

Smart Contract Generation and Visualization for Construction Business Process Collaboration and Automation: Upgraded Workflow Engine

Xuling Ye, S.M.ASCE¹; Ningshuang Zeng, Ph.D.²; Xingyu Tao, Ph.D.³; Daguang Han, Ph.D.⁴; and Markus König, Ph.D.⁵

Abstract: With the digital transformation of the construction industry, the need to improve construction business process collaboration and automation is increasing. However, as construction projects usually involve many stakeholders with complex relationships and insufficient mutual trust, the existing technologies cannot fulfill the requirements of the construction industry. This paper explores the integration of blockchain, smart contracts, and process automation technologies into construction business processes, shedding light on both managerial and technical challenges. The study introduces a comprehensive technical framework meticulously designed to align with the intricate nature of construction practices while focusing on the resolution of legal complexities associated with smart contract applications and the facilitation of process collaboration and automation. The framework comprises three core modules: (1) action definition and extension to process models, (2) standardized mapping for automatic smart contract generation, and (3) smart contract visualization for reliable process collaboration and automation. This multifaceted approach caters to the specific needs of construction management, standardization, interoperability, and visualization. The framework's practicality is further evaluated through real-world testing within a construction payment case, effectively showcasing its efficacy and applicability in tangible business scenarios. While this paper represents a significant step forward in addressing construction business process collaboration and automation challenges, it acknowledges the necessity for ongoing research and development to refine and expand these innovative solutions to meet the evolving demands of construction management. DOI: [10.1061/JCCEE5.CPENG-5938](https://doi.org/10.1061/JCCEE5.CPENG-5938). This work is made available under the terms of the Creative Commons Attribution 4.0 International license, <https://creativecommons.org/licenses/by/4.0/>.

Practical Applications: Currently, there is a significant escalation in the severity of disputes within the construction industry. The adoption of blockchain technology allows for the redistribution of trust within a construction project, shifting reliance from individuals to the system itself for better collaboration and fewer disputes. Although blockchain can help foster a more secure and transparent environment, at the process level, an efficient workflow engine is needed to activate processes in this environment. Therefore, this study proposed an approach to generating smart contract codes based on construction business processes in a standardized way and further visually executing these codes for the processes with historical transactions stored in blockchain. This approach can reduce the effort of smart contract programming and improve the standardization and credibility of smart contract generation. It can also streamline the procedure from construction business process design to final execution via predefined and visual modules. Meanwhile, the generation prototype is open sourced to provide transparency and further enhance the approach. To indicate practical implications in the real world, this study provided a case of construction payment, but the solution is not limited to the payment scenario.

Author keywords: Smart contract; Blockchain; Business process collaboration; Process automation; Workflow engine.

¹Ph.D. Candidate, Dept. of Civil and Environmental Engineering, Ruhr-Univ. Bochum, Universitätsstr. 150, Bochum 44801, Germany. ORCID: <https://orcid.org/0000-0001-9857-4288>. Email: xuling.ye@rub.de

²Postdoctoral Research Fellow, School of Civil Engineering, Southeast Univ., No. 2 Southeast University Rd., Nanjing 211189, China (corresponding author). ORCID: <https://orcid.org/0000-0003-3963-9488>. Email: ningshuang_zeng@seu.edu.cn

³Postdoctoral Research Fellow, Dept. of Civil and Environmental Engineering, Hong Kong Univ. of Science and Technology, Clear Water Bay, Hong Kong, China. Email: turbetao@ust.hk

⁴Associate Professor, School of Civil Engineering, Southeast Univ., No. 2 Southeast University Rd., Nanjing 211189, China. Email: daguanghan@seu.edu.cn

⁵Professor, Dept. of Civil and Environmental Engineering, Ruhr-Univ. Bochum, Universitätsstr. 150, Bochum 44801, Germany. ORCID: <https://orcid.org/0000-0002-2729-7743>. Email: koenig@inf.bi.rub.de

Note. This manuscript was submitted on January 8, 2024; approved on May 6, 2024; published online on July 29, 2024. Discussion period open until December 29, 2024; separate discussions must be submitted for individual papers. This paper is part of the *Journal of Computing in Civil Engineering*, © ASCE, ISSN 0887-3801.

Introduction

Construction projects with various business and management processes typically operate within temporary and unique organizational frameworks (Adriaanse et al. 2010; Turner and Müller 2003). The efficacy of collaborative process execution is significantly undermined in contexts where trust among project stakeholders is deficient or, more egregiously, in the circumstances characterized by adversarial relationships (Emaminejad et al. 2024). For instance, if a contractor intentionally conceals a quality defect from the owner, the owner will pay the cost while concurrently assuming the risk associated with the compromised quality (Sohail and Cavill 2008). Arcadis's 2023 Global Construction Dispute Report indicates a significant escalation in the severity of disputes within the construction industry. Specifically, the average value of these disputes saw a substantial increase of 42% from 2021 to 2022 (Arcadis 2023; Tabish and Jha 2023). Therefore, it is necessary to seek process collaboration models, methods, and solutions in a trust-based environment to reduce disputes and improve efficiency

in construction. Concurrently, the advent of the latest generation of information technology lays a solid foundation for these endeavors, providing innovative tools and platforms to facilitate effective collaboration.

Previous research on strategies aimed at cultivating trust in construction business and management scenarios emphasizes the indispensable role of trust in achieving business success and as a fundamental human value (Chang and Hasanzadeh 2024; Emaminejad et al. 2024; Kochovski et al. 2019; Lee et al. 2022; Li et al. 2023). These efforts have provided a theoretical basis for creating a trust-based environment for collaboration. In recent years, the emergence of disruptive information technology has changed the industrial landscape and provided innovative ways to enhance interorganizational trust (Choi et al. 2022). As one of the representative technologies, blockchain can provide a distributed, secure, traceable, and immutable information environment with its flexibly executed smart contracts, helping build trust and eliminate friction in the construction field (Isaja et al. 2023; Lu et al. 2023a, b).

Within the realm of construction business and management, existing blockchain platforms like Ethereum and Hyperledger have been extensively utilized or custom-developed to serve as foundational support for various functions, including contract payments (Hamledari and Fischer 2021), claims management (Kim et al. 2022), and supplier management (Tezel et al. 2021). Beyond the technical benefits, another crucial factor driving the adoption of blockchain applications lies in the transfer of trust (Qian and Papadonikolaki 2021). More precisely, the implementation of blockchain technology allows for the redistribution of trust within a construction project, shifting reliance from individuals to the system itself. This transition underlines the transformative potential of blockchain to foster a more secure and transparent environment for all project stakeholders. However, coordination at the process level relies on the workflow engine, which is the technical method and standard achievement of long-term research and practice in the business process management (BPM) field (Kozma et al. 2021). Blockchain alone cannot fully support construction business process coordination by providing the environment, but the self-diagnosing and self-enforcement smart contracts can evoke a business process (BP) by triggering the next task when the multipartite conditions are met (Mars et al. 2023) and can act as an upgraded workflow engine, inheriting the traceable, consistent, and undisputed technical advantages of blockchain (Sigalov et al. 2021).

Currently, the smart contract-based workflow engine is just conceptually proposed as a part of a blockchain-enabled solution with low technical details (Jiang et al. 2021), and there are technical and managerial challenges. First, the unclear legal responsibility arises due to the complexity of smart contracts, which are often not precisely understood by practitioners without an information technology (IT) background, coupled with the difficulty for IT specialists in assessing legal requirements (Dwivedi et al. 2023). Second, interoperability issues stem from the complexity of ensuring compatibility between process model-based smart contracts (SC) and various blockchain platforms (Khan et al. 2021). Third, the absence of standardized tools and guidelines for BP models, such as an internationally standardized BP model called business process model and notation (BPMN), for SC generation hinders adoption, emphasizing the need for standardized practices in this field (Milani et al. 2021). The first two challenges significantly impact the third, highlighting the need for a standardized and visually enhanced method to generate and execute SCs, thereby facilitating collaborative consensus and enhancing efficiency.

Therefore, this paper proposes a comprehensive approach to generating smart contract codes based on construction BPs in a

standardized way and further visually executing these codes for the processes with historical transactions stored in blockchain. Three objectives of this research are summarized as:

1. Developing flexible and tailor-made modules for generating smart contracts, including adaptable module creation that can be customized for different aspects of construction BPs, to ensure that the smart contracts are not one-size-fits-all but can be tailored to specific needs and requirements of various construction projects.
2. Proposing unified structuring and automated creation of BP-based smart contracts for enhancing standardization and clearly delineating legal responsibilities between code programming and process design.
3. Implementing graphical and managerial representation of smart contracts for facilitating process collaboration and addressing process disruptions promptly.

The remainder of this paper is organized as follows. Section "Literature Review" presents state-of-the-art approaches and identifies the gaps this study addresses. Section "Research Methodology" describes the methodology. Section "SC-Enabled Construction BP Collaboration and Automation Framework" elaborates the framework with key functional modules and their technical details. Section "Implementation and Evaluation" uses a construction delivery, acceptance, and payment process case to illustrate implementation and provide an interview-based evaluation for validation. Section "Discussion" explores theoretical contributions and practical implications. Finally, the section "Conclusion" summarizes the paper and discusses the future research directions of this study.

Literature Review

Construction BP Collaboration

In the construction field, the need for collaboration stems from the nature of projects that bring together multiple organizations on a temporary basis, each with different interests (Mashali et al. 2023). Emerging technologies like building information modeling (BIM), business process automation (BPA), and artificial intelligence (AI) have streamlined collaboration in construction business processes. As one of the representative research studies, Singh et al. (2011) proposed a theoretical framework of a BIM-based collaboration platform, which triggered a series of explorations about BIM-enabled collaboration at the platform level (Alreshidi et al. 2018; Lai et al. 2019; Zhang et al. 2017). Most BIM-enabled collaboration solutions did not drill down to the process level. Oraee et al. (2019) pointed out that the lack of standards to inform collaboration processes in BIM and the security problem of BIM models being shared on cloud platforms are significant barriers.

There is ongoing development of integrated applications combining BIM with other technologies to improve collaboration, such as solutions that merge BIM and process automation techniques (Pan and Zhang 2021), as well as the integration of BIM with the internet of things (IoT) for collaborative tasks in facility management (Dahanayake and Sumanarathna 2022). However, the underlying problems, as Oraee et al. (2019) proposed, still exist. At the process level, trust issues affect various aspects of the construction BPs, significantly influencing the success of collaborative business objectives. For example, if construction delays occur because of material supply problems, it can lead to disputes among the participants about who is responsible (Qian and Papadonikolaki 2021). Traditional supply chain management faces issues such as tampering with information flow, untraceable logistics, and inaccurate capital flow, leading to increased potential costs due to a lack of trust among transaction entities (Wu and Zhang 2022).

Therefore, additional measures are imperative to resolve trust-related concerns, ultimately augmenting the efficiency of collaborative construction BP. Due to the technical characteristics of blockchain, such as noncomparability, data fidelity, and traceability, as well as its identified management value regarding trust transfer (Qian and Papadonikolaki 2021), research on collaborative processes with blockchain as the underlying support has been gradually carried out (Jiang et al. 2023; Yang et al. 2020). In the present study, the approach of employing a Petri netbased workflow model configured as a SC is highly customized and lacks a standardized, automated method for its generation and visual execution.

Blockchain-Based SC for Construction BP Collaboration and Automation

Blockchain is a decentralized and transparent technology for secure and immutable record-keeping. It uses interconnected blocks to store data, ensuring tamper resistance and enhancing trust. It has applications in various construction BPs, offering benefits like transparency and increased efficiency (Torkanfar et al. 2023). A SC is a self-executing agreement with its terms directly encoded into code, facilitating secure and automated data interactions between on-chain (i.e., on the blockchain) and off-chain storages (Mars et al. 2023; Oraskari et al. 2024). This adaptability enables SC to address complex requirements, empowering developers to create decentralized applications with domain-specific functions (Hunhevicz and Hall 2020; Liu et al. 2023; Ye et al. 2022a). Meanwhile, as SCs are executed and recorded on the blockchain, their functionality inherits the key features of blockchain, including credibility (Ilbeigi et al. 2022), accountability (Zhu et al. 2019), and support for collaboration (Garcia-Garcia et al. 2020).

Blockchain-based SC offers significant advantages for BP collaboration and automation, as highlighted by various researchers. For instance, Mendling et al. (2018) emphasized the importance of SC, especially in interorganizational BPs, where rules governing responses to specific conditions can be expressed through SC. Di Ciccio et al. (2018) proposed leveraging blockchains as a technological infrastructure for BP collaboration, introducing an approach to execute processes through SC on the Ethereum blockchain. Such an approach can be considered as using a SC-based workflow engine. Garfatta et al. (2021) stressed that SC is ideal for implementing and automating BPs due to its ability to execute interdependent transactions and enforce rule-based relationships, aligning with the sequential tasks in a BP. Concurrently, Viriyasitavat et al. (2020, 2022a, b) underscored three key benefits of using blockchain in BP collaboration (i.e., build trust, reduce costs, and accelerate transactions) and conducted comprehensive reviews on blockchain as a service and blockchain applications, highlighting the immense potential of blockchain-based SC in BP collaboration and automation.

In recent years, there has been a substantial increase in published research on blockchain-based SC in various construction BP collaborations and automation. This growth encompasses diverse aspects, including general SC application reviews (Li and Kassem 2021; Ye et al. 2022a), and domain-specific reviews focusing such as on smart construction (Liu et al. 2023), construction supply chain (Yoon and Pishdad-Bozorgi 2022), and BIM security (Das et al. 2021). Yang et al. (2020) asserted that blockchain-based SC has the potential to digitize the future of construction BP in three key domains: smart contracts and cryptocurrencies, supply chain management, and information management. In addition, some blockchain-based SC applications were developed focusing on payment automation (Hamledari and Fischer 2021; Sigalov et al. 2021), modular integrated construction process (Jiang et al. 2021;

Li et al. 2021), and BIM versioning process (Tao et al. 2023). Two construction BPs, namely, cladding selection and equipment procurement, were demonstrated to test the feasibility of blockchain and SC in the construction industry (Yang et al. 2020). Leveraging the technical advantages of blockchain technology, fully or semi-automated construction BPs driven by SC offer enhanced trust and collaboration when compared to conventional software-driven methods.

