase databases, (2) inclusion of additional charts and visualisations required by practice, (3) development of multiple project environments, (4) stakeholder-specific dashboard instances and (5) a direct link with blockchain data.

4.3.6. *Reflection*

After all tests are executed, the results are validated against each of the stated requirements. An overview of the results of this validation are presented hereafter.

1 Successful migration of data into and from smart contracts was demonstrated by transacting obligations/asset states into the Remix environment and by generation of an overview in the dashboard WebApp respectively.
2 The functionalities and resilience of the developed applications were successfully demonstrated in multiple scenarios developed in collaboration with an industry partner.

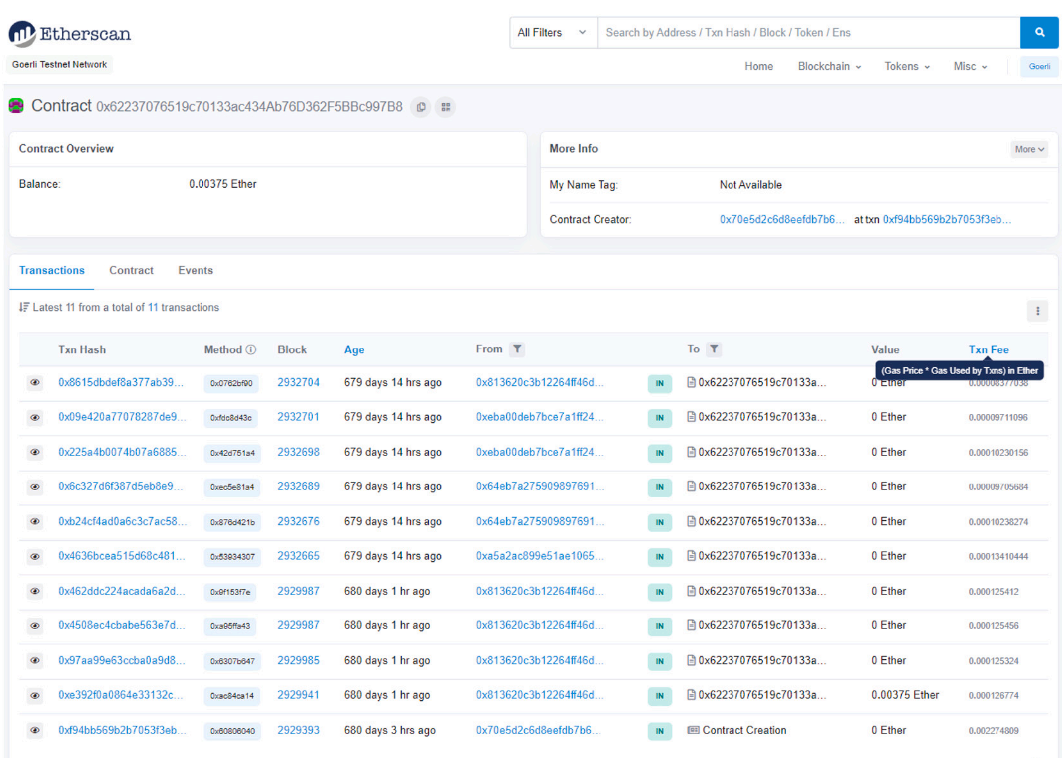

Fig. 13. Smart contract transaction log on ethereum blockchain.

