International Journal of Production Research

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tprs20

Blockchain-enabled digital twin collaboration platform for heterogeneous socialized manufacturing resource management

Ming Li, Yelin Fu, Qiqi Chen & Ting Qu

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
https://doi.org/10.1080/00207543.2021.1966118

Taylor & Francis
Taylor & Francis Group

INDUSTRIAL INTERNET OF THINGS-ENABLED DIGITAL SERVITIZATION

Blockchain-enabled digital twin collaboration platform for heterogeneous socialized manufacturing resource management

Ming Li[a,b], Yelin Fu [a], Qiqi Chen[c] and Ting Qu[a,b]

[a]School of Intelligent Systems Science and Engineering, Jinan University (Zhuhai Campus), Zhuhai, People's Republic of China; [b]Institute of Physical Internet, Jinan University (Zhuhai Campus), Zhuhai, People's Republic of China; [c]HKU-ZIRI Lab for Physical Internet, Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong, Hong Kong SAR, People's Republic of China

ABSTRACT

Social manufacturing is conceptualised as a new type of networked manufacturing paradigm that supports the organisation of socialised manufacturing resources (SMRs) to satisfy the growth of personalised demands on crowd intelligence in a timely manner when co-creating open architecture products. The high participation of each stakeholder in this type of product production places a higher requirement on sufficient collaboration among social, cyber and physical spaces. However, under a decentralised social manufacturing network, the management of SMRs has encountered some real-life challenges in terms of their distributed and heterogeneous features. Hence, this paper proposes a blockchain-enabled digital twin collaboration platform (BcDTCP) as an integrated solution to address these challenges. A hybrid domain-driven design method is presented to design and implement business-knowledge driven systems. To address the heterogeneity of SMRs, a ubiquitous object structure is designed to flexibly adjust the functionality of the digital twin. Blockchain is introduced to construct a peer-to-peer network to organise the SMRs in a decentralised manner. Additionally, a timed coloured Petri net-based workflow is adopted to formulise the collaboration logic into a smart contract executed on the blockchain. Finally, a demonstrative case study is conducted to verify and evaluate the proposed BcDTCP under a 3D printing scenario.

ARTICLE HISTORY
Received 27 March 2021
Accepted 3 August 2021

KEYWORDS
Blockchain; digital twin; social manufacturing; domain driven design; resource collaboration

1. Introduction

Social manufacturing has been conceptualised as a new type of networked manufacturing paradigm that is established on the self-organisation and self-configuration of distributed and socialised manufacturing resources (SMRs) and enabled by social media-driven interaction, large-scale collaboration and sharing (Jiang, Ding, and Leng 2016). It is regarded as the next frontier in the manufacturing industry, and aims to make rational and sufficient use of SMRs under the circumstance of mass individualisation with diverse customer requirements and dynamic market trends through cross-enterprise collaborative manufacturing. Social manufacturing can also be regarded as a key enabler for universal manufacturing (Kusiak 2021), from the organisation of manufacturing resources perspective, since social manufacturing absorbs and transforms SMRs to enable manufacturing-as-a-service. In social manufacturing, SMR owners, such as micro-/small-/medium-/large-sized manufacturers, personal workshops, logistics and warehouse service providers, constitute the social manufacturing community (SMC) to collaborate for manufacturing tasks, especially for emerging tasks for personalised or innovative products. The loose structure of SMC makes it flexible for organising SMRs to cope with dynamic manufacturing circumstances. Additionally, it has also brought more collaboration challenges with the decentralised distribution of SMRs to execute manufacturing tasks.

Current studies have focused more on addressing collaboration issues between social space and cyber space in order to achieve collaboration at the business level (Leng and Jiang 2016). Multidimensional collaboration among physical, cyber, and social space has not yet been formed in industry. Especially in physical space, there is a lack of effective integration, ubiquitous deployment, and autonomous collaboration for SMRs. In addition, the deep participation of demanders in social manufacturing has brought more possibilities for collaborative innovation and improvement in product design, production, assembly and other aspects (Xiong et al. 2017), which accordingly increases the cost for product verification