Research Gaps

Most emerging technologies-enabled collaboration solutions did not drill down to the process level, but at the platform level, such as BIM-based collaboration platforms (Alreshidi et al. 2018; Lai et al. 2019; Zhang et al. 2017). The lack of standards for collaboration processes and the security problem of BP information being shared on platforms have been recognized as essential barriers. Applying blockchain with smart contracts as an upgraded workflow engine can enhance trust and improve BP collaboration from both the secure information environment and traceable process execution mechanism perspectives. However, using blockchain-based smart contracts at the process level to enhance BP collaboration is not a simple task for the following three reasons. First, construction BPs are complex and dynamic, which are difficult to predefine in SCs (Abuezhayeh et al. 2022; Köpke et al. 2019); Second, in the current smart contract solutions, smart contracts are defined case by case, meaning that when a construction BP is changed, the defined smart contracts cannot be used anymore (Ye et al. 2022a); Third, smart contracts are implemented as a bunch of codes, which are difficult to be read, write, or verify (Liu et al. 2022). These three reasons can be further interpreted and form three research gaps in the paper:

1. Unclear legal responsibility. The legal responsibility of SC generation and execution is a bilateral issue. On the one hand, SCs developed using high-level programming languages like Solidity or Vyper are not easily understood by practitioners without an IT background, creating collaboration barriers for construction project participants (e.g., clients, construction managers, and suppliers). On the other hand, assessing the legal requirements of SCs is challenging for IT specialists (Dwivedi et al. 2023). Therefore, distinguishing the composition and mechanism of legal responsibility in collaborating construction BPs is essential for SC adoption.
2. Lack of interoperability. Achieving seamless integration of BP-based SC with existing blockchain platforms and tools is a complex challenge that often leads to interoperability issues (Bodorik et al. 2023; Liu et al. 2022; Shen et al. 2023). This complexity primarily stems from the need to ensure the smooth interaction and compatibility between BP-based SC and the diverse array of blockchain systems in use. As blockchain platforms continue to evolve and diversify, achieving interoperability becomes crucial to facilitate the exchange of data and processes across these varied platforms. Interoperability challenges can lead to inefficiencies and complications in the execution of SC within construction processes.
3. Lack of standardization. The absence of standardized tools or guidelines from construction BP models (e.g., BPMN) to SCs generation poses a significant adoption barrier for existing approaches (Liu et al. 2022; López-Pintado et al. 2022; Loukil et al. 2021; Shen et al. 2023). Standardization is crucial to ensure consistency, reliability, and ease of implementation across different projects and organizations. With a lack of standardized tools, each entity or project may need to reinvent the wheel, resulting in inefficiencies and delays in the adoption

of BP-based SC. Establishing a set of best practices, guidelines, and tools tailored for this specific context is essential to streamline the integration of SCs into construction BPs.

The implementation of SC-based solutions in the construction industry holds great potential for enhancing collaboration and automating processes. However, the integration of SC-based solutions in construction faces the aforementioned challenges, which are crucial to be addressed for streamlining collaboration, automating processes, and ensuring successful adoption in the construction industry.

Research Methodology

The methodology is a guideline to design and validate the construction framework for BP collaboration and automation (Fig. 1). In Stage 1, the literature review is conducted for problem identification and objectives proposal. A conceptual framework is developed with three main parts in Stage 2. Finally, Stage 3 includes prototype implementation and validation. In this stage, a prototype is first developed and validated via a construction BP case, and the proposed framework is evaluated by a semistructured

interview associated with the prototype development and case study.

SC-Enabled Construction BP Collaboration and Automation Framework

Based on the objectives and research gaps of this research, the paper proposes an approach to generate and execute smart contracts using standardized BP models. This approach provides a standardized extension of process models to make them SC-executable, defines an automatic mechanism to map and generate from process models to smart contracts, and streamlines the smart contract execution by visualizing the SC functions using process model tasks. Consequently, the framework can be divided into three key parts (see Fig. 2): (1) Action definition and extension to process models; (2) Standardized mapping and automatic SC generation; and (3) SC visualization for reliable process collaboration and automation.

Action Definition and Extension to Process Models

The proposed framework involves designing SC-executable actions, initially defined by an “Action” tag representing various

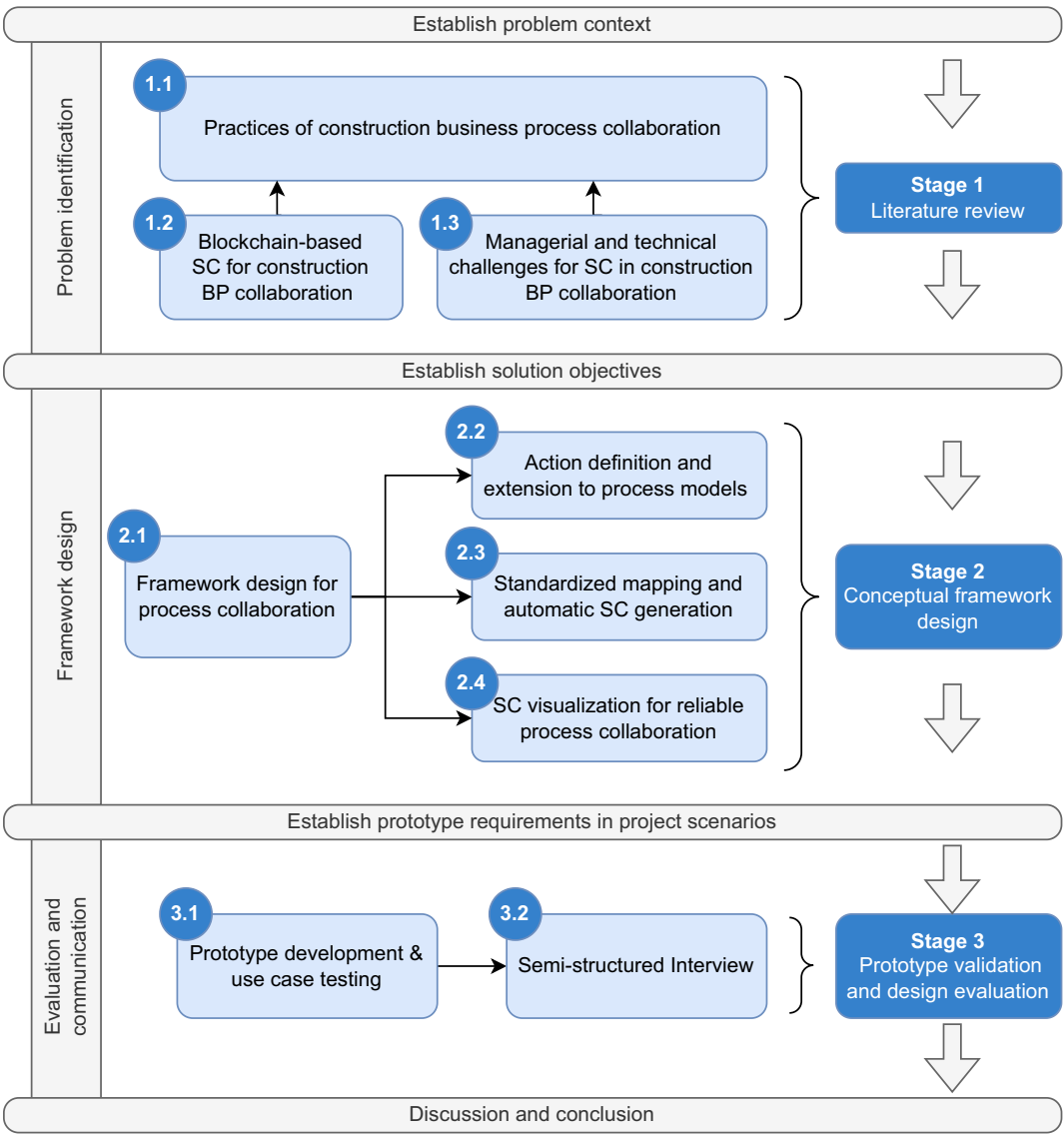


Fig. 1. Research methodology.

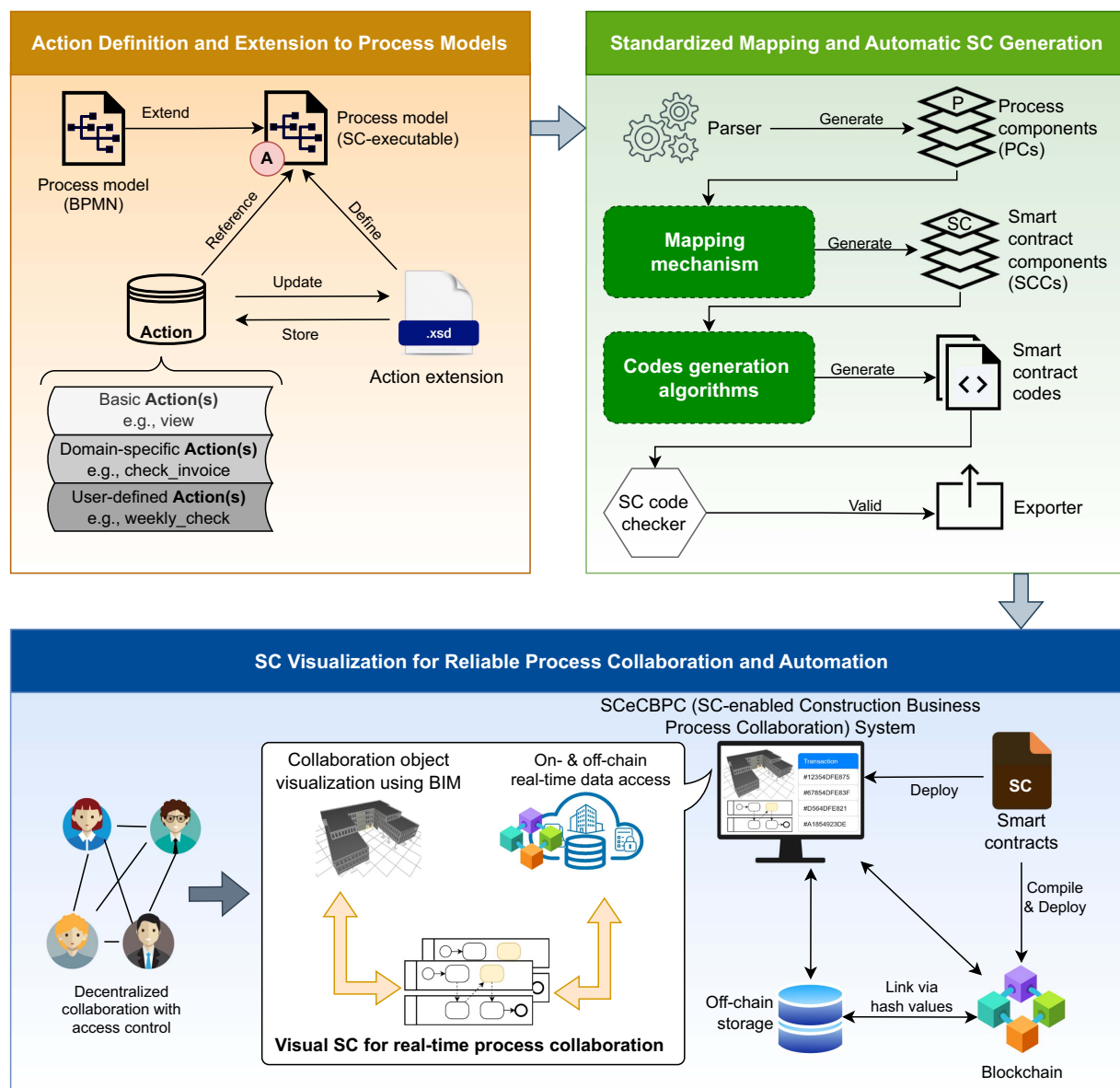


Fig. 2. Proposed framework for construction BP collaboration and automation.

SC-executable behaviors in tasks. Actions are categorized as manual, semiautomated, or fully automated based on task automation degrees. Semiautomated or fully automated actions can be implemented through SC or RPA technologies, following the TBM method by (Shi et al. 2008) to guide automation possibilities. For example, a “send email” task involves automated actions with specified contents (e.g., using Fortra RPA for email automation), while a “make decision” task can be defined as a semiautomated action performed by a manager assisted by tools.

However, these action degrees are dynamic, contingent on actual process scenarios, and can evolve with technological advancements. A “send email” action can be automated in one case but semiautomated in another and can be improved with the technology development from manual to semiautomated or even fully automated. To address this, a deterministic and extensible action classification system is proposed based on the development logic of programming and modeling languages (Bagge and Haverlaen 2003; Lee et al. 2015; Lucic and Blake 2015; Siek 2010). Actions are defined at three levels: general (basic

for general purposes), domain-specific (contextualized within construction BP domains), and user-defined (customized under specific user requirements). The initial definition of actions with level classification is presented in Fig. 3. Level 1 includes frequently used actions of construction BP tasks like “add” and “delete” for read-write operations, and “link” and “reset” for related data transmission. These actions are crucial for handling data to be stored in the blockchain. Level 2 introduces domain-specific actions like “pay” and “notify” for the contract and payment process and “delivery” and “store” for the supply chain process. Level 3 allows users to define custom actions, ensuring extensibility for future applications.

As a general action, “add” is for adding additional securely storing and tracing crucial information, such as the agreed completion deadline of certain construction work, on the blockchain. In Level 2, the “pay” action can be used to trigger an automated payment from one user address to another with a specific amount and payment method. The “notify” action allows manual message composition instead of merely automatic notifications, which is useful in

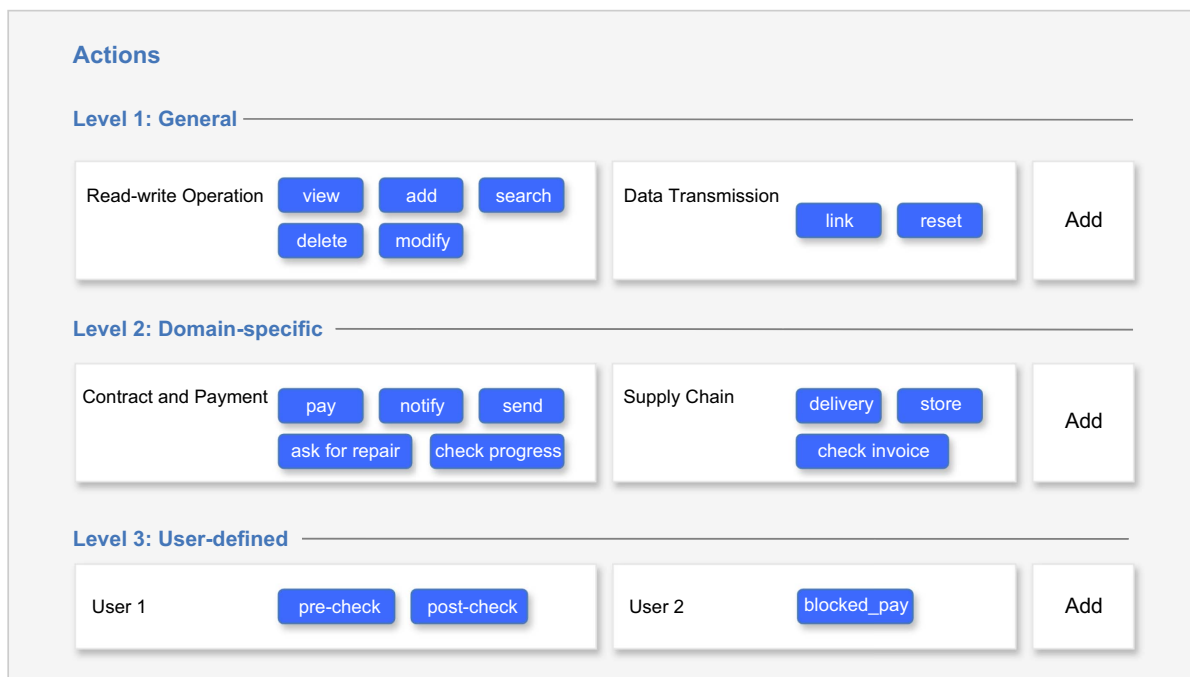


Fig. 3. Definition of actions with level classification.

situations like reporting construction defects. For example, when clients detect a defect in the construction work, they can activate a “notify” action with a specific message, indicating the repair reason and the damage level. Additionally, the “send” action facilitates file transfer, requiring participants to upload files. The hash values of the uploaded file should be generated automatically and stored on the blockchain or connected to off-chain storage.