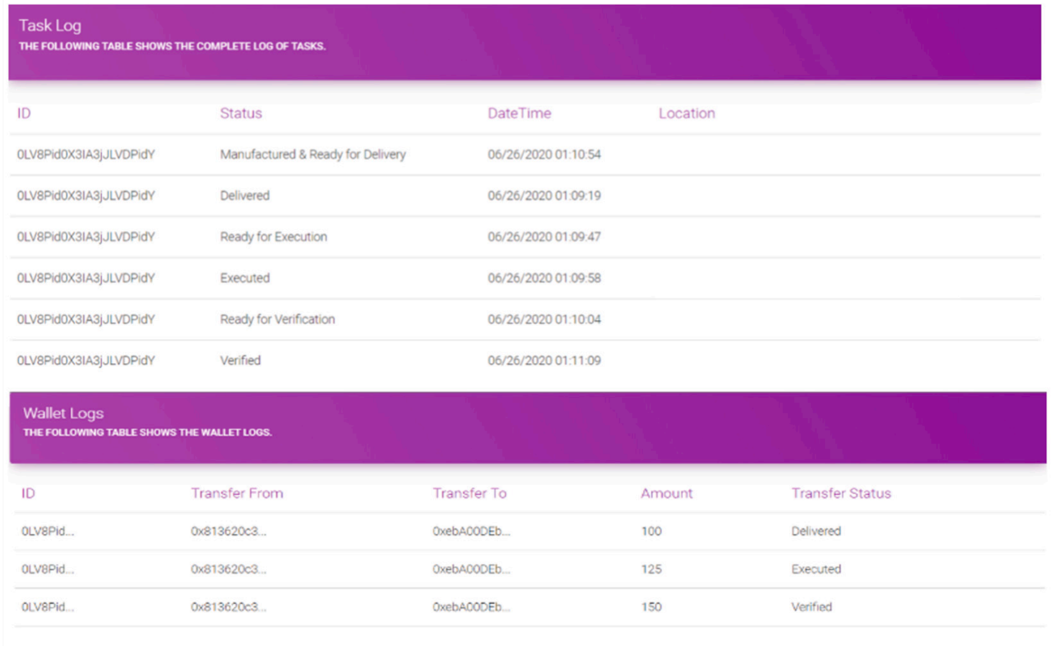

Fig. 14. Snapshot of populated dashboard webapp page.

3 An immutable log of transactions and payments was registered on the Göerli Ethereum testnet using the developed applications.
4 Tokens were distributed to the correct stakeholders in each scenario with limited human interference.

These positive tests show that the proposed framework is successful and promising. Nevertheless, our validation and results are currently on a small scale, and further tests are needed on a larger scale and within a commercial context to further evaluate the feasibility of our framework to track prefabricated assets and compliance with the used technologies. Important caveats for those future tests are included in the conclusion and discussion below.

5. Conclusion and discussion

Based on the constructed framework, developed prototypes and executed tests, semi-automation of planning compliance checking, payments and notifications is achieved. The developed PAT application provides physical states whereas contract parameters are provided by means of a hypothetical jointly-managed BIM model. Smart contracts enable a semi-automated comparison of the “as-planned” and “as-built” state and transfers tokens based on the results. Furthermore, notifications which relate to compliance or non-compliance are successfully communicated to supply chain stakeholders in the Remix environment and on the dashboard WebApp. A log of all smart contract interactions is retained in a distributed ledger. Hence, (1) semi-automated compliance tracking, (2) subsequent (token) payments and (3) an immutable/transparent log are realized and tested in practical use cases.

Although there is still space for optimization, more insight is obtained into the joint deployment of asset tracking techniques throughout the supply chain. If developed appropriately, the deployment of an optimized version of the proposed concept can increase transparency. Increased transparency might gradually enable the development of trust between stakeholders. We do of course acknowledge that there are plenty of other non-technological factors that impact trust and transparency. However we believe that the developed infrastructure contributes to a reduction of risks and thus, conflicts between stakeholders.

5.1. A network of components

Our proposal and results provide a combination of components. For each of those components, a few conclusions can be made.

5.1.1. Framework

Because developed applications depend on data storage on centralized servers, this counteracts the distributed nature of DLT. The development of DApps which run on the blockchain themselves, are key to fully capitalize on the benefits DLT. While we hope that the development and application of DApps in the built environment will become possible in the future, the use of currently available centralized applications is required to make practical advancements.