CONTACT Yelin Fu msylfu@gmail.com School of Intelligent Systems Science and Engineering, Jinan University (Zhuhai Campus), 519070 Zhuhai, Guangdong, People's Republic of China

and evaluation in physical space. Therefore, driven by the collaboration between social and cyber spaces, there must be an effective collaboration among physical and cyber spaces to realise the necessary simulation, optimisation, computation, and experimentation based on the physical SMRs, which finally achieves full collaboration among physical, cyber and social spaces. However, according to our investigations of social manufacturing companies, collaboration between physical space and cyber space has encountered more real-life problems, which could be embodied in the following two aspects:

- SMRs have a large gap in terms of their intelligence level. The insufficient intelligence of SMRs may lead to a weak collaboration ability between the upstream and downstream of the social manufacturing chain and influence information traceability. Hence, it is difficult to trace the root of a problem for a product across its lifecycle, which may result from individual problems within the design, production and assembly stage, or connection problems between these stages.
- SMRs from different owners may have heterogeneity in terms of communication standards, protocols, interfaces, data structures, etc. Thus, system integration is usually inevitable considering interoperability and compatibility during manufacturing collaboration, which is both time-consuming and expensive.

Digital twins have emerged as an effective method for establishing the connection between cyber space and physical space (Cheng et al. 2020). It creates a dynamic virtual model based on the physical entity concerning multi-dimension, multi-spatial–temporal scale, multi-discipline and multi-physical quantities in a digital way in order to depict and simulate its properties, behaviours and rules in the physical environment to realise the mapping, interaction and fusion between cyber and physical space. This has been widely applied in smart factories and shop floors to settle collaboration matters (Tao, Qi, et al. 2019). Hence, considering expanding its application scale to social manufacturing, SMRs could be promoted into digital twins. Then, digital twins act as the core link to realise multi-dimensional collaboration among physical, cyber and social spaces, which could be a feasible solution to address the current problems. However, some underlying problems hinder the frequent adoption of digital twins in social manufacturing. Two research questions that are more concerned about the industry have been extracted and summarised as follows:

- How can digital twins be ubiquitously implemented at the structural level so that they can be dynamically and flexibly configured concerning the heterogeneity of SMRs?
- How can digital twins be managed and coordinated for collaborative manufacturing task execution under a decentralised and dynamic social manufacturing network (SMN)?

To address the above questions, this paper has proposed a blockchain-enabled digital twin collaboration platform (BcDTCP) as an integrated solution to achieve decentralised cyber-physical collaborative execution in social manufacturing. A hybrid domain-driven design (HDDD) paradigm is proposed to analyse the problem domain from multiple perspectives, creating a corresponding abstract model that can be implemented in a particular set of enabling technologies. The research questions in this problem domain are systematically resolved in the knowledge domain from three views: object, network, and workflow, which are wrapped by the bounded context.

The remainder of this paper is organised as follows. Section 2 reviews the relevant studies on social manufacturing and the state-of-the-art digital twin and blockchain technology. Section 3 specifies the HDDD paradigm to conduct business-knowledge driven complicated system design and implementation. Then, a conceptual framework of the BcDTCP is designed in section 4, identifying specific components of HDDD. Section 5 illustrates the key enabling methods in the knowledge domain to address the involved research questions. A demonstrative case study is conducted in section 6 to verify and evaluate the proposed BcDTCP. Finally, the conclusion and future works are given in section 7.

2. Literature Review

2.1. Social manufacturing collaboration

In social manufacturing, a series of interaction relations need to be established between participants, such as end users, manufacturers, suppliers, and service providers, in order to achieve efficient collaboration. However, interactive relationships have the characteristics of socialisation, complexity, dynamics and diversity, so the analysis and management of interactive relationships have become the key to the effective implementation of social manufacturing. Current studies on social manufacturing collaboration focus more on one or two dimensions between social and cyber spaces. For cyber collaboration, Shang et al. (2019) adopted analytic hierarchy process (AHP) and fuzzy evaluation methods to integrate qualitative and quantitative indicators in order to analyse supplier selection and collaboration in social manufacturing. Leng et al. (2019) proposed a blockchain-based construction method for self-organised SMCs and developed a prototype system to achieve cyber collaboration between producers and consumers.