For standardizing the defined actions, Fig. 4 illustrates the extensible markup language (XML) schema definition (XSD) for extending any XML-enabled BP modeling languages, such as BPMN. The full XSD of defined actions is available on GitHub

(Ye 2022). One or more *action* elements may be defined in the *actions* element with an identifier to distinguish different actions. A *tRPAType* type to indicate if the action can be realized using RPA programs and optional attributes for the user-defined extension. The *rpaType* is defined based on type *tRPAType* whose value can only be one of *FullyAutomated*, *SemiAutomated*, or *Manual*, where fully or semiautomated actions are fully or partly executed by RPA programs. An action may or may not be associated with input and output variables (i.e., *variableInput* and *variableOutput*) according to its design, where these variables are defined based on the characteristics of each action.

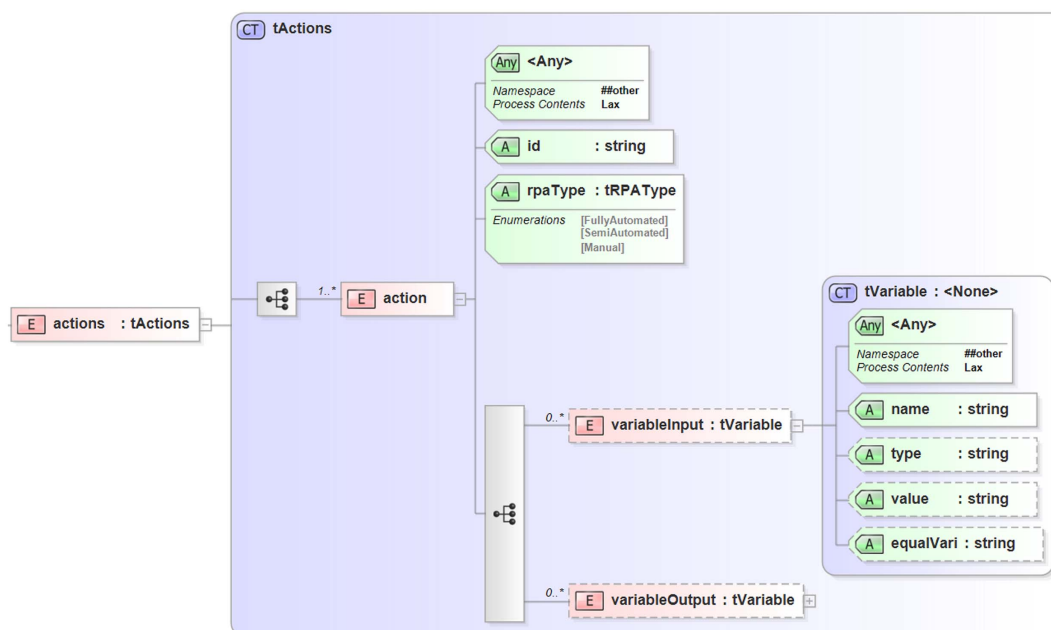


Fig. 4. The XML Schema Definition Visualization of actions.

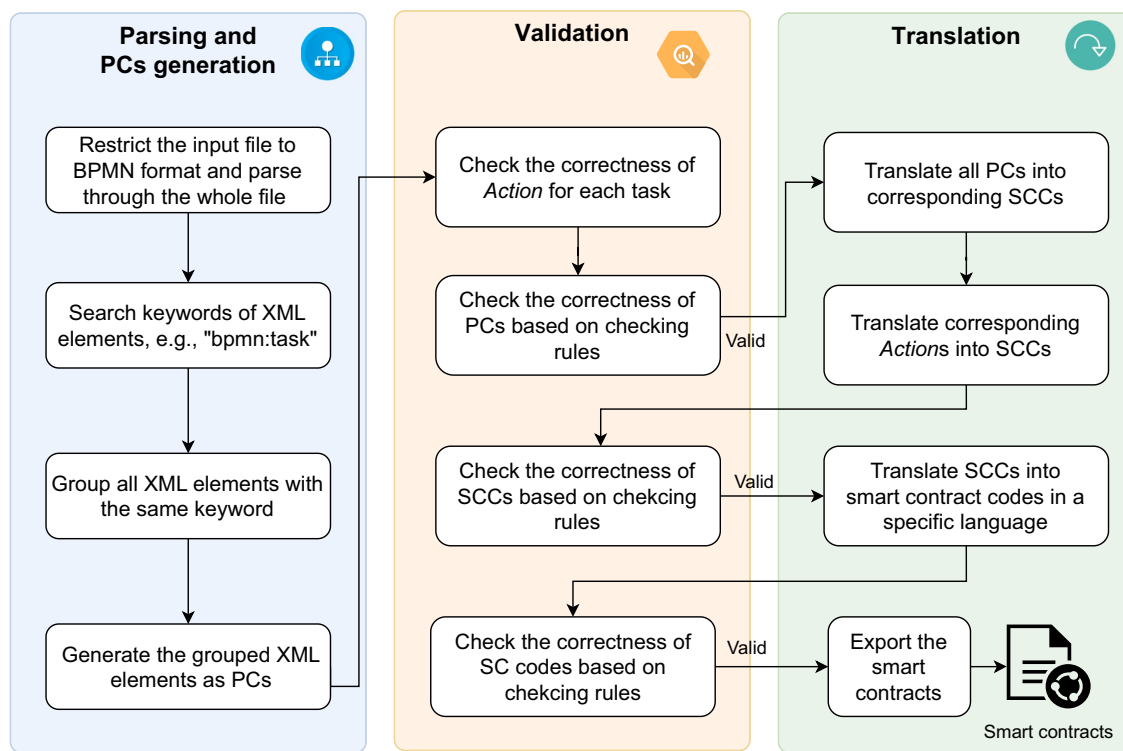


Fig. 5. Workflow of SC generation.

In the context of process automation, the paper proposes replacing manual construction BP tasks with fully or semiautomated actions leveraging SC as an RPA program, exemplified by the “pay” action. Nevertheless, for manual actions such as “check progress” that are not seamlessly executable via SC, the combination of SC and other programs or devices, such as IoT sensors, becomes necessary. As a result, integrating these actions into construction BP models and executing them via SC contributes to enhanced process automation.

Standardized Mapping and Automatic SC Generation

To ensure objective comprehension and programming precision, standardized mapping and automatic SC generation are designed as guaranteed mechanisms for programming responsibility. Once actions are defined within a construction BP model, the model undergoes a crucial step: generation into SC for execution and visualization. The SC generation procedure unfolds in three phases, as illustrated in Fig. 5: parsing and process components (PCs) generation, validation, and translation. Following the creation of a construction BP model with an extension of actions in the BPMN XML standard (see Fig. 4), it becomes the input for SC generation. Such input is parsed to generate PCs, storing all model information in a structured manner based on process modeling language logic (section “Process Component Generation”). These PCs are then translated into SC components (SCCs) with validation, which mirror the information under the SC language logic. These SCCs, after further validation, are ultimately translated into SC codes, enabling the realization of an SC-based workflow engine and enhanced automation. Detailed descriptions of the translation and validation phases are presented in the sections “Mapping Mechanism and Action Generation Algorithms” and “Generation Checking Mechanism,” respectively.

Process Component Generation

In the parsing phase, a process model (e.g., in BPMN format) is the input of the SC generation. As process modeling languages

(e.g., BPMN) can share the same logic of information, a data structure called a PC is defined to store all information in a process model. Because the PC shares the same BPMN keywords, which provide clear information on the process model, it is easily translated into SCCs. The present study only considers BPMN, as it is the most-recognized process modeling language in the construction industry. When a process model is imported into the SC generator, only files with the BPMN extension (“.bpmn”) will be parsed. During the parsing procedure, keywords (e.g., “bpmn:participant,” “bpmn:task,” “bpmn:sequenceFlow”) are located to group different XML elements. Each grouped XML element is defined as a PC, and the entire process model is parsed and divided into three main PCs: participants, tasks, and flows. Each task PC includes information regarding its corresponding participant (task operator) and actions.

Mapping Mechanism and Action Generation Algorithms

If all generated PCs are valid, they are translated into corresponding SCCs via the action and SCC translators. A corresponding translation mapping is illustrated in Fig. 6. The action translator translates actions of task PC into input parameters, return variables, called action functions, and contents in the *SCGAction* contract of SCCs (as marked in red in Fig. 6), whereas the SCC translator handles the rest. In the SCC translator, the “Participants” PC are translated into modifiers and state variables of SCCs, tasks are translated into functions, and flows are translated into specific function statements and contents of the *SCGFlow* contract. All SCCs can be divided into three contracts: main contract, *SCGFlow* contract, and *SCGAction* contract. All contents from a specific BPMN file are translated into the main contract, the logic of flows and gateways is handled by the *SCGFlow* contract, and the defined actions are handled by the *SCGAction* contract. SCCs follow the logic of SC codes, and the translation from PCs to SCCs is equivalent to that from a process model to SC logic.

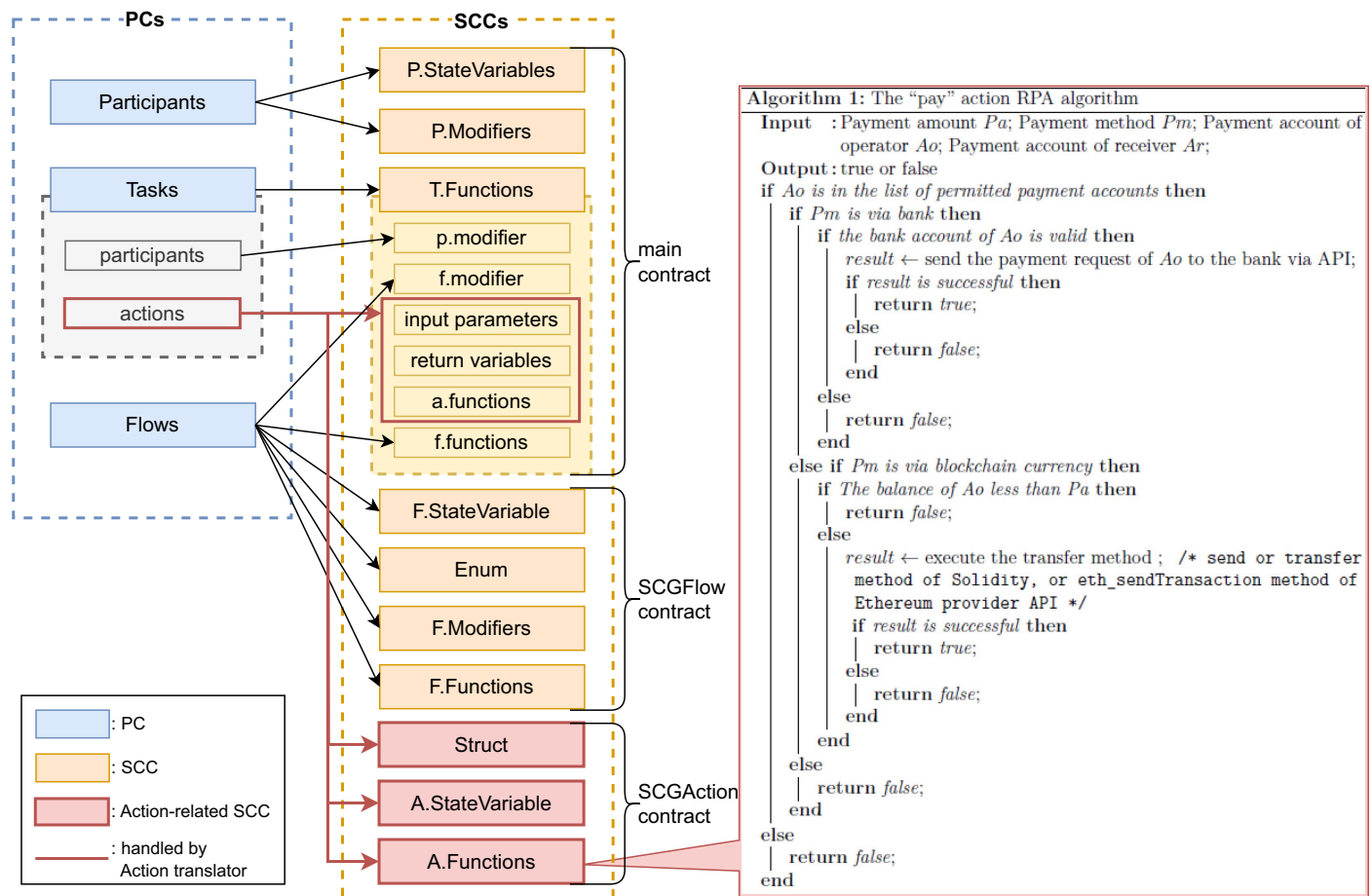


Fig. 6. Translation mapping with a "pay" action algorithm example.

There are five types of SCCs (Fig. 6): state variables, modifiers, functions, enum, and struct. A state variable is a variable used in a contract, a modifier is used to restrict a function's behavior, and both enum and struct are data structures used to define a variable's data type. In the translation of participants, $P.StateVariables$ stores all information variables of participants (e.g., payment account), and $P.Modifiers$ restricts the operation of certain task functions ($T.Functions$) to specific participants. $T.Functions$ stores all functions translated from tasks, where $p.modifier$ and $f.modifier$ encompass modifiers of participant and flow, respectively.