5.1.2. Asset tracking

Because we focus on PAT for prefabricated elements, geospatial technologies (QR-codes) are considered to be most suitable as they allow for on and off-site tracking. The application of geospatial technologies for collective asset tracking throughout the supply chain has several advantages in comparison to manual asset tracking. These advantages include the increased efficiency of data collection, communication and centralized storage. However, these potential advantages are especially realized due to the collective effort of asset tracking activities by collaborating stakeholders.

In order to track all physical assets in the future, geospatial technologies have to be combined with scanning technologies (e.g. point cloud), to enable efficient on-site progress monitoring. Scanning technologies have to be mainly considered to capture the on-site progress of smaller construction elements (e.g. brickwork) and elements which are manufactured on site (casted concrete). Hence, combining both technologies is required in the future. Ideally, the semi-automated transfer of sensor data to distributed databases or the distributed ledger itself has to be realized.

5.1.3. Asset management

BIM models are not the only mean to store the required contract parameters. Alternatives such as distributed databases, WebApps and linked Enterprise Resource Planning (ERP) systems are available and might have significant benefits over BIM models. Regardless, sharing sensitive data (e.g. cost data) might incur resistance in industry due to the exposure of price structures between stakeholders. Furthermore, centralization of contract parameters counteracts the distributed nature of DLT's. To enable efficient management and processing of contract parameters (planning and costs) in the DLT environment, storing this data in a distributed information system is preferred. Further distribution of such data and its utilization on a DLT has to be further investigated.

5.1.4. Distributed ledger

In previous research, it is often suggested that distributed ledgers provide transparency and thus trust by means of immutability, security and re-distribution of power. Hence, the technology is identified as a mean to reduce risks and potential conflicts between supply chain stakeholders. Furthermore, smart contracts are used to enable semi-automation of planning compliance checking and payment procedures. In this research, we focus on the application of an Ethereum-based blockchain network, due to the related smart contract technology. Although not all types of blockchain networks enable the utilization of smart contract technologies, less prominent alternatives for the Ethereum blockchain are available (e.g. Hyperledger).

5.1.5. Smart contracts

Optimizations of the developed smart contracts can focus on: (1) optimization of value inputs, (2) a reduction of transaction costs and (3) linking the blockchain to an external decentralized application (to set contract parameters and track assets), instead of using various interlinked applications. Addressing point (1), the optimization of value inputs into smart contracts, mechanisms such as the Ethereum Alarm Clock can be used. Because human intervention is desirable to ensure the validity of provided data, the trade-offs of employing such a mechanism have to be investigated.

Regarding point (2), reduction of transaction costs, each smart contract execution or transaction on the blockchain requires computational power. The magnitude of required computational power depends on the code-efficiency in a smart contract and results in transactions costs. These transaction costs are expressed on the Ethereum blockchain as a gas fee. The costs for a unit of gas is expressed in GWEI representing a value of 0.000000001 Ethereum.[13] Because transactions were executed on a Ethereum testnet in this research, these transactions currently do not require any payment. When our smart contracts are to be deployed on the main net however, these transaction costs have to be paid. Therefore, it is important to provide insight into the costs for smart contract execution and an estimation of these transaction costs in construction projects. Because transactions were executed in the Ethereum test environment, the transaction costs for contract execution were recorded in an online log.[14] Using this record and considering the exchange price of 1 ETH which is 2699,08 *euro* on 02/05/2022,[15] transaction costs can be determined. For contract creation these costs are 6,14 *euro* and the average transaction cost per interaction with the smart contract are 0,30 *euro*. Because the smart contract is used throughout the states the related physical assets pass through, the total transaction costs including contract creation are 9,16 *euro*. Considering that a large construction project contains at least a thousand prefabricated assets and assuming the proposed workflow is limited to these assets, the total transaction costs for a project are estimated to be 9160 *euro*. Due to the high price of these large assets, the transaction cost do not seem to be disproportionate. If and when smart contract utilization are not limited to prefabricated assets but also used for smaller components however, the number of physical assets and thus smart contracts would increase significantly (9,16 *euro* per contract). Furthermore, if the number of transactions increase (e.g. due to defects), these costs are expected to rise (0,30 *euro* per transaction). In these cases, it is easy to imagine that the size of transaction costs has to be addressed to increase the concept's feasibility. This can be partially done by increasing smart contract efficiency and partially has to be realized by merging or batching multiple physical assets into a single smart contract.