For social-cyber collaboration, Leng and Jiang (2016) adopted a semi-supervised machine learning method to mine the implicit patterns and relationships of the participants in social manufacturing based on their social history and interaction context to facilitate supply and demand matching. In addition, the frequent interaction of social manufacturing has brought a growing breadth and depth of coordination and collaboration among participants. W. Zhang et al. (2013) combined social networks and collaborative filtering techniques in a novel framework in order to assess the quality of manufacturing services for personalised production selection. Zheng et al. (2019) proposed a new product development paradigm, called SCOAP, to enhance open innovation collaboration throughout the lifecycle for personalisation concerns.

To further extend social and cyber collaboration to physical space, F.-Y. Wang (2010) proposed a novel cyber-physical-social system (CPSS) considering information sharing, physical interaction and the social behaviour of participants in social manufacturing collaboration and analysed the influence of social context on manufacturing collaboration. CPSS provides a new solution from a technical aspect to address the overall collaboration in social manufacturing. Further implementation of the CPSS will be certain to integrate heterogeneous socialised manufacturing resources. Thus, addressing the heterogeneity of SMRs and establishing the resource-level link between cyber space and physical space is worth studying.

2.2. Digital twin technology

The concept of a digital twin originated from the information mirroring model proposed by Grieves (2005). It has been formally defined as an integrated multi-physics, multiscale, probabilistic simulation of an as-built system that uses the best available physical models, sensor updates, behaviour history, etc., to mirror the life of its corresponding flying twin (Ding et al. 2019). As a key enabler to implement cyber physical system (CPS) in industry 4.0, digital twin has been more concerned recently (Ivanov et al. 2021). Traditional studies on the digital twin focus on important asset management (Zheng and Sivabalan 2020). For example, GE used a digital twin for real-time aircraft engine monitoring and simulation in order to optimise predictive maintenance and improve inspection timeliness (Lim, Zheng, and Chen 2019). Lu et al. (2020) systemically analysed existing technologies for digital twin development based on a reference model. Recent research has expanded the target object of the digital twin, from important assets to more types of smart objects. For example, the research team from Beihang University is committed to studying digital twin-enabled frameworks and methods for collaborative product design, production and service, with specific digital twin application case studies on bicycle design, smart shop floors and maintenance service of complex equipment, respectively (Tao, Zhang, et al. 2018; Tao, Qi, et al. 2018; Tao, Sui, et al. 2019). H. Zhang et al. (2017) proposed a digital twin-based approach for designing and adjusting customised production lines and then implemented the approach with a hollow glass production line for verification and evaluation under a multi-objective optimisation scenario. However, to further expand the application scope of digital twins to various types of SMRs, the heterogeneity problem cannot be avoided when we establish and develop digital twins for SMRs, which is currently a problem that needs to be solved.

2.3. Blockchain technology

Blockchain is a novel decentralised infrastructure and distributed computing paradigm that takes a chained data structure for verification and storage and uses distributed consensus algorithms to generate and update data (Leng et al. 2020). It adapts cryptology methods to protect data transmission and access and applies automated script-based smart contracts to operate rules and data (Wang et al. 2019). It was first proposed by Nakamoto (2008), and is a type of decentralised ledger widely used in initial coin offerings, e.g. Bitcoin. The successful applications of blockchain in financial field facilitate its promising innovations in manufacturing (Zheng et al. 2021) and supply chain management (Rodríguez-Espíndola et al. 2020). Especially, the decentralisation feature of blockchain has attracted more attention from researchers to address the traditional issues arising from centralisation, which always puzzles manufacturing industry (Leng, Yan, et al. 2019). For example, Z. Li et al. (2019) proposed a public blockchain service to realise secured knowledge sharing of an injection mold redesign, which decreases the dependency on centralised authorities for knowledge exchange. Chunxia Yu et al. (2020) presented a blockchain-based cloud manufacturing architecture to enhance information transparency and decentralisation. In addition to the decentralisation feature, the other benefits of blockchain in manufacturing have also been explored by researchers. Z. Li, Barenji, and Huang (2018) designed and developed a decentralised cloud manufacturing system based on a consortium blockchain to improve its security and scalability. M. Li et al. (2021) presented a blockchain-based digital twin sharing framework to facilitate the knowledge transfer and migration among decentralised social manufacturing networks. Liu et al. (2020) proposed an industrial blockchain-based framework to improve interoperability and cooperation between stakeholders in the entire product lifecycle management. Helo and Shamsuzzoha (2020) combined IoT technology with blockchain to improve real-time visibility and traceability for project deliveries under a multi-company project environment. To address the trust problem for resource sharing in manufacturing and logistics, Chunyang Yu et al. (2020) proposed a blockchain-based resource sharing framework in support of the application of cyber-physical systems. M. Li, Shen, and Huang (2019) developed a blockchain-based workflow operating system for e-commerce logistics sharing among different stakeholders. Vatankhah Barenji et al. (2020) designed a trusted network based on blockchain to secure the communication for CPS and improved the Byzantine consensus algorithm. These studies provide reference insights into facilitating P2P-based resource sharing, which is also a concern in social manufacturing. However, the technological maturity of blockchain in manufacturing is in a low rank and worth increasing explorations (Núñez-Merino et al. 2020). Particularly, the heterogeneity of resources in social manufacturing brings more complexities for resource operations in smart contracts, like the collaborative execution control issue in smart contracts. Thus, more attention should be given to complicated business logic processing in smart contracts.