Each action defined in tasks is translated into an action function (i.e., $A.Functions$ in Fig. 6), and executing this action is by calling this action function in its task function (i.e., calling an $a.function$ in a $T.Function$). When the action type is fully or semiautomated, executing such action is the realization of process automation. All defined action functions can be directly executed via SC with or without external help from other technologies or systems. For example, the "pay" action is defined to automatically operate a payment function and store the corresponding information in the blockchain. The algorithm for translating this action (shown in Fig. 6) is subsequently translated into SC codes. The automation level can be further improved by extending these action functions. For example, to extend the "pay" algorithm, a "blocked_pay" action can be user-defined for more secure payment, such as using blocking process logic (Ahmadisheykhsarmast and Sonmez 2020). Moreover, some manual actions can be replaced by semiautomated or fully automated actions when more technologies are combined into SC. For example, by combining IoT senses with SC, construction progress can be automatically recorded and checked. Thus, a

"check progress" manual action can be improved into a semiautomated or fully automated action.

As the final translation step, SCCs are translated into SC codes in a specific language (e.g., Solidity). Although SCCs are designed based on Solidity logic, they can be translated into other SC languages, most of which are object-oriented programming languages that allow for direct translation from SCCs. For example, when using the Node.js language as SC (e.g., chaincode of Hyperledger Fabric), each contract is translated as a class, modifiers are translated as functions, and all state variables are denoted by global variables. However, the present study only considers Solidity, as it is currently the most-recognized SC language. The generated SC codes can be exported and further used in an Ethereum blockchain platform or any other Solidity-supported distributed ledger platform, such as Hedera Hashgraph (Baird et al. 2020).

Generation Checking Mechanism

Based on the workflow defined in Fig. 5, four validation procedures are defined to validate the generated actions, PCs, SCCs, and SC codes. After parsing and generation, the PCs are validated by the PC checker, first verifying the existence of "Action" in tasks and then checking the attributes for accuracy. If there are no errors, the PCs and their corresponding actions are translated into SCCs (such as modifiers and functions), which are then verified by the SCC checker. All verified SCCs are then translated into SC codes in a specified programming language. The generated SC codes are checked by the SC code checker according to specific rules (e.g., the rules defined for checking Solidity SC codes shown in Fig. 7) and exported upon verification. In addition to the checking

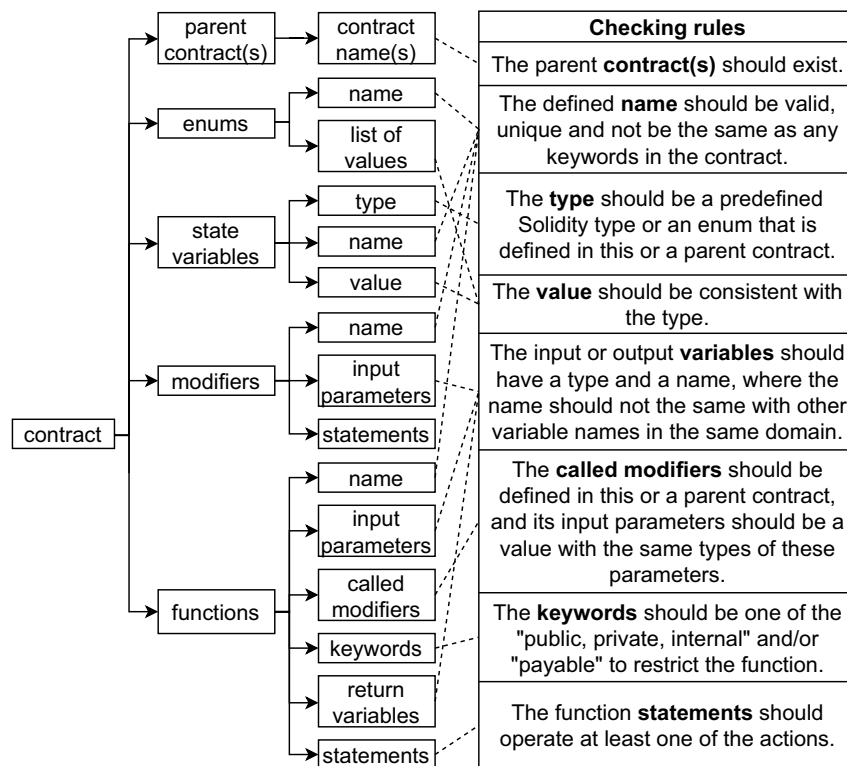


Fig. 7. Checking rules of Solidity SC codes. (Reprinted from Ye and König 2021b.)

rules in Fig. 7, more rules are specified according to the action definition. For example, if a function is denoted as a “pay” action, the payment accounts of the payer and payee, as well as the payment method and amount, must be valid.

SC Visualization for Reliable Process Collaboration and Automation

This procedure details the SC visualization for reliable process collaboration and automation (Fig. 2). The generated SC codes from the last procedure are imported into this procedure, compiled and deployed into the blockchain network, and then deployed into a self-developed decentralized application (dApp) called SC-enabled construction BP collaboration (SCeCBPC) System. The interaction (e.g., messaging and signing) between the blockchain network and the system is done via interfaces (e.g., application binary interface for Ethereum blockchain). An off-chain storage, namely, a database external to the blockchain, stores all the large-scale files that need a lower level of security and traceability. All off-chain data, such as files uploaded via the “send” action, are stored in the off-chain storage, whereas their corresponding hash values are stored in the blockchain.

Within the SCeCBPC system, the BPMN process model is used to visualize SC for real-time process collaboration by linking every BPMN task with each corresponding SC function. Meanwhile, each BPMN process model is seamlessly interconnected with its respective BIM model (or BIM elements), thereby enabling the real-time visualization of construction project progress. All data generated during the specific construction BP are stored on-chain or off-chain based on data demands. In this way, the system can be considered as an implementation of a reliable SC workflow engine under the construction sector, executing the generated SC under the workflow logic designed in the BPMN process model. The connectivity among SC functions, BPMN tasks, BIM, and data storage

facilitates the observation and tracking of construction projects in the system. It is also noteworthy that each construction project may encompass multiple distinct BPMN models as per its specific requirements.

Decentralized collaboration among participants is realized by the combination of BPMN, SC, dApp, and blockchain. Based on the roles designed in the BPMN diagram, its corresponding SC is programmed with function access control. In practice, this translates to a situation where only specific roles or designated accounts can execute specific SC functions, strictly in accordance with their assigned role or participant status for a given BPMN task. Moreover, fine-grained control over data visibility is afforded by granting different permissions to various roles within the system depending on whether they are accessing the frontend, blockchain, or off-chain storage repositories. This access control strategy ensures that the right participants have access to the right functions and data, enhancing the overall security and integrity of the collaborative process.

Implementation and Evaluation

To foster trust in the applicability and address real-world scenarios of the proposed framework, this chapter designs a use case for testing the framework, implements the prototype system (includes a tool for SC generation and a web-based application for SC visualization), and evaluates the framework and the prototype system based on the insights from industrial (including IT and construction fields) and research experts (Fig. 8).

Case Introduction

The proposed framework is tested through a case study of an office building construction project in Germany. The office building, located on the south shore in the second row of Lake Phoenix on an

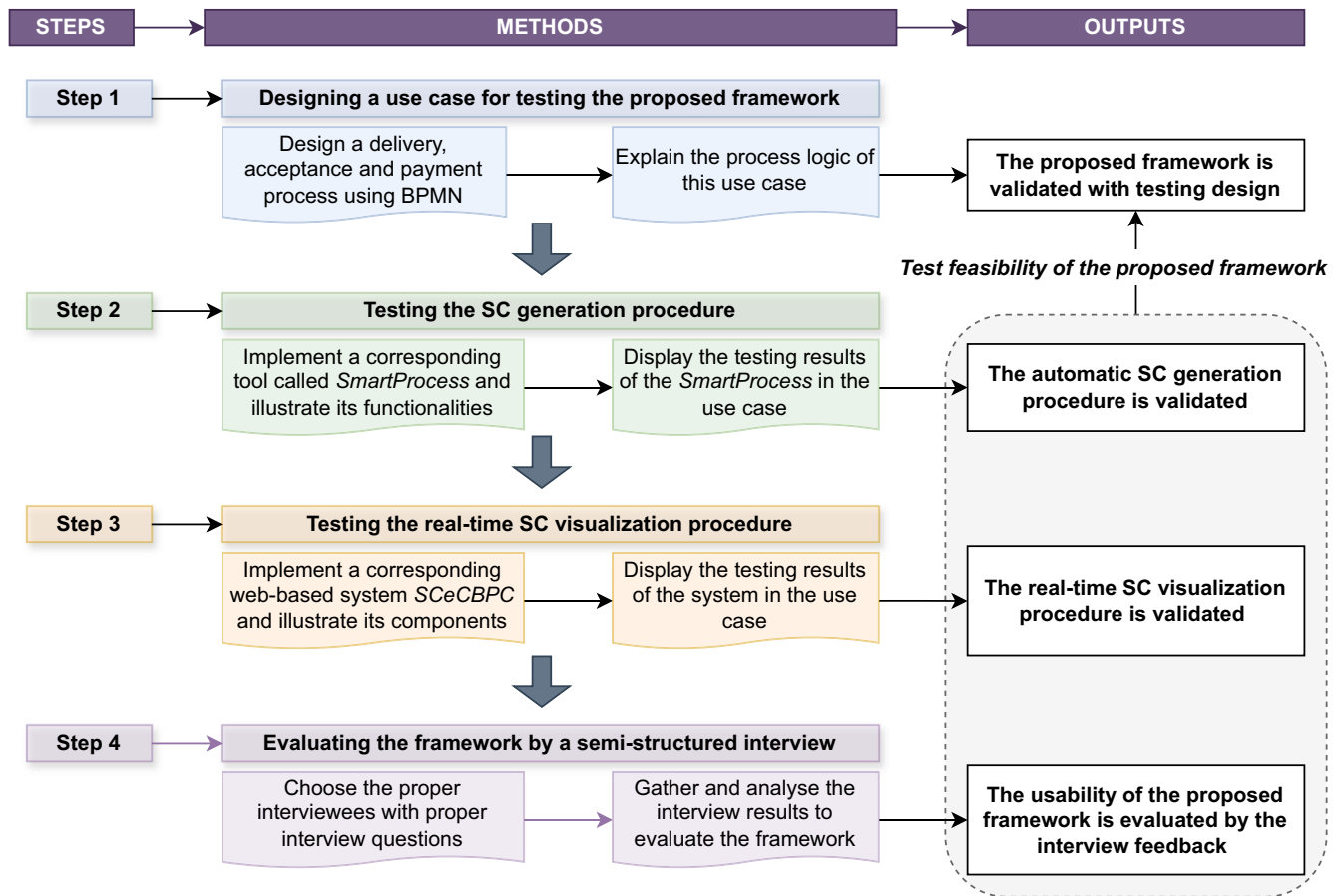


Fig. 8. Implementation and evaluation steps.

approximately 3,200 m² site on Phoenixseestrasse in Dortmund, Germany, features a four-story, U-shaped design. With a gross floor area of 4,500 m², the building includes two roof terraces on the recessed second floor, above-ground parking, and a main entrance on the northwest side. The building, intended for leasing, commences construction 14 months post-contract, coinciding with a four-month lead time that overlaps with the building permit phase. “Südufer GmbH & Co. KG” (the client) oversees the project, which employs BIM for preliminary planning. A fixed-price contract with a selected general contractor covers all aspects from planning to turnkey delivery. The contract includes standard provisions for handling defects and incomplete services, emphasizing the importance of documentation and the financial implications for the contractor. In this case, the primary focus is on the two key stakeholders, the client and the (general) contractor, along with their interactive business and management behaviors in the construction delivery, acceptance, and payment (DAP) process. This case scenario is developed based on the insights from project team members, including the contractor company, a legal firm specializing in the German construction industry, and several software and research institutions.

Implementation

A BPMN example of the DAP process is designed based on the construction project and illustrated in Fig. 9. This DAP process represents the lifecycle of a group of certain construction work (called billing unit, abbreviated as BU) that can be paid at once. The process includes two participants (i.e., the contractor and the client) and nine tasks.

To test the integrity and functionality of the action creation and SC generation procedures, a SC generator (Fig. 10) called *Smart-Process* was developed to ensure that the generated smart contracts accurately represent the logic of the BP and to address trust concerns related to the reliability and accuracy of the generated smart contracts. The source codes of the generator are available on GitHub (Ye 2023). These features include (1) displaying the BPMN diagram and adding actions [Fig. 10(a)]; (2) parsing the BPMN process model into a structured format as a set of PCs [Fig. 10(b)]; (3) translating the PCs into SC components using the mapping mechanism and code generation algorithms introduced in the proposed framework [Fig. 10(c)]; (4) displaying the checking results throughout the procedures [Fig. 10(d)]; and (5) generating the SC codes as the final results of the tool [Fig. 10(e)].