Regarding point (3), distribution can only be achieved if the entire infrastructure is decentralized, future research efforts have to focus on the development of Decentralized Applications (DApps). As these types of application essentially run on a blockchain network, distributed data storage is also realized. This further benefits the proposed concept and result in more distribution, addressing point (3): linking the blockchain to an external decentralized application. The implementation of a monitoring mechanism (which does not depend on the admin), is required to ensure data validity during intermediate steps in future concepts. It does however remain to be debated if decentralization is practically achievable as decentralization of power in the industry is difficult to achieve. To fully benefit from the distributed nature of blockchain networks, equal distribution of power within blockchain networks needs to be established. Additionally, equal distribution of power among the collaborating stakeholders outside of the blockchain network and decentralization of all used digital applications is required as well. Although the use of blockchain and smart contracts can aid towards the purpose of more distribution of power, a large number of cultural changes and other developments are required.

5.1.6. Dashboard

The dashboard WebApp is developed to provide insight into PAT data, contract parameters and smart contract computation results (e.g. token balance of stakeholders). Although ideally such a dashboard is directly linked to blockchain data, establishment of such a link on a blockchain test net was not possible. If the developed smart contracts are deployed on a customized main net instance, several dashboard services can be used. However, the functionalities of these services are limited due to unresolved technical issues. Deployment of the developed smart contracts on the main net is not realized within the scope of this work.

5.2. Contribution

Previous research on DLT, smart contracts and QR codes for asset/compliance tracking mainly focuses on on-site construction activities. The presented research also provides more insight into the combined application of these concepts throughout full construction supply chains. Previous work has contributed significantly to the development of frameworks and theories which are required for the application of these technologies in practice. The application of these individual and combined technologies in actual industry use cases however, is identified to be lacking. The developed framework and prototypes which are presented and tested in this article, provide us with more insight into the actual application of such a concept in industry. Furthermore, the development of an immutable construction log to increase transparency and trust has been widely covered in literature. The proposed and tested concept includes a prototype of such a construction log, which is tested in industry scenarios.

Overall, these insights aid towards the optimization and semi-automation of: (1) prefabricated asset tracking, (2) compliance checking and (3) token transfers (payments). The developed prototypes enable the combination of various technologies in a supply chain context, which can be considered as an essential basic concept for the development of sustainable collaborations. Such a concept can be advanced towards an operational and integrated distributed system. This system enables industry stakeholders to jointly capitalize on the advantages that ledgers offer: (1) distributed, (2) immutable, (3) secure and (4) transparent. Although its effects on built environment collaborations have to be empirically verified, the developed concept provides a

[13] https://ethereum.org/en/developers/docs/gas/

[14] https://goerli.etherscan.io/address/0x62237076519c70133ac434ab76d362f5bbc997b8

[15] https://www.coinbase.com/price/ethereum

basis for increasing trust and reducing conflicts.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors of this article like to express their sincere thanks to the Royal VolkerWessels company in the Netherlands for their participation in the expert panel, provision of practical data to construct the required use cases and verification of results of test-case execution.

References

[1] S. Nanayakkara, S. Perera, S. Senaratne, Stakeholders' perspective on blockchain and smart contracts solutions for construction supply chains, in: Proceedings of the CIB World Building Congress 2019: Constructing Smart Cities, The Hong Kong Polytechnic University, Hong Kong, China, 2019, pp. 1–11, https://doi.org/10.6084/m9.figshare.8868386.v1.

[2] A. Ashworth, S. Perera, Contractual Procedures in the Construction Industry, Taylor & Francis, Andover, England UK, 2018, https://doi.org/10.1201/9781315529059.

[3] R. Vrijhoef, L. Koskela, The four roles of supply chain management in construction, Eur. J. Purchase. Supply Manage. 6 (3–4) (2000) 169–178, https://doi.org/10.1016/S0969-7012(00)00013-7.