3. Hybrid domain driven design method

Since this study involves multiple areas and processes that require specific domain knowledge combined with a cross-domain view, domain-driven design (DDD) is adopted as the basic method to address these complexities by connecting the implementation of this solution to an evolving model of the core business concepts.

DDD has been widely applied to guide the analysis and design of complex software systems, and is based on making the domain itself the main focus of the project as well as maintaining a software model that reflects a deep understanding of the domain (Evans 2004). It provides both strategic and tactical levels of methods to identify, separate and organise domain concepts and their underlying relationships based on objects. Compared with object-oriented analysis and design, DDD focuses more on the behaviours and business rules of objects to satisfy business requirements in a definite knowledge domain. It promotes the involvement of domain experts to analyse the system demands and implement the design.

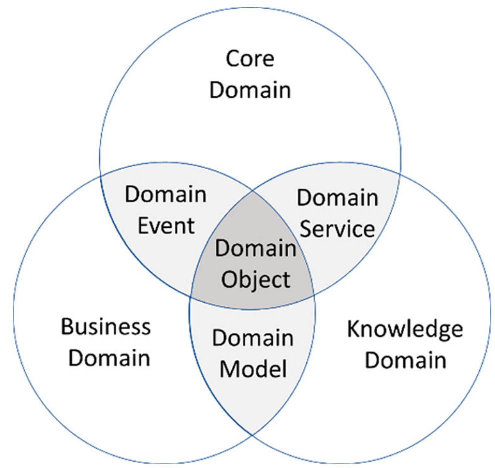


Figure 1. Hybrid domain-driven design (HDDD) paradigm

Most of the current DDD methods depend on business scope and logic to partition the domains in order to facilitate the extraction of domain knowledge for further design reuse (Rademacher, Sorgalla, and Sachweh 2018). However, these methods neglect the theoretical and technical knowledge that is potentially presented in a business domain. It is difficult for domain experts who specialised in business to identify, master, and apply these types of knowledge, especially with emerging advanced technologies or optimisation algorithms, in their domain designs. Hence, a hybrid domain-driven design (HDDD) method is proposed in Figure 1 to separate the business domain from the knowledge domain, and aims to facilitate the design and development of complex systems that are driven by both business and knowledge. In HDDD, a problem space will be divided into three domains with four overlapping domain components. Specific descriptions for each part are given as follows:

- Core Domain

The core domain is the high abstraction of feasible solutions facing a problem. It includes the near-minimum number of objects that could provide essential functionalities to address the core issues in the problem space. The core domain recognises the foundational elements behind the business and encapsulates the most valuable and stable parts of a system.

- Business Domain

The business domain refers to real-world aspects of the solution. It is used to categorise and contain a set of business objects that represent the autonomous units of the business modelled during the business analysis. The business domain depicts the business logic to satisfy the system demands and maintain the changes in the demand through the flexible organisation of business objects.