Using the “Pay BU” task as an example, its generation procedure is highlighted in purple in Fig. 10 to illustrate relevant functionalities. Upon selecting a specific task (labeled as number 1), the inner actions (“add” and “pay”) are displayed, and additional actions can be added (labeled as number 2). Once all actions for each task are created, the BPMN model becomes SC-executable and can be parsed into PCs via a “Parse” button. The “PayBU” Task PC, labeled with the number 4, is then transformed into a corresponding Function SCC through a “Translate” button. In this translation, the client participant is constrained to ensure only the client can operate this function, and actions are linked with the “Action” data pool to invoke the corresponding action functions. The final SC function codes (labeled as number 8) are obtained from the Function SCC via another “Translate”

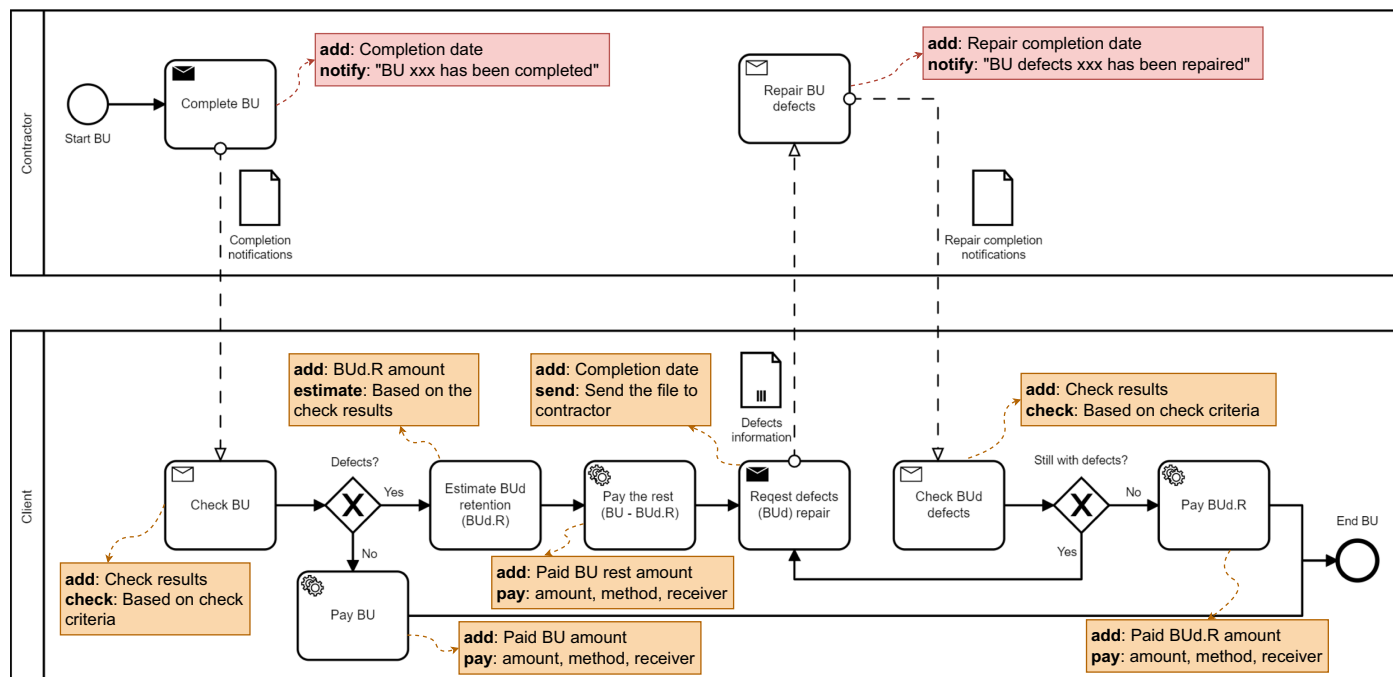


Fig. 9. A BPMN example of a DAP process.

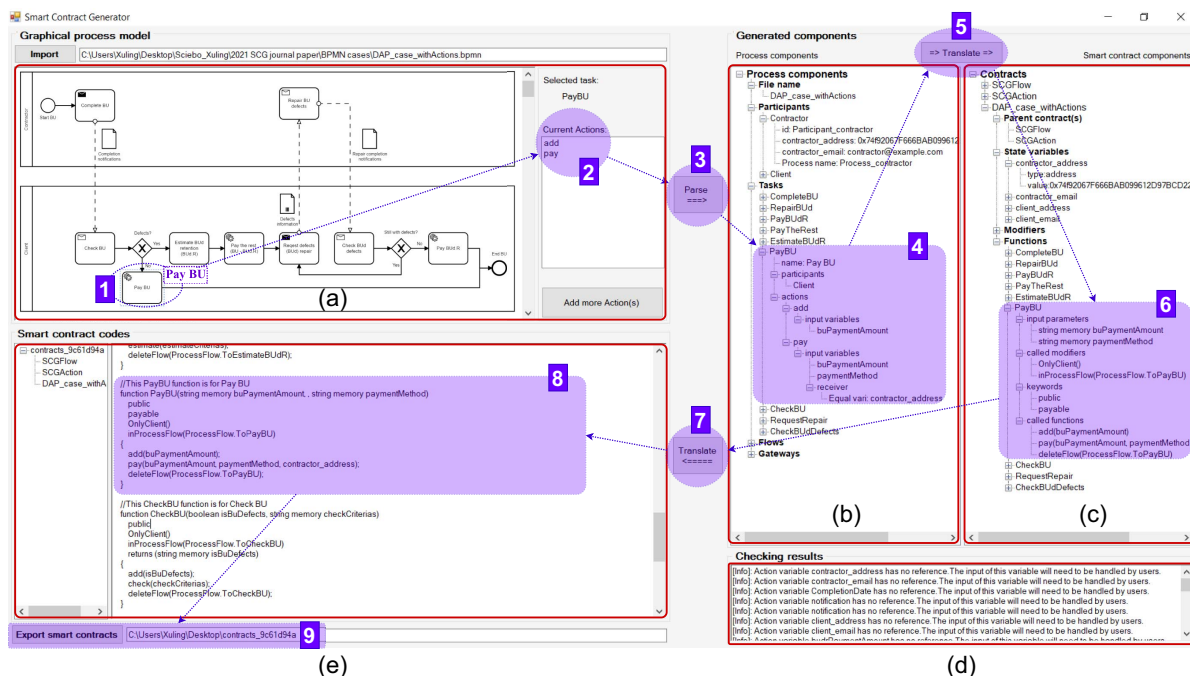


Fig. 10. Generated results of the SmartProcess.

button. Finally, the entire generated contract folder containing three SCs can be exported for further deployment, execution, and visualization (labeled as number 9).

The mapping from PCs to SCCs is presented in Fig. 11. Each participant's PC, along with its identifier, address, email, and process name [Fig. 11(a)], is translated into corresponding state variables and a modifier to store all necessary information and restrict the function operator in SCCs [Fig. 11(e)]. Each task flow PC indicates the task sequence, and its next task can be

either a gateway or a task [Fig. 11(b)]. The corresponding logic is translated into an *SCGFlow* contract [Fig. 11(c)]. For example, the "CheckBU" flow PC contains an OR gateway with two outgoing tasks, whereas the "PayBU" flow PC has only one outgoing task. The logic of all actions is handled in the *SCGAction* contract [see Fig. 11(d)]. For example, a pay RPA action with three input variables is realized in the *SCGAction* contract and can be called by an automated payment task to realize the payment RPA.

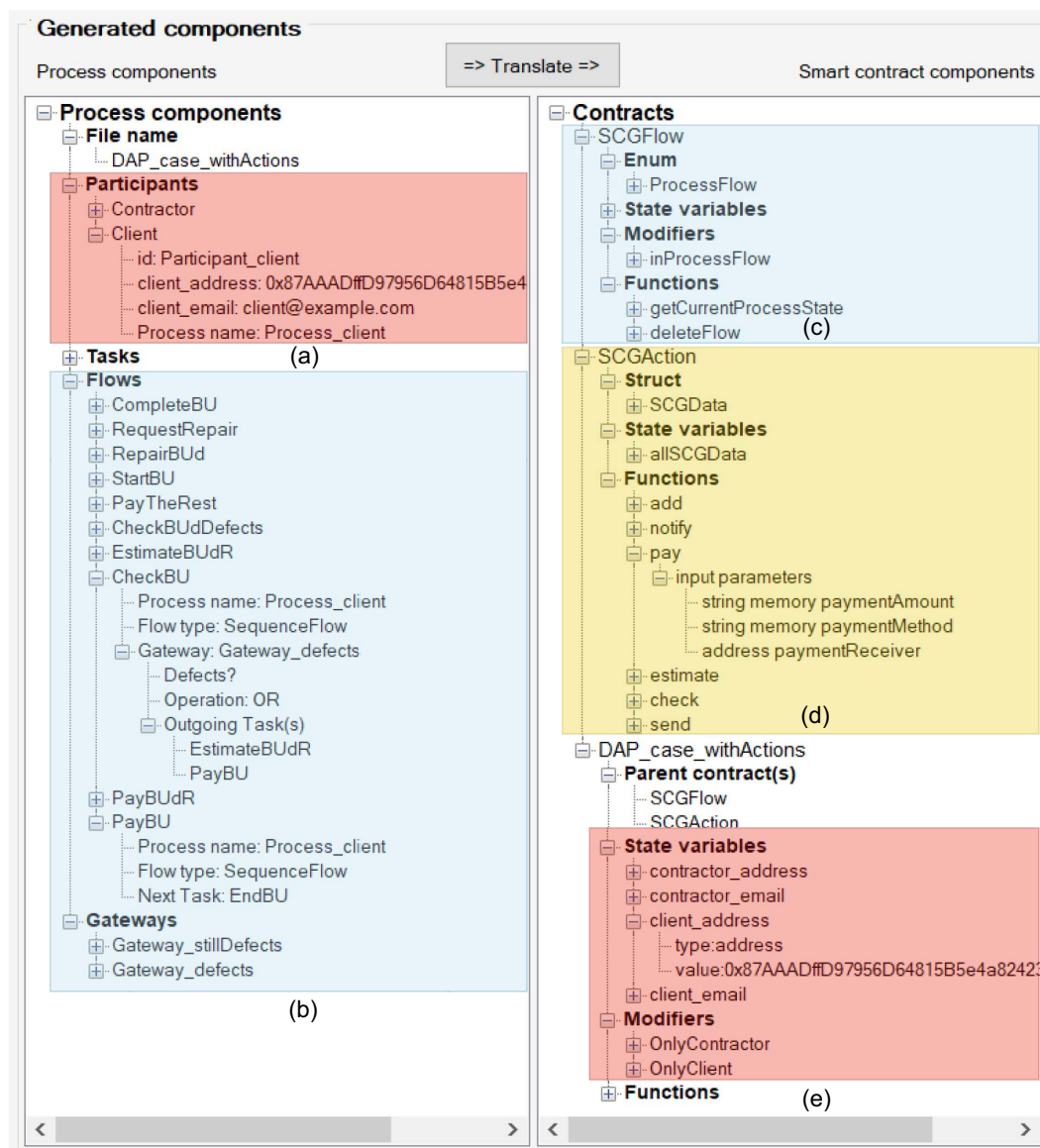


Fig. 11. Generated results from PCs to SCCs.

Validating the real-time visualization of SCs through the implementation of a web-based system (called SCeCBPC, with the implementation detailed in Ye et al. (2022b)) increases transparency and accountability in the execution of BPs, allowing stakeholders to track and verify the progress of tasks, thereby improving trust in the automation process. The generated SC codes can be directly deployed into the blockchain using tools such as Remix or Hardhat for the Ethereum blockchain. All transactions executing the construction DAP process using the deployed SC are stored in the Ethereum blockchain and are illustrated in Fig. 12.

The corresponding SCeCBPC system for executing and visualizing deployed SC for process automation is shown in Fig. 13. The entire system is implemented by creating a frontend using React.js and connecting it to the deployed SC using the provider from Ethers.js. All required construction works were displayed using a BIM model and grouped into five BUs within a billing plan [Fig. 13(c)]. Each BU features its own SC process [Fig. 13(a)], which is linked to the deployed SC functions [Fig. 13(b)] via the application binary interface (ABI) to visualize the execution status.

The information stored in the blockchain is printed, as shown in Fig. 13(d).

The framework realization in the developed system was demonstrated and validated using a payment example. Generally, clients are required to pay contractors manually by transferring money via a bank after the completion of certain construction work, and the transaction is only stored in the bank without linking with any other systems for further operations. In this study, after uploading the construction contract (Ye and König 2021a) and the corresponding BPMN process model into the system, all the payment-related information (e.g., payment amount, payment method) is parsed and stored in the system. The “pay” action enables secured and automated payment via SC. When executing the “PayBU” task, the inner RPA action “pay” is activated, and the payment is automatically operated by the corresponding SC function via the blockchain payment method (see “pay” algorithm in Fig. 6). An automated payment via blockchain can be made by the MetaMask extension from the Ethereum blockchain using its provider application programming interface (API). Such payment procedure is illustrated in Fig. 13(e).

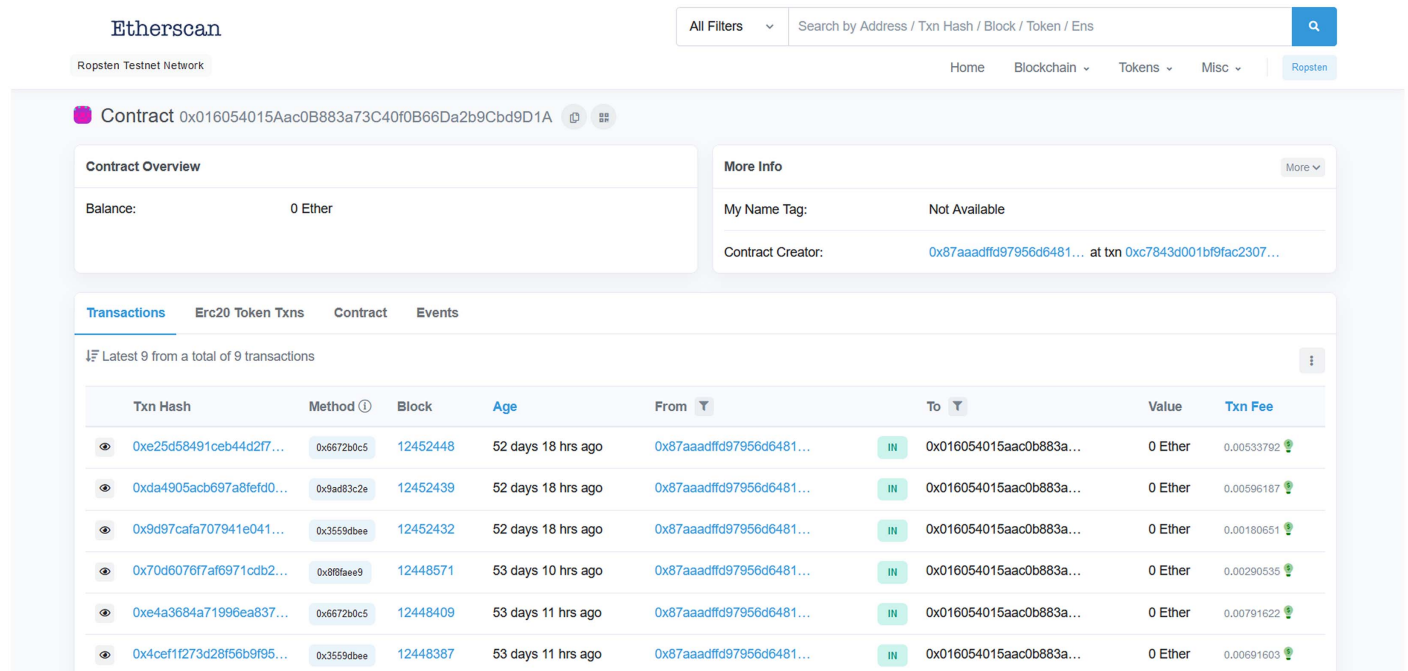


Fig. 12. Transactions of SC execution.