[4] M.N.A. Azman, N. Dzulkalnine, Z. Abd Hamid, K.W. Beng, Payment issue in Malaysian construction industry: contractors' perspective, J. Technol. 70 (1) (2014) 57–63, https://doi.org/10.11113/jt.v70.2804.

[5] P. Behera, R.P. Mohanty, A. Prakash, Understanding construction supply chain management, Prod. Plan. Control 26 (16) (2015) 1332–1350, https://doi.org/10.1080/09537287.2015.1045953.

[6] J. Mason, Intelligent contracts and the construction industry, J. Legal Affair. Dispute Resolut. Eng. Constr. 9 (3) (2017) 04517012, https://doi.org/10.1061/(ASCE)LA.1943-4170.0000233.

[7] A. Shojaei, I. Flood, H. Izadi Moud, M. Hatami, X. Zhang, An implementation of smart contracts by integrating bim and blockchain, in: Proceedings of the Future Technologies Conference, 2019, pp. 519–527, https://doi.org/10.1007/978-3-030-32523-7-36.

[8] A. Heiskanen, The technology of trust: how the internet of things and blockchain could usher in a new era of construction productivity, Constr. Res. Innovat. 8 (2) (2017) 66–70, https://doi.org/10.1080/20450249.2017.1337349.

[9] S. El-Omari, O. Moselhi, Integrating 3D laser scanning and photogrammetry for progress measurement of construction work, Automat. Constr. 18 (1) (2008) 1–9, https://doi.org/10.1016/j.autcon.2008.05.006.

[10] J. Wang, P. Wu, X. Wang, W. Shou, The outlook of blockchain technology for construction engineering management, Front. Eng. Manage. 4 (1) (2017) 67–75, https://doi.org/10.15302/J-FEM-2017006.

[11] J. Li, M. Kassem, A. Ciribini, M. Bolpagni, A proposed approach integrating dlt, bim, iot and smart contracts: demonstration using a simulated installation task, in: International Conference on Smart Infrastructure and Construction 2019 (ICSIC), 2019, pp. 275–282, https://doi.org/10.1680/icsic.64669.275.

[12] M. Kopsida, I. Brilakis, P.A. Vela, A review of automated construction progress monitoring and inspection methods, in: Proceedings of the 32nd CIB W78 Conference (2015), 2015, pp. 421–431, https://itc.scix.net/paper/w78-2015-paper-044.

[13] B. Hubbard, H. Wang, M. Leasure, T. Ropp, T. Lofton, S. Hubbard, S. Lin, Feasibility study of UAV use for RFID material tracking on construction sites, in: 51st ASC Annual international conference proceedings, 2015, pp. 669–676.

[14] A. Lanko, N. Vatin, A. Kaklauskas, Application of rfid combined with blockchain technology in logistics of construction materials, in: MATEC Web Conferences Vol. 170, 2018, p. 03032, https://doi.org/10.1051/matecconf/201817003032.

[15] A. Guillen, A. Crespo, J. Gómez, V. González-Prida, K. Kobbacy, S. Shariff, Building information modeling as assest management tool, IFAC-PapersOnline 49 (28) (2016) 191–196, https://doi.org/10.1016/j.ifacol.2016.11.033, in 3rd IFAWorkshop on Advanced Maintenance Engineering Services and Technology AMEST.

[16] M. Grieves, Digital twin: manufacturing excellence through virtual factory replication, White Paper 1 (2014) 1–7, https://www.researchgate.net/publication/275211047.

[17] J. Li, D. Greenwood, M. Kassem, Blockchain in the built environment and construction industry: a systematic review, conceptual models and practical use cases, Automat. Constr. 102 (2019) 288–307, https://doi.org/10.1016/j.autcon.2019.02.005.

[18] Turk F.Z., R. Klinc, Potentials of blockchain technology for construction management, in: Creative Construction Conference 2017, Primosten, Croatia, 2017, pp. 638–645, https://doi.org/10.1016/j.proeng.2017.08.052. CC.