- Knowledge Domain

The knowledge domain is the knowledge collection required to implement a solution. It consists of the knowledge of a specific, specialised discipline or field, in contrast to general knowledge, or domain-independent knowledge, which is scattered across other domains. Knowledge domain extracts the critical knowledge related to defining, recognising, and solving a specific scientific or technical question, and then wraps it into a reusable and reconfigurable form for knowledge sharing, improvement, and innovation.

- Domain Object

The domain object is the most important software entity and is usually the starting point when building a system. It maintains a consistent lifecycle across all domains and dominates other objects to work collaboratively for a definite task. The domain object takes critical data and behaviour responsibilities, and represents the core competence or advantages of a solution.

- Domain Event

The domain event is a serialisable representation of a stateful fact or change that has happened within the domains. They are usually published within a bounded context when something happens inside a business domain that is also relevant for the core domain. Domain events provide an asynchronous way to share information between separate objects without establishing dedicated communication channels and acts as a driving force for business logic execution with the organisation and coordination of related objects.

- Domain Service

The domain service is simply defined as a stateless object that performs a significant behaviour or operation to the domain that does not conceptually belong to any object. It usually declares a standalone interface in order to support the operations in the core domain and instantiates in a fine-grained manner, conforming to the single responsibility principle. The domain service separates the knowledge attributes and responsibilities from the objects and achieves the servitization of knowledge.

- Domain Model

The domain model is a system of abstractions that describes the selected aspects of a sphere of knowledge, business logic, influence, or activity. It is the representation of significant real-life concepts pertinent to the domain that need to be modelled in software. The concepts include the data involved in the business and the knowledge the business uses in relation to that data.

Based on the HDDD, this study is conducted as follows: (1) Extract the core domain from the problem domain. (2) Identify the business contexts in the business domain that support the business processes of the core domain driven by the problem domain. (3) Locate the knowledge contexts in the knowledge domain that are essential for business disposal. (4) Create the domain object, model, service, and event that are associated with three domains and design the system framework. (5) Address the critical issues in each knowledge context. (6) Encapsulate the contexts in each domain into microservices and build the entire system. (7) Verify and evaluate the system.

4. Conceptual framework

Based on the HDDD, the framework of BcDTCP is designed, as shown in Figure 2. It consists of four conceptual layers.

4.1. Infrastructure layer

This layer provides generic resources that support the higher layer. Generic resources mainly contain physical SMRs, which enable both data and behaviour responsibilities. Data responsibility means that SMRs could provide real-time information concerning the state, operation, and interaction. Behavioural responsibility refers to the method, process and reaction of resources that could be performed related to manufacturing. Additionally, computing resources, storage resources and external systems are also a part of generic resources. They are the essential dependencies for platform running.

4.2. Domain layer

The domain layer is responsible for representing the concepts of business, including the business logic, rules, and knowledge. A state that reflects the business situation is controlled and maintained in this layer, and the technical details of physically capturing and recording it are delegated to the infrastructure layer. The concept of a bounded context is introduced to define the tangible boundaries for the applicability of domains. The detailed description for each domain is illustrated as follows:

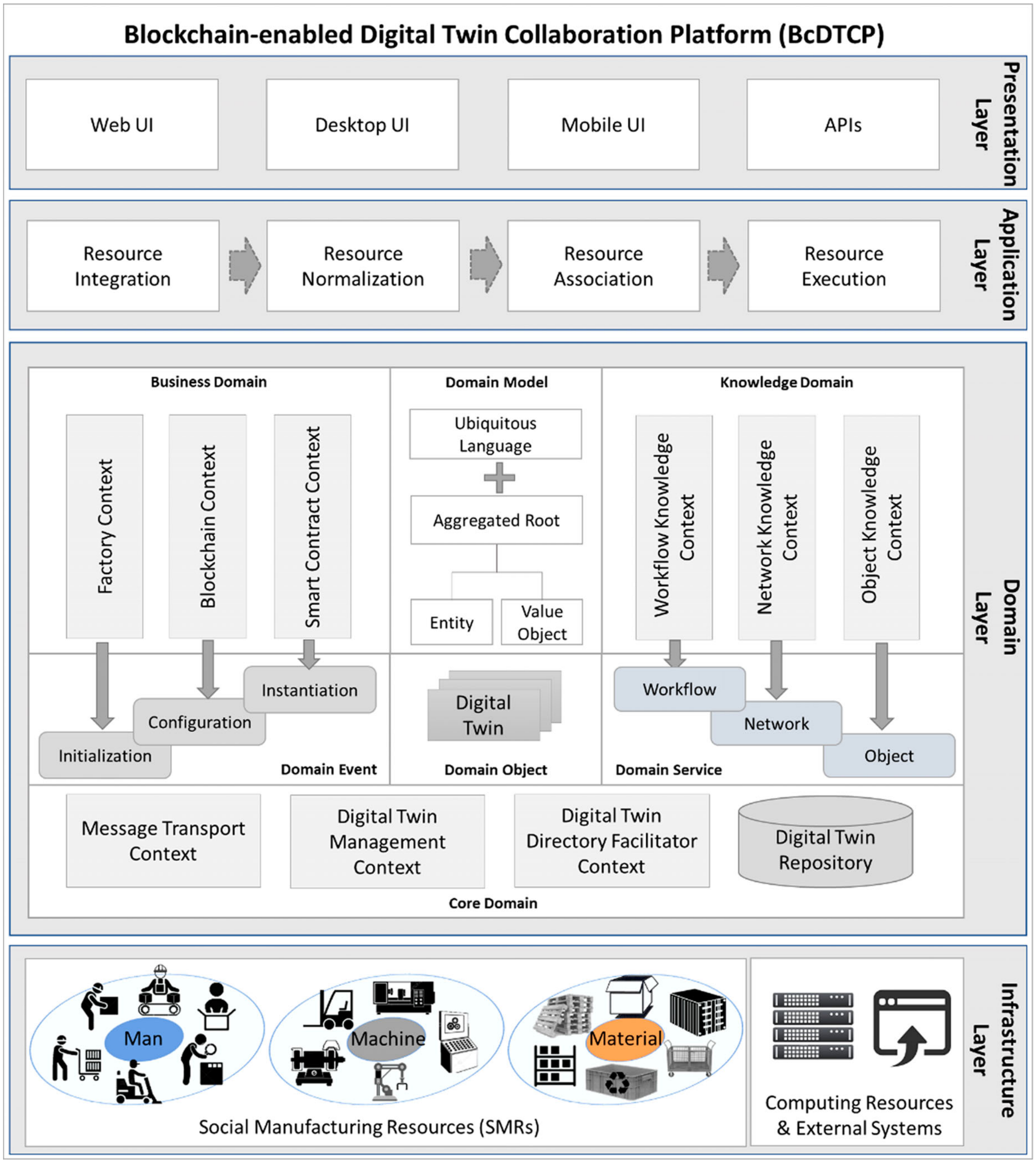


Figure 2. Blockchain-enabled digital twin collaboration platform (BcDTCP) framework

- Core Domain

The digital twin is recognised as the domain object because all businesses are associated with digital twins of different lifecycle phases. Additionally, as the development of the digital twin is usually knowledge- and technique-intensive, it could fully embody the core competence and most of the value-added part of the platform. Hence, the establishment of the core domain is around the basic functional requirements for digital twins. Three major bounded contexts are involved in this domain. The digital twin management context exerts supervisory control over access to and the use of digital twins, such as creation, registration, migration and deprecation. The digital twin directory facilitator context provides categorising, indexing, locating, and persisting services for digital twins in the repository. The message transport context enables an asynchronous communication mechanism considering the decoupling among contexts.

Three major events are designed to identify the lifecycle phases for digital twins in terms of business logic. Each event will depend on the bounded contexts in the business domain for disposal. In addition, three domain services are essential for digital twins during business disposal, which address the corresponding research issues for each lifecycle phase. These domain services are the interactive interfaces for the bounded contexts in the knowledge domain.