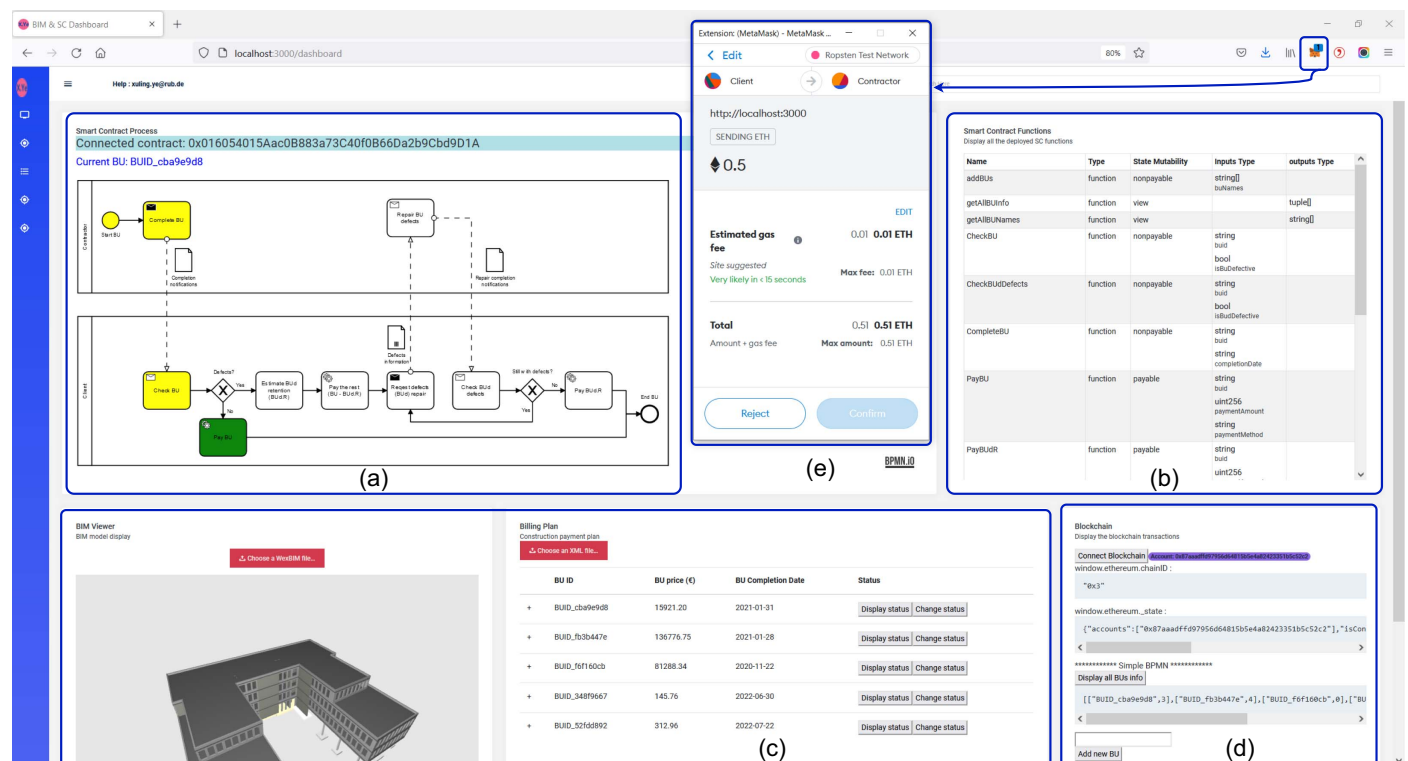


Fig. 13. SC visualization results for executing the “PayBU” task with pay action: (a) smart contract process; (b) smart contract functions; (c) billing information; (d) blockchain information; and (e) pay RPA action via blockchain.

After successfully executing the task with a “pay” action, the updated result is shown in Fig. 14, where the successful transaction result is returned to the system and the background color of the task changes from green to yellow indicating the task status changed from executing to executed. Thus, the system not only functions as an SC-based workflow engine to execute a process model but

also carries out automated actions in each task to improve process automation, even providing possible extensions to further automate actions.

In addition to the automated actions defined in this approach, manual actions can be converted into automated actions by combining SC with other technologies or programs. For example, a

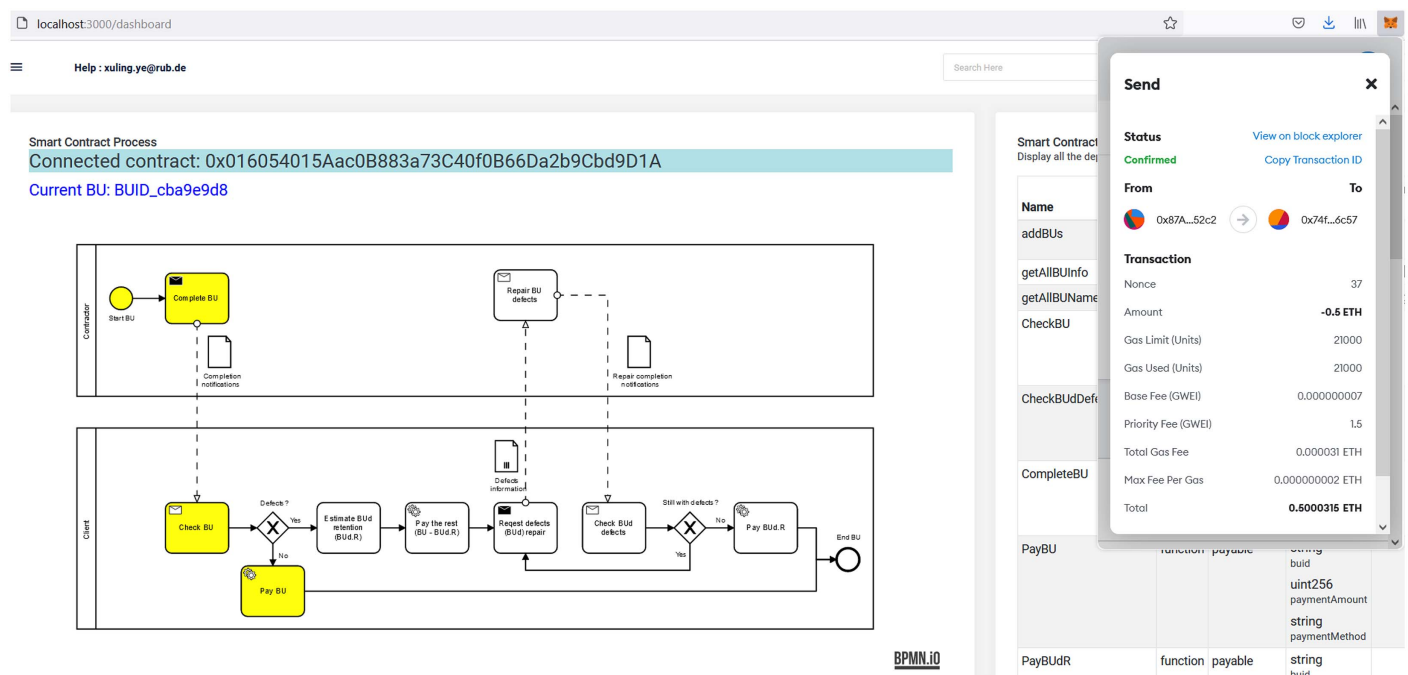


Fig. 14. SC process update after executing the “PayBU” task for payment.

“send email” manual action can be replaced with an automated action by combining SC with an email agent (e.g., Twilio SendGrid) for automatic execution.

Evaluation

Through semistructured interviews with industrial and research experts, the usability and practicality of the framework and the developed prototype system are evaluated, providing valuable feedback on the advantages, disadvantages, and suggestions for further development, contributing to the continuous improvement of the framework’s efficiency and effectiveness. Nine interviewees from both industrial (including IT and construction fields) and research backgrounds were invited to evaluate the proposed framework and the implemented prototype by answering questions in the third stage (refer to Appendixes I, II, and III). The nine interviewees encompassed a diverse array of professions, spanning IT software, IT security, the client (owner), construction main contractors, engineering offices, construction digitalization, and research domains. Their feedback was solicited from various areas of expertise. In this structured interview, four questions about the advantages and disadvantages of the proposed framework, willingness to use this framework, and suggestions for further development, were asked (see Appendix III).

In response to the first question about the advantages of the proposed framework, more than half of the interviewees (5 out of 9) considered data traceability as a significant advantage. The proposed framework was believed to be useful for certain real-world scenarios and was seen as having the potential to provide insights for an integrated platform by certain interviewees (3 out of 9). Especially, the interviewee from the client side (OW1) emphasized the proposed framework would be suitable for data integrity, reducing doubts, improving visualization in process collaboration, and simplifying management problems and future risk management.

Regarding the second question about the disadvantages of the proposed framework, these interviewees mainly discussed two aspects, namely, the functional design and the additional efforts.

For the functional design, four interviewees were concerned about the lack of flexibility for BPMN. Meanwhile, three interviewees mentioned that there is no one-size-fits-all solution for all processes in the construction industry. One experienced project manager from the main contractor company thought the process automation should be guaranteed error-free. As for the additional efforts, it was noted by one interviewee that user-defined actions could potentially increase the workload.

In response to the third question about the users’ willingness to use the proposed framework, 5 out of 9 interviewees replied that they would use it without condition. Only one project manager hesitated to use this framework and doubted if it could significantly save time and effort compared to the current work mode. The specification of the framework is essential as a condition for the other three interviewees to use.

Interviewees provided valuable opinions in response to the fourth question about suggestions for further development. Process flexibility was suggested by 5 out of 9 interviewees, and framework smoothness was advised by two interviewees. One interviewee from the contractor side (MC1) suggested considering reusability, availability, ease of use, and improved efficiency. Another from the contractor side (MC2) emphasized to consider data accessibility and integration based on a more specific problem. One interviewee from the research institute suggested defining process management by constraints.

Discussion

Most construction automation research focuses on the automatic execution of production activities or construction tasks (Alaloul et al. 2021; Muhammad et al. 2021). In contrast, the present study considers solutions and provides implications for the collaboration and automation of construction BPs in which research outcomes lag behind the informatization and automation of production processes in construction. After exploring the feasibility and superiority of integrating blockchain-enabled SCs and process automation

techniques, this study systematically developed SC generation and visualization mechanisms and demonstrated a smart contract-enabled construction BP collaboration system to not only advance the technical domain but significantly bolster trust among stakeholders.

With the development of blockchain and smart contracts, a new generation of workflow engines, which offer distributed, trust, and automated execution of construction BPs, has been developed (Hamledari and Fischer 2021; Ye and König 2021b). Therefore, the requirements of trust and information security for BPA due to the peculiarities rooted in the construction field can be met. This study extended existing research outcomes and addressed critical issues in a new technological environment. Specifically, this study explored the superiority and technical feasibility of adopting smart contract-based workflow engines for flow automation and simultaneously applying RPA actions for task automation. The contributions of this study are listed as follows.

From the managerial perspective, we provided a blockchain-based SC solution to shift trust from individuals to a secure digital infrastructure. Demand-oriented functional design and technical support for a clear division of legal responsibilities are the basis for process collaboration and automation based on SCs (Lu et al. 2023a). Based on the gradually formed and strengthened consensus on the acceptance of blockchain technology (Li and Kassem 2021; Ye et al. 2022a), we strengthened the credibility, security, and effectiveness of SCs through customized and versatile modules and standardized mapping and generating progress.

Recognizing the significance of the interoperability issue, we took a holistic approach that encompasses the entire process from BPMN model generation to SC execution. By considering the complete procedure, we ensured that our BPMN-based smart contracts are not only compatible with a wide range of blockchain platforms but also seamlessly interact with existing blockchain tools and ecosystems. This comprehensive approach eliminates the barriers to interoperability, offering organizations a versatile and harmonious integration solution.

Regarding the standardization issue, we went beyond theory and provided practical support. Our toolkit, available on GitHub, streamlines the BPMN to smart contract generation process, offering a user-friendly and standardized solution. Furthermore, we offer comprehensive guidance for the mapping mechanism, simplifying the intricacies of the conversion process. Additionally, we provide detailed instructions for the smart contract generation procedure, ensuring a consistent and straightforward approach. This robust set of resources collectively addresses the tooling and standards gap, empowering organizations to readily and confidently embrace our approach without the burden of tooling or guideline development.

In addition, we rigorously tested and evaluated our framework in a real-world construction payment case. This practical scenario

served as an evaluation of our approach, demonstrating its applicability in a tangible business context. Through the inclusion of the case study, we not only evaluated our framework but also provided researchers and practitioners with insights into its real-world benefits, demonstrating its potential to enhance efficiency and trust in collaborative and automated construction BPs.

Conclusion

In the context of applying blockchain, SCs, and process automation technologies to construction BPs, this paper addressed both managerial and technical challenges. The study presented a comprehensive technical framework aligned with construction practice, emphasizing the resolution of legal issues in SC applications while meeting process collaboration and automation requirements. The framework comprised three main modules: customized and versatile modules for SC generation, standardized mapping and automatic SC generation, and SC visualization for reliable process collaboration and automation. From a managerial perspective, this study proposed a blockchain-based SC solution that shifted trust from individuals to a secure digital infrastructure, highlighting the importance of demand-oriented functional design and a clear division of legal responsibilities for process collaboration and automation. The study adopted a holistic approach, ensuring compatibility of BP-based SC with diverse blockchain platforms and tools, facilitating seamless interaction within blockchain ecosystems. Addressing the standardization challenge, the paper provided practical support through a user-friendly toolkit, comprehensive mapping guidance, and detailed SC generation instructions. Finally, the framework's practicality was evaluated through prototype validation in a construction payment case and a corresponding semiinterview, demonstrating its effectiveness and applicability in tangible business contexts. This approach catered to the specific needs of construction management, emphasizing standardization, interoperability, automation, and visualization.

While this paper has made significant strides in addressing critical challenges associated with the collaboration and automation of construction BPs, several challenges remain to be addressed in future research endeavors. Specifically, the paper has ventured into the innovative realm of realizing SC-enabled RPA modules, delving comprehensively into a pay RPA action with detailed algorithm design and automatic execution results via blockchain. However, there is room for future studies to extend this approach for a broader exploration of other SC-enabled RPA actions. Moreover, as initial technical guidance for developing SC generation and visualization methods, the present implementation accounts for limited scenarios, wherein merely BPMN process models and Solidity SCs were used. In future construction BP research, these aspects will be gradually strengthened and further developed.

Appendix I. Information of Interviewees

ID	Organization	Position	Expertise	Work experience	Education
OW1	Owner	Process Manager	Construction supply chain process	10	Bachelor's degree
MC1	Main Contractor	Project Manager	Construction contracting and cost	7	Bachelor's degree
MC2	Main Contractor	Project Engineer	Construction process and cost	10	Master's degree
UN1	Research Institute	Senior Researcher	Construction management	4	Doctorate
IT1	IT Software	Software Engineer	Software backend development	7	Master's degree
GE1	Geotechnical Engineering	Project Engineer	Construction planning process optimization and software development	8	Master's degree

Appendix I. (Continued.)