[19] N.O. Nawari, S. Ravindran, Blockchain and the built environment: potentials and limitations, J. Build. Eng. 25 (2019) 100832, https://doi.org/10.1016/j.jobe.2019.100832.

[20] Z. Wang, T. Wang, H. Hu, J. Gong, X. Ren, Q. Xiao, Blockchain-based framework for improving supply chain traceability and information sharing in precast construction, Automat. Constr. 111 (2020) 103063, https://doi.org/10.1016/j.autcon.2019.103063.

[21] H. Hamledari, M. Fischer, Construction payment automation using blockchain-enabled smart contracts and robotic reality capture technologies, Automat. Constr. 132 (2021) 103926, https://doi.org/10.1016/j.autcon.2021.103926.

[22] H.R. Hasan, K. Salah, Proof of delivery of digital assets using blockchain and smart contracts, IEEE Access 6 (2018) 65439–65448, https://doi.org/10.1109/ACCESS.2018.2876971.

[23] A.A. Hijazi, S. Perera, R.N. Calheiros, A. Alashwal, Rationale for the integration of bim and blockchain for the construction supply chain data delivery: a systematic literature review and validation through focus group, J. Constr. Eng. Manage. 147 (10) (2021) 03121005, https://doi.org/10.1061/(ASCE)CO.1943-7862.0002142.

[24] I. Lokshina, M. Greguš, W.L. Thomas, Application of integrated building information modeling, iot and blockchain technologies in system design of a smart building, Proc. Comput. Sci. 160 (2019) 497–502, https://doi.org/10.1016/j.procs.2019.11.058, the 10th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2019).

[25] I. Belle, The architecture, engineering and construction industry and blockchain technology, in: G. Ji, Z. Tong (Eds.), Digital Culture - Proceedings of 2017 National Conference on Digital Technologies in Architectural Education and DADA 2017 International Conference on Digital Architecture, China, Architecture Industry Publishers, Nanjing: China, 2017, pp. 279–284, https://www.researchgate.net/publication/322468019.

[26] ISO, Asset Management – Overview, Principles and Terminology, Standard, International Organization for Standardization, Geneva Switzerland, 2014 https://www.iso.org/standard/55088.html.

[27] N. Anthony, J. Hastings, Physical Asset Management: With an Introduction to the ISO 55000 Series of Standards, Springer: Cham, Switzerland, 2015, https://doi.org/10.1007/978-3-030-62836-9.

[28] Z. Ma, L. Zhou, W. Sheng, Analysis of the new asset management standard iso 55000 and pas 55, in: 2014 China International Conference on Electricity Distribution (CICED), 2014, pp. 1668–1674, https://doi.org/10.1109/CICED.2014.6991990.

[29] H. Braaksma, Smart Asset Management in the Construction Sector: A Holistic Research into Utilising the Potential of Sensor Technology, Master's Thesis, Delft University of Technology, Delft, 2016 https://repository.tudelft.nl/islandora/object/uuid:af074ec5-b8c7-4a30-9f16-e3ac7ff5e900.

[30] M. Sakikhales, S. Stravoravdis, Using agile project management and bim for improved building performance, in: Building Information Modelling, Building Performance, Design and Smart Construction, Springer: Cham, Switzerland, 2017, pp. 65–78, https://doi.org/10.1007/978-3-319-50346-2-5.

[31] Q. Lu, X. Xie, J. Heaton, A.K. Parlikad, J. Schooling, From bim towards digital twin: Strategy and future development for smart asset management, in: International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing, Springer: Cham, Switzerland, 2020, pp. 392–404, https://doi.org/10.1007/978-3-030-27477-1-30.

[32] C. Boje, A. Guerriero, S. Kubicki, Y. Rezgui, Towards a semantic construction digital twin: directions for future research, Automat. Constr. 114 (2020) 103–179, https://doi.org/10.1016/j.autcon.2020.103179.