- Business and Knowledge Domain

The domain model is the foundation for establishing a uniform description between the business and knowledge domains. It consists of a ubiquitous language and an aggregated root. The ubiquitous language is a standardised modelling language that provides a descriptive method to help developers, users, and domain experts specify, visualise, construct, and document common specifications for the domain they are working. The aggregated root is generated through the aggregate operation for a collection of one or more related entities and possible value objects. It is responsible for maintaining the consistency of the aggregate across domains and providing consistent states for invoking. Hence, the aggregate root is usually the key object that needs to persist. An entity refers to objects that are not fundamentally defined by their attributes, but rather by a thread of continuity and identity. A value object is the abstraction of objects that have no conceptual identity, which usually describes the characteristics of a thing. The shared domain model offers a unified and interoperable method for system description and communication among stakeholders from business and knowledge perspectives, which aims to facilitate the construction and development of a complicated system.

The business domain could be further divided into three bounded contexts based on business logic. The factory content is the abstraction of generation operations to integrate SMRs as digital twins. It enables a series of construction, virtualisation and cryptographic methods behind digital twin creation, deployment, and authentication. The blockchain context constructs a peer-to-peer network for the connection of digital twins through a definite P2P protocol and establishes the transaction execution and persistence environment for smart contracts via the deployment of specific consensus mechanisms and distributed ledger storage policies. A smart contract context is the encapsulation of business logic to organise, associate with, and drive digital twins for manufacturing task execution.

The knowledge domain contains three aspects of knowledge wrapped by bounded contexts. The object knowledge context provides object structure knowledge to develop the ubiquitous object of digital twins to make them easy to configure and simple to deploy for heterogeneous SMRs. The network knowledge context is responsible for providing the network mechanism to construct the decentralised organisation and execution network for digital twins. Then the workflow knowledge context specialises in business logic modelling and descriptions for smart contracts that could be executed and verified under the above network.

4.3. Application layer

The application layer defines the jobs (use cases) that the platform is supposed to do and coordinates the domain objects to work out a problem. It follows a thin design principle, which means this layer only coordinates tasks and delegates work to collaborations of domain objects in the next layer down, without any business logic, rules or knowledge. Hence, it does not maintain the state reflecting the business situation, but it can have a state that reflects the progress of a task for the user or the programme to provide real-time visibility and traceability. This layer consists of four types of tasks. Resource integration is used to wrap SMRs into digital twins. Then, resource normalisation is responsible for converting the behaviour responsibilities of digital twins into standardised social manufacturing services. Resource association establishes the connection between social manufacturing demands and suitable services. Finally, resource execution performs the service processes with digital twin-driven SMRs.

4.4. Presentation layer

The presentation layer is designed to show information to the user and interpret the user's commands. It could be implemented for web, desktop, mobile or any presentation technology via APIs. The separation of this layer makes a radical change in the user interface have a minimal or controlled impact on the rest of the platform.

5. Knowledge domain

The knowledge domain aims to address the critical research questions separately that impede the implementation of specific bounded contexts in the business domain. The technical and theoretical details for each bounded context of the knowledge domain are illustrated in this section.

5.1. Object knowledge context

The object knowledge context aims to address the heterogeneity problem for digital twins during initialisation. The heterogeneities of SMRs may bring a large number