ID	Organization	Position	Expertise	Work experience	Education
IT2	IT Security	Application security researcher	Researching and coding in the application security aspect	8	Master's degree
EC1	Construction engineering consulting	Cost Engineer	Project cost and quality check	5	Bachelor's degree
ED1	Construction engineering digitalization	BIM manager and Data scientist	BIM software development and machine learning research	3	Master's degree

Appendix II. Interview Questions

Question 1: What are the advantages of the proposed framework?

Question 2: What are the disadvantages of the proposed framework?

Question 3: Would you use the prototype system implemented within this framework? Why/Why not?

Question 4: Do you have other suggestions to improve the prototype system and the framework?

Appendix III. Responses from Interviewees

Interviewee	Response	Response classification
Question 1: What are the advantages of the proposed framework?		
OW1	It is useful for data integrity; It is good to use actual scenarios; Status tracking is important for the supply chain process, and data storing and searching are important for the claim process; Using blockchain can reduce doubts; Visualization is important for managing requirements in process collaboration; Processes are in general complex, it can simplify many management problems; All the collected data could be useful for future risk management.	a1 Data integrity a2 Good for an actual scenario a3 Data traceability a4 Reduce doubts a5 Visualization a6 Simplify management problems a7 Future risk management
MC1	It could be useful in some processes for providing evidence; The overall idea can provide some insights for an integrated platform.	a3 Data traceability a8 Insights for an integrated platform
MC2	It is okay to use the framework for the DAP process; Construction BP management urgently needs digitalized management.	a2 Good for actual scenario a9 Digitalized management
UN1	It can provide technical value.	a10 Technical value
IT1	It is useful to provide traceability and completed process status; Such an integrated platform can be valuable.	a3 Data traceability a8 Insights for an integrated platform
GE1	It is good to use BPMN for generating smart contract codes because BPMN is easier to be understood if you haven't learned all the notations in advance; The idea of automatically writing SC codes can simplify the use of SC; The concept is designed in a good way.	a11 Good to use BPMN for SC generation a12 Simplify SC usage a13 Good designed concept
IT2	Connecting to SC makes the BPMN execution more reality; It is useful to view the execution and process details; Transparency and historical information can ensure things are actually done.	a14 Ensured BPMN execution by SC a5 Visualization
EC1	It can reduce disputes with unchangeable historical transactions; It can improve digitalization and reduce paperwork.	a3 Data traceability a15 Reduce disputes a9 Digitalized management
ED1	It is useful to record process history for years; It is useful to link documents and record them throughout the processes; It can also be used as a general-purpose system without using BIM.	a3 Data traceability a16 Link documents throughout processes a17 Use as a general-purpose system
Question 2: What are the disadvantages of the proposed framework?		
OW1	Actual relationships are complex, it could be difficult to make an integrated system.	d1 Not easy to make an integrated system
MC1	For process automation, it is only useful when it is certain that nothing will go wrong; Every project and change could be unique, no general process for management and it could take time to generate each unique process.	d2 The process automation is not guaranteed error-free
MC2	Programmer cannot be fully reduced when customized actions are needed, and user-defined actions can increase workload; Very detailed tasks cannot be represented by BPMN; Process management should not be the only focus, because the most important thing is the handle the project right, not just process; Many details cannot be handled by this prototype system (maybe neither can the proposed framework); The framework can only smooth and speed up the process, but not automate.	d3 No one-size-fits-all solution for all processes d3 No one-size-fits-all solution for all processes d4 User-defined actions increase workload d5 Lack of flexibility for BPMN d6 Cannot handle complete project management d7 Process automation is not actually achieved
UN1	The idea is too general without a range; The problem to be solved is not specific enough; the Framework is too complex; The boundary for applications is not clear.	d3 No one-size-fits-all solution for all processes d8 The framework is too complex

Appendix III. (Continued.)

Interviewee	Response	Response classification
IT1	The interviewee is concerned about the level of detail when using BPMN for different scales of projects; It could be challenging when BPMN is changed during the project process.	d5 Lack of flexibility for BPMN
GE1	Everyone must use BPMN; Not everything will be executed as planned.	d5 Lack of flexibility for BPMN
IT2	The interviewee is concerned about data privacy when using public blockchains.	d9 Data privacy issue of public blockchain
EC1	The interviewee is concerned about the detail degree of the BPMN design (might not be detailed enough for quality checking).	d5 Lack of flexibility for BPMN
ED1	In a big project, it could be difficult to define the processes, because they can be very complex with rapid changes.	d10 Difficult to define processes for the system
Question 3: Would you use the prototype system implemented within this framework? Why/Why not?		
OW1	Many problems can be solved.	Yes
MC1	The interviewee will only use it if it can prove to significantly save time and effort.	No
MC2	For some specific processes (e.g., the DAP process), the framework can be useful.	Conditionally
UN1	The interviewee might use it for solving a specific problem.	Conditionally
IT1	It is good to use SC for traceability.	Yes
GE1	The framework can be useful for recording.	Yes
IT2	The framework cannot be used at work, but the interviewee will use it if it fits the situation.	Conditionally
EC1	The interviewee is very willing to advertise and use this system.	Yes
ED1	The system seems interesting! The interviewee is willing to use it and test how well it works.	Yes
Question 4: Do you have other suggestions to improve the prototype system and the framework?		
OW1	Should consider flexibility and interoperability.	s1 Consider interoperability s2 Consider the flexibility of the process
MC1	Adding a function for decision making could be useful, especially notifying before making a wrong decision; Should consider reusability, availability, ease of use, and improved efficiency.	s3 Add function for decision making s4 Consider reusability, availability, ease of use, improved efficiency
MC2	BPMN should only be used for logic design and flexible; The framework should focus on a specific construction BP for specific problem solving and should consider how to integrate all data together, data like construction progress, schedule, and requirements; The framework should use the least effort to manage the data.	s2 Consider flexibility of process s5 Consider a more specific problem s6 Consider data accessibility and integration
UN1	Compare existing solutions with the proposed solution; Point out the current value; Mind your definition: Everyone might have a different understanding of process management.	s7 Compare with existing solutions s8 Define process management by constraints
IT1	The platform should have the ability of flexibility (don't fully rely on BPMN); Experience from operators is important and should be considered (respect their original way of doing things); Data should be provided in more dimensions or angles (fast and better, for assisting operators); Be ready that something didn't go as expected.	s2 Consider flexibility of process s9 Consider experience from operators s6 Consider data accessibility and integration
GE1	The process should be able to quickly respond to the changes and should put efforts into making the users trust the involved technologies and the implemented software.	s2 Consider the flexibility of the process s10 Consider user trust in the technology and software
IT2	Make it smoother from designing BPMN to execution (e.g., combine the generation tool with the web application); The BIM model, the table (e.g., billing plan), and their corresponding BPMN should appear in the web application at the same time to keep consistent of the web application.	s11 Consider framework smoothness s12 Consider data consistency
EC1	Consider the smoothness of the framework; Take rapid changes of processes in mind and reduce the process changing time and efforts for the users.	s11 Consider framework smoothness s13 Reduce time and effort for changing a process
ED1	Allow modification of BPMN in the web application; Allow communication among participants in the web application; Add issue management function in the BIM viewer of the web application	s2 Consider flexibility of the process s14 Allow participants to communicate via the system s15 Add issue management function in the BIM viewer

Data Availability Statement

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

This work was supported by the National Key R&D Program of China (No. 2023YFC3804300) and the National Natural Science Foundation of China (Grant Nos. 72301068 and 52378492).

The study was conducted as part of the BIMcontracts research project funded by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the “Smart Data Economy” technology program (Project No. 01MD19006B).

References

- Abuezhayeh, S. W., L. Ruddock, and I. Shehabat. 2022. “Integration between knowledge management and business process management and its impact on the decision making process in the construction sector: A case study of Jordan.” *Constr. Innovation* 22 (4): 987–1010. <https://doi.org/10.1108/CI-02-2020-0021>.
- Adriaanse, A., H. Voordijk, and G. Dewulf. 2010. “Adoption and use of interorganizational ICT in a construction project.” *J. Constr. Eng. Manage.* 136 (9): 1003–1014. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0000201](https://doi.org/10.1061/(ASCE)CO.1943-7862.0000201).
- Ahmadisheykhsarmast, S., and R. Sonmez. 2020. “A smart contract system for security of payment of construction contracts.” *Autom. Constr.* 120 (Dec): 103401. <https://doi.org/10.1016/j.autcon.2020.103401>.
- Alaloul, W. S., A. H. Qureshi, M. A. Musarat, and S. Saad. 2021. “Evolution of close-range detection and data acquisition technologies towards automation in construction progress monitoring.” *J. Build. Eng.* 43 (Nov): 102877. <https://doi.org/10.1016/j.job.2021.102877>.
- Alreshidi, E., M. Mourshed, and Y. Rezgui. 2018. “Requirements for cloud-based BIM governance solutions to facilitate team collaboration in construction projects.” *Requir. Eng.* 23 (1): 1–31. <https://doi.org/10.1007/s00766-016-0254-6>.
- Arcadis. 2023. “2023 Global construction disputes report.” Accessed March 24, 2024. <https://www.arcadis.com/en-us/knowledge-hub/perspectives/north-america/united-states/2023/construction-disputes-report-2023>.
- Bagge, O. S., and M. Haverlaen. 2003. “Domain-specific optimisation with user-defined rules in codeboost.” *Electron. Notes Theor. Comput. Sci.* 86 (2): 119–133. [https://doi.org/10.1016/S1571-0661\(04\)80679-0](https://doi.org/10.1016/S1571-0661(04)80679-0).
- Baird, L., M. Harmon, and P. Madsen. 2020. “Hedera: A public hashgraph network & governing council.” Accessed July 10, 2024. https://hedera.com/hh_whitepaper_v2.1-20200815.pdf.
- Bodorik, P., C. G. Liu, and D. Jutla. 2023. “TABS: Transforming automatically BPMN models into blockchain smart contracts.” *Blockchain: Res. Appl.* 4 (1): 100115. <https://doi.org/10.1016/j.bcr.2022.100115>.
- Chang, W.-C., and S. Hasanzadeh. 2024. “Toward a framework for trust building between humans and robots in the construction industry: A systematic review of current research and future directions.” *J. Comput. Civ. Eng.* 38 (3): 3124001. <https://doi.org/10.1061/JCCEE5.CPENG-5656>.
- Choi, T.-M., S. Kumar, X. Yue, and H.-L. Chan. 2022. “Disruptive technologies and operations management in the industry 4.0 era and beyond.” *Prod. Oper. Manage.* 31 (1): 9–31. <https://doi.org/10.1111/poms.13622>.
- Dahanayake, K. C., and N. Sumanarathna. 2022. “IoT-BIM-based digital transformation in facilities management: A conceptual model.” *J. Facil. Manage.* 20 (3): 437–451. <https://doi.org/10.1108/JFM-10-2020-0076>.
- Das, M., X. Tao, and J. C. Cheng. 2021. “BIM security: A critical review and recommendations using encryption strategy and blockchain.” *Autom. Constr.* 126 (Jun): 103682. <https://doi.org/10.1016/j.autcon.2021.103682>.
- Di Ciccio, C., A. Cecconi, J. Mendling, D. Felix, D. Haas, D. Lilek, F. Riel, A. Rumpl, and P. Uhlig. 2018. “Blockchain-based traceability of inter-organisational business processes.” In *Proc., Int. Symp. on Business Modeling and Software Design*, 56–68. Cham, Switzerland: Springer.
- Dwivedi, V., M. Iqbal, A. Norta, and R. Matulevičius. 2023. “Evaluation of a legally binding smart-contract language for blockchain applications.” *JUCS* 29 (7): 691–717. <https://doi.org/10.3897/jucs.97112>.
- Emaminejad, N., L. Kath, and R. Akhavian. 2024. “Assessing trust in construction ai-powered collaborative robots using structural equation modeling.” *J. Comput. Civ. Eng.* 38 (3): 4024011. <https://doi.org/10.1061/JCCEE5.CPENG-5660>.
- Garcia-Garcia, J. A., N. Sanchez-Gomez, D. Lizcano, M. J. Escalona, and T. Wojdyski. 2020. “Using blockchain to improve collaborative business process management: Systematic literature review.” *IEEE Access* 8 (Aug): 142312–142336. <https://doi.org/10.1109/ACCESS.2020.3013911>.
- Garfatta, I., K. Klai, M. Graïet, and W. Gaaloul. 2021. “Model checking of solidity smart contracts adopted for business processes.” In *Proc., Int. Conf. on Service-Oriented Computing*, 116–132. Cham, Switzerland: Springer.
- Hamledari, H., and M. Fischer. 2021. “Construction payment automation using blockchain-enabled smart contracts and robotic reality capture technologies.” *Autom. Constr.* 132 (Dec): 103926. <https://doi.org/10.1016/j.autcon.2021.103926>.
- Hunhevciz, J. J., and D. M. Hall. 2020. “Do you need a blockchain in construction? Use case categories and decision framework for DLT design options.” *Adv. Eng. Inf.* 45 (Aug): 101094. <https://doi.org/10.1016/j.aei.2020.101094>.
- Ilbeigi, M., A. Morteza, and R. Ehsani. 2022. “An infrastructure-less emergency communication system: A blockchain-based framework.” *J. Comput. Civ. Eng.* 36 (2): 4021041. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0001011](https://doi.org/10.1061/(ASCE)CP.1943-5487.0001011).
- Isaja, M., et al. 2023. “A blockchain-based framework for trusted quality data sharing towards zero-defect manufacturing.” *Comput. Ind.* 146 (Apr): 103853. <https://doi.org/10.1016/j.compind.2023.103853>.
- Jiang, Y., X. Liu, K. Kang, Z. Wang, R. Y. Zhong, and G. Q. Huang. 2021. “Blockchain-enabled cyber-physical smart modular integrated construction.” *Comput. Ind.* 133 (Dec): 103553. <https://doi.org/10.1016/j.compind.2021.103553>.
- Jiang, Y., X. Liu, Z. Wang, M. Li, R. Y. Zhong, and G. Q. Huang. 2023. “Blockchain-enabled digital twin collaboration platform for fit-out operations in modular integrated construction.” *Autom. Constr.* 148 (Apr): 104747. <https://doi.org/10.1016/j.autcon.2023.104747>.
- Khan, S., M. B. Amin, A. T. Azar, and S. Aslam. 2021. “Towards interoperable blockchains: A survey on the role of smart contracts in blockchain interoperability.” *IEEE Access* 9 (Aug): 116672–116691. <https://doi.org/10.1109/ACCESS.2021.3106384>.
- Kim, E. W., M. S. Park, K. Kim, and K. J. Kim. 2022. “Blockchain-based automatic tracking and extracting construction document for claim and dispute support.” *KSCE J. Civ. Eng.* 26 (9): 3707–3724. <https://doi.org/10.1007/s12205-022-2181-z>.
- Kochovski, P., S. Gec, V. Stankovski, M. Bajec, and P. D. Drobintsev. 2019. “Trust management in a blockchain based fog computing platform with trustless smart oracles.” *Future Gener. Comput. Syst.* 101 (Dec): 747–759. <https://doi.org/10.1016/j.future.2019.07.030>.
- Köpke, J., M. Franceschetti, and J. Eder. 2019. “Balancing privacy and enforceability of BPM-based smart contracts on blockchains.” In *Proc., Int. Conf. on Business Process Management*, 87–102. Cham, Switzerland: Springer.
- Kozma, D., P. Varga, and F. Larrinaga. 2021. “Dynamic multilevel workflow management concept for industrial IoT systems.” *IEEE Trans. Autom. Sci. Eng.* 18 (3): 1354–1366. <https://doi.org/10.1109/TASE.2020.3004313>.
- Lai, H., X. Deng, and T.-Y. P. Chang. 2019. “BIM-based platform for collaborative building design and project management.” *J. Comput. Civ. Eng.* 33 (3): 5019001. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000830](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000830).
- Lee, C.-Y., H.-Y. Chong, B. L. Tanko, and M. Klufallah. 2022. “Effect between trust in communication technology and interorganizational trust in BIM-enabled projects.” *J. Constr. Eng. Manage.* 148 (8): 04022059. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0002299](https://doi.org/10.1061/(ASCE)CO.1943-7862.0002299).
- Lee, J.-K., C. M. Eastman, and Y. C. Lee. 2015. “Implementation of a BIM domain-specific language for the building environment rule and analysis.” *J. Intell. Rob. Syst.* 79 (3–4): 507–522. <https://doi.org/10.1007/s10846-014-0117-7>.
- Li, J., and M. Kassem. 2021. “Applications of distributed ledger technology (DLT) and Blockchain-enabled smart contracts in construction.” *Autom. Constr.* 132 (Dec): 103955. <https://doi.org/10.1016/j.autcon.2021.103955>.
- Li, X., H. Li, R. Zhang, Y. Yin, S. Sun, J. Bai, and R. Liu. 2023. “Impact of asymmetric trust on construction project management performance: The mediating role of knowledge sharing.” *Eng. Constr. Archit. Manage.* <https://doi.org/10.1108/ECAM-05-2022-0400>.