[33] R. Zeibak-Shini, R. Sacks, L. Ma, S. Filin, Towards generation of as-damaged bim models using laser-scanning and as-built bim: first estimate of as-damaged locations of reinforced concrete frame members in masonry infill structures, Adv. Eng. Informat. 30 (3) (2016) 312–326, https://doi.org/10.1016/j.aei.2016.04.001.

[34] Z. Pucko, D. Vincek, A. Štrukelj, N. Šuman, Application of 6d building information model (6d bim) for business-storage building in slovenia, in: IOP Conference Series: Materials Science and Engineering Vol. 245, 2017, p. 062028, https://doi.org/10.1088/1757-899X/245/6/062028.

[35] Y. Fang, Y.K. Cho, S. Zhang, E. Perez, Case study of bim and cloud - enabled real-time rfid indoor localization for construction management applications, J. Constr. Eng. Manage. 142 (7) (2016) 05016003, https://doi.org/10.1061/(ASCE)CO.1943-7862.0001125.

[36] Y. Turkan, F. Bosche, C.T. Haas, R. Haas, Automated progress tracking using 4d schedule and 3d sensing technologies, Automat. Constr. 22 (2012) 414–421, https://doi.org/10.1016/j.autcon.2011.10.003.

[37] J. Mason, Bim fork: are smart contracts in construction more likely to prosper with or without bim? J. Legal Affair. Dispute Resolut. Eng. Constr. 11 (4) (2019) 02519002, https://doi.org/10.1061/(ASCE)LA.1943-4170.0000316.

[38] J. Gong, C.H. Caldas, Data processing for real-time construction site spatial modeling, Automat. Constr. 17 (5) (2008) 526–535, https://doi.org/10.1016/j.autcon.2007.09.002.

[39] T. Cheng, M. Venugopal, J. Teizer, P. Vela, Performance evaluation of ultra wideband technology for construction resource location tracking in harsh environments, Automat. Constr. 20 (8) (2011) 1173–1184, https://doi.org/10.1016/j.autcon.2011.05.001.

[40] W. Lu, G.Q. Huang, H. Li, Scenarios for applying rfid technology in construction project management, Automat. Constr. 20 (2) (2011) 101–106, https://doi.org/10.1016/j.autcon.2010.09.007.

[41] S. Yuvaraj, M. Sangeetha, Smart supply chain management using internet of things (iot) and low power wireless communication systems, in: 2016 International Conference on Wireless Communications, Signal Processing and Networking