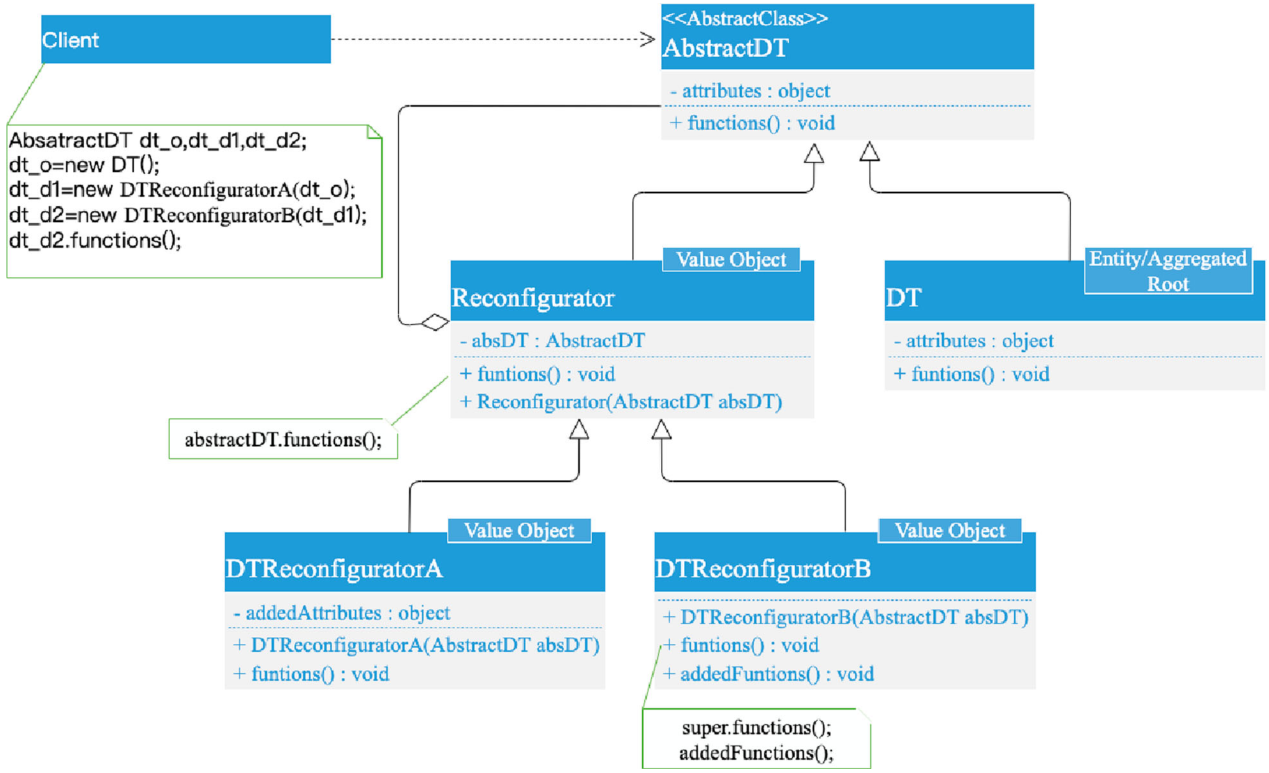


Figure 3. Ubiquitous object structure of digital twins

of independent features for digital twins in terms of attributes, functions, and runtime, and it is impractical to produce an explosion of subclasses to support every combination of features. Additionally, it is impossible to simply use object abstraction and inheritance methods for digital twins because the class of a digital twin may also be hidden or otherwise unavailable for inheritance, for example, to protect an advanced algorithm within a digital twin. Hence, a ubiquitous object structure based on the domain model is proposed, as shown in Figure 3, to make it possible to attach additional responsibilities to an individual digital twin dynamically and transparently without affecting other digital twins.

The structure consists of four types of participants. AbstractDigitalTwin defines the common interfaces for digital twins, which represent the common features for digital twins. DigitalTwin is the implementation of AbstractDigitalTwin, which acts as the runnable software object for a specific SMR, to which additional responsibilities can be attached. The Reconfigurator maintains the references to an AbstractDigitalTwin object and obtains a DigitalTwin object through its constructor. It also defines an interface that conforms to AbstractDigitalTwin's interface without concrete implementations, which invokes the same method of the injected DigitalTwins instead. Hence, the specific DTReconfigurators will inherit from the Reconfigurator and implement this interface with the additional responsibilities enclosed.

Due to the uniqueness of SMRs, DigitalTwin is identified as the entity based on the domain model, as its instantiation should be associated with a specific SMR. Reconfigurator and DTReconfigurators are defined as two types of value objects, as their responsibilities are immutable after definition. In particular, the instantiated object of DigitalTwin is also the aggregate root. The original DigitalTwin could be reconfigured with heterogeneous features that are wrapped in concrete DTReconfigurators, so that it is the carrier and the result for aggregate operation. Additionally, the application scope of an instantiated digital twin is not limited to a certain domain, and its essence of global identity also makes itself the aggregate root. Hence, as the aggregate root, DigitalTwin could persist to support lifecycle management.

5.2. Network knowledge context

The network knowledge context outputs the knowledge that is required to construct the decentralised network of digital twins. A network topology is given in Figure 4 to illustrate this context. An SMN is logically composed of channels, MSPs and SMCs. A channel is a logical subnet for communications among specific network members to conduct private and confidential business. In principle, a channel maintains one shared ledger that records the transactions that are executed with both authentication and authorisation on