- Li, X., L. Wu, R. Zhao, W. Lu, and F. Xue. 2021. "Two-layer adaptive blockchain-based Supervision model for off-site modular housing production." *Comput. Ind.* 128 (Jun): 103437. <https://doi.org/10.1016/j.compind.2021.103437>.
- Liu, H., S. Han, and Z. Zhu. 2023. "Blockchain technology toward smart construction: Review and future directions." *J. Constr. Eng. Manage.* 149 (3): 3123002. <https://doi.org/10.1061/JCEMD4.COENG-11929>.
- Liu, Y., Z. Zhou, Y. Yang, and Y. Ma. 2022. "Verifying the smart contracts of the port supply chain system based on probabilistic model checking." *Systems* 10 (1): 19. <https://doi.org/10.3390/systems10010019>.
- López-Pintado, O., M. Dumas, L. García-Bañuelos, and I. Weber. 2022. "Controlled flexibility in blockchain-based collaborative business processes." *Inf. Syst.* 104 (Feb): 101622. <https://doi.org/10.1016/j.is.2020.101622>.
- Loukil, F., K. Boukadi, M. Abed, and C. Ghedira-Guegan. 2021. "Decentralized collaborative business process execution using blockchain." *World Wide Web* 24 (5): 1645–1663. <https://doi.org/10.1007/s11280-021-00901-7>.
- Lu, W., J. Lou, and L. Wu. 2023a. "Combining smart construction objects-enabled blockchain oracles and signature techniques to ensure information authentication and integrity in construction." *J. Comput. Civ. Eng.* 37 (6): 04023031. <https://doi.org/10.1061/JCCEE5.CPENG-5268>.
- Lu, W., L. Wu, and R. Zhao. 2023b. "Rebuilding trust in the construction industry: A blockchain-based deployment framework." *Int. J. Constr. Manage.* 23 (8): 1405–1416. <https://doi.org/10.1080/15623599.2021.1974683>.
- Lucic, A., and C. Blake. 2015. "Learning user-defined, domain-specific relations: A situated case study and evaluation in plant science." *Proc. Assoc. Inf. Sci. Tech.* 52 (1): 1–12. <https://doi.org/10.1002/pra2.2015.145052010033>.
- Mars, R., S. Cheikhrouhou, S. Kallel, and A. Hadj Kacem. 2023. "A survey on automation approaches of smart contract generation." *J. Supercomput.* 79 (14): 16065–16097. <https://doi.org/10.1007/s11227-023-05262-8>.
- Mashali, A., E. Elbeltagi, I. Motawa, and M. Elshikh. 2023. "Stakeholder management challenges in mega construction projects: Critical success factors." *J. Eng. Des. Technol.* 21 (2): 358–375. <https://doi.org/10.1108/JEDT-09-2021-0483>.
- Mendling, J., et al. 2018. "Blockchains for business process management—Challenges and opportunities." *ACM Trans. Manage. Inf. Syst.* 9 (1): 1–16. <https://doi.org/10.1145/3183367>.
- Milani, F., L. Garcia-Banuelos, S. Filipova, and M. Markovska. 2021. "Modelling blockchain-based business processes: a comparative analysis of BPMN vs CMMN." *Bus. Process Manage. J.* 27 (2): 638–657. <https://doi.org/10.1108/BPMJ-06-2020-0263>.
- Muhammad, I., K. Ying, M. Nithish, J. Xin, Z. Xinge, and C. C. Cheah. 2021. "Robot-assisted object detection for construction automation: Data and information-driven approach." *IEEE/ASME Trans. Mechatron.* 26 (6): 2845–2856. <https://doi.org/10.1109/TMECH.2021.3100306>.
- Oraee, M., M. R. Hosseini, D. J. Edwards, H. Li, E. Papadonikolaki, and D. Cao. 2019. "Collaboration barriers in BIM-based construction networks: A conceptual model." *Int. J. Project Manage.* 37 (6): 839–854. <https://doi.org/10.1016/j.ijproman.2019.05.004>.
- Oraskari, J., S. Törmä, M. Yalcinkaya, V. Singh, and J. Beetz. 2024. "Sharing linked building data in a peer-to-peer network: ifcOWL meets interplanetary file system." *J. Comput. Civ. Eng.* 38 (1): 04023037. <https://doi.org/10.1061/JCCEE5.CPENG-5381>.
- Pan, Y., and L. Zhang. 2021. "Automated process discovery from event logs in BIM construction projects." *Autom. Constr.* 127 (Jul): 103713. <https://doi.org/10.1016/j.autcon.2021.103713>.
- Qian, X., and E. Papadonikolaki. 2021. "Shifting trust in construction supply chains through blockchain technology." *Eng. Constr. Archit. Manage.* 28 (2): 584–602. <https://doi.org/10.1108/ECAM-12-2019-0676>.
- Shen, X., W. Li, H. Xu, X. Wang, and Z. Wang. 2023. "A reuse-oriented visual smart contract code generator for efficient development of complex multi-party interaction scenarios." *Appl. Sci.* 13 (14): 8094. <https://doi.org/10.3390/app13148094>.
- Shi, J. J., D.-E. Lee, and E. Kuruku. 2008. "Task-based modeling method for construction business process modeling and automation." *Autom. Constr.* 17 (5): 633–640. <https://doi.org/10.1016/j.autcon.2007.10.010>.
- Siek, J. G. 2010. "General purpose languages should be metalanguages." In *Proc., PEPM '10: Partial Evaluation and Program Manipulation*, edited by J. Gallagher and J. Voigtländer, 3–4. New York: ACM Special Interest Group on Programming Languages.
- Sigalov, K., X. Ye, M. König, P. Hagedorn, F. Blum, B. Severin, M. Hettmer, P. Hückinghaus, J. Wölkerling, and D. Groß. 2021. "Automated payment and contract management in the construction industry by integrating building information modeling and blockchain-based smart contracts." *Appl. Sci.* 11 (16): 7653. <https://doi.org/10.3390/app11167653>.
- Singh, V., N. Gu, and X. Wang. 2011. "A theoretical framework of a BIM-based multi-disciplinary collaboration platform." *Autom. Constr.* 20 (2): 134–144. <https://doi.org/10.1016/j.autcon.2010.09.011>.
- Sohail, M., and S. Cavill. 2008. "Accountability to prevent corruption in construction projects." *J. Constr. Eng. Manage.* 134 (9): 729–738. [https://doi.org/10.1061/\(ASCE\)0733-9364\(2008\)134:9\(729\)](https://doi.org/10.1061/(ASCE)0733-9364(2008)134:9(729)).
- Tabish, S. Z. S., and K. N. Jha. 2023. "Dispute avoidance in public construction projects." *J. Leg. Aff. Dispute Resolut. Eng. Constr.* 15 (1): 4522033. [https://doi.org/10.1061/\(ASCE\)LA.1943-4170.0000577](https://doi.org/10.1061/(ASCE)LA.1943-4170.0000577).
- Tao, X., P. K.-Y. Wong, Y. Xu, Y. Liu, X. Gong, C. Zheng, M. Das, and J. C. Cheng. 2023. "Smart contract swarm and multi-branch structure for secure and efficient BIM versioning in blockchain-aided common data environment." *Comput. Ind.* 149 (Aug): 103922. <https://doi.org/10.1016/j.compind.2023.103922>.
- Tezel, A., P. Febrero, E. Papadonikolaki, and I. Yitmen. 2021. "Insights into blockchain implementation in construction: Models for supply chain management." *J. Manage. Eng.* 37 (4): 4021038. [https://doi.org/10.1061/\(ASCE\)ME.1943-5479.0000939](https://doi.org/10.1061/(ASCE)ME.1943-5479.0000939).
- Torkanfar, N., E. R. Azar, and B. McCabe. 2023. "BidChain: A blockchain-based decentralized application for transparent and secure competitive tendering in public construction projects." *J. Constr. Eng. Manage.* 149 (7): 4023050. <https://doi.org/10.1061/JCEMD4.COENG-12449>.
- Turner, J., and R. Müller. 2003. "On the nature of the project as a temporary organization." *Int. J. Project Manage.* 21 (1): 1–8. [https://doi.org/10.1016/S0263-7863\(02\)00020-0](https://doi.org/10.1016/S0263-7863(02)00020-0).
- Viriyasitavat, W., L. Da Xu, Z. Bi, and A. Sapsomboon. 2020. "Blockchain-based business process management (BPM) framework for service composition in industry 4.0." *J. Intell. Manuf.* 31 (7): 1737–1748. <https://doi.org/10.1007/s10845-018-1422-y>.
- Viriyasitavat, W., L. Da Xu, G. Dhiman, and Z. Bi. 2022a. "Blockchain-as-a-service for business process management: Survey and challenges." *IEEE Trans. Serv. Comput.* 16 (3): 2299–2314. <https://doi.org/10.1109/TSC.2022.3199232>.
- Viriyasitavat, W., L. Da Xu, D. Niyato, Z. Bi, and D. Hoonsopon. 2022b. "Applications of blockchain in business processes: A comprehensive review." *IEEE Access* 10 (Oct): 118900–118925. <https://doi.org/10.1109/ACCESS.2022.3217794>.
- Wu, Y., and Y. Zhang. 2022. "An integrated framework for blockchain-enabled supply chain trust management towards smart manufacturing." *Adv. Eng. Inf.* 51 (Jan): 101522. <https://doi.org/10.1016/j.aei.2021.101522>.
- Yang, R., R. Wakefield, S. Lyu, S. Jayasuriya, F. Han, X. Yi, X. Yang, G. Amarasinghe, and S. Chen. 2020. "Public and private blockchain in construction business process and information integration." *Autom. Constr.* 118 (Oct): 103276. <https://doi.org/10.1016/j.autcon.2020.103276>.
- Ye, X. 2022. "Action XSD for generating smart contracts from BPMN." Accessed July 11, 2024. https://github.com/XulingYe/SmartProcess_ActionXSD.
- Ye, X. 2023. "Generate solidity smart contracts from YAWL and BPMN." Accessed July 11, 2024. https://github.com/XulingYe/SmartProcess_SmartContractGenerator.
- Ye, X., and M. König. 2021a. "Framework for automated billing in the construction industry using BIM and smart contracts." In *Proc., Int. Conf. on Computing in Civil and Building Engineering*, 824–838. Cham, Switzerland: Springer.
- Ye, X., and M. König. 2021b. "From the graphical representation to the smart contract language: A use case in the construction industry." In Vol. 38 of *Proc., Int. Symp. on Automation and Robotics in Construction*, 272–279. Dubai, UAE: International Association for Automation and Robotics in Construction.

- Ye, X., N. Zeng, and M. König. 2022a. "Systematic literature review on smart contracts in the construction industry: Potentials, benefits, and challenges." *Front. Eng. Manage.* 9 (2): 196–213. <https://doi.org/10.1007/s42524-022-0188-2>.
- Ye, X., N. Zeng, and M. König. 2022b. "Visualization of blockchain-based smart contracts for delivery, acceptance, and payment process using BIM." *IOP Conf. Ser.: Earth Environ. Sci.* 1101 (4): 42013. <https://doi.org/10.1088/1755-1315/1101/4/042013>.
- Yoon, J. H., and P. Pishdad-Bozorgi. 2022. "State-of-the-art review of blockchain-enabled construction supply chain." *J. Constr. Eng. Manage.* 148 (2): 3121008. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0002235](https://doi.org/10.1061/(ASCE)CO.1943-7862.0002235).
- Zhang, S., F. Pan, C. Wang, Y. Sun, and H. Wang. 2017. "BIM-based collaboration platform for the management of EPC projects in hydropower engineering." *J. Constr. Eng. Manage.* 143 (12): 4017087. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0001403](https://doi.org/10.1061/(ASCE)CO.1943-7862.0001403).
- Zhu, L., Y. Wu, K. Gai, and K.-K. R. Choo. 2019. "Controllable and trustworthy blockchain-based cloud data management." *Future Gener. Comput. Syst.* 91 (Feb): 527–535. <https://doi.org/10.1016/j.future.2018.09.019>.