(WiSPNET), 2016, pp. 555–558, https://doi.org/10.1109/WiSPNET.2016.7566196.
[42] C. Scardovi, Restructuring and Innovation in Banking, Springer, Cham, Switzerland, 2016, https://doi.org/10.1007/978-3-319-40204-8.
[43] J. Li, D. Greenwood, M. Kassem, Blockchain in the construction sector: a socio-technical systems framework for the construction industry, in: Advances in Informatics and Computing in Civil and Construction Engineering, Springer International Publishing, Cham, Switzerland, 2019, pp. 51–57, https://doi.org/10.1007/978-3-030-00220-6-7.
[44] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, Decentralized Business Review, Seoul, 2019. https://www.debr.io/article/21260-bitcoin-a-peer-to-peer-electronic-cash-system.
[45] N. Szabo, Formalizing and Securing Relationships on Public Networks 2, First Monday, Chicago, 2012, https://doi.org/10.5210/fm.v2i9.548.
[46] G. Peters, E. Panayi, A. Chapelle, Trends in cryptocurrencies and blockchain technologies: a monetary theory and regulation perspective, J. Finan. Perspect. 3 (3) (2021) https://ssrn.com/abstract=3084011.
[47] R. Herian, Smart contracts: a remedial analysis, Informat. Commun. Technol. Law 30 (1) (2021) 17–34, https://doi.org/10.1080/13600834.2020.1807134.
[48] H. Luo, M. Das, J. Wang, J. Cheng, Construction payment automation through smart contract-based blockchain framework, in: Proceedings of the 36th International Symposium on Automation and Robotics in Construction, ISARC 2019 International Association for Automation and Robotics in Construction I.A.A.R.C, 2019, pp. 1254–1260, https://doi.org/10.22260/isarc2019/0168.
[49] A.M. Antonopoulos, G. Wood, Mastering Ethereum, O'Reilly Media, Inc., Sebastopol, CA, USA, 2018 https://www.oreilly.com/library/view/mastering-ethereum/9781491971932.
[50] S. Ferretti, G. D'Angelo, On the ethereum blockchain structure: a complex networks theory perspective, Concurrency Computat. Practice Experience 32 (12) (2020) 1–12, https://doi.org/10.1002/cpe.5493.
[51] G. Wood, Ethereum: a secure decentralised generalised transaction ledger, Ethereum Project Yellow Paper 151 (2014) 1–32, http://gavwood.com/Paper.pdf.
[52] A. Baliga, Understanding Blockchain Consensus Model 4, Persistent Ltd., Santa Clara, 2017 https://www.persistent.com/wp-content/uploads/2017/04/WP-Understanding-Blockchain-Consensus-Models.pdf.
[53] M. Crosby, P. Pattanayak, S. Verma, V. Kalyanaraman, Blockchain technology: beyond bitcoin, Appl. Innovat. 2 (6–10) (2016) 71, https://j2-capital.com/wp-content/uploads/2017/11/AIR-2016-Blockchain.pdf.
[54] D. Kifokeris, C. Koch, A conceptual digital business model for construction logistics consultants, featuring a sociomaterial blockchain solution for integrated economic, material and information flows, J. Informat. Technol. Constr. 25 (2020) 500–521, https://doi.org/10.36680/J.ITCON.2020.029.
[55] G. Hileman, M. Rauchs, 2017 global blockchain benchmarking study, SSRN Electr. J. 1 (2018) 1–115, https://doi.org/10.2139/ssrn.3040224.
[56] V. Acharya, A.E. Yerrapati, N. Prakash, Oracle Blockchain Quick Start Guide: A Practical Approach to Implementing Blockchain in your Enterprise, Packt Publishing Ltd, Birmingham, England, UK, 2019 https://subscription.packtpub.com/product/big-data-and-business-intelligence/9781789804164.
[57] D. Lee, S.H. Lee, N. Masoud, M. Krishnan, V.C. Li, Integrated digital twin and blockchain framework to support accountable information sharing in construction projects, Automat. Constr. 127 (2021) 103688, https://doi.org/10.1016/j.autcon.2021.103688.
[58] S. Huang, G. Wang, Y. Yan, X. Fang, Blockchain-based data management for digital twin of product, J. Manufact. Syst. 54 (2020) 361–371, https://doi.org/10.1016/j.jmsy.2020.01.009.
[59] X. Ye, M. König, Framework for automated billing in the construction industry using bim and smart contracts, in: International Conference on Computing in Civil and Building Engineering, 2021, pp. 824–838, https://doi.org/10.1007/978-3-030-51295-8-57.
[60] H. Hamledari, M. Fischer, The application of blockchain-based crypto assets for integrating the physical and financial supply chains in the construction & engineering industry, Automat. Constr. 127 (2021) 103711, https://doi.org/10.1016/j.autcon.2021.103711.
[61] J. Li, M. Kassem, R. Watson, A Blockchain and smart contract-based framework to inrease traceability of built assets, in: Proc. 37th CIB W78 Information Technology for Construction Conference (CIB W78), Sao Paolo, Brazil, 2020, pp. 347–362, https://doi.org/10.46421/2706-6568.37.
[62] R. Wieringa, Design Science Methodology for Information Systems and Software Engineering, Springer: Cham, Switzerland, 2014, https://doi.org/10.1007/978-3-662-43839-8.
[63] K. Peffers, T. Tuunanen, M.A. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, JMIS 24 (3) (2007) 45–77, https://doi.org/10.2753/MIS0742-1222240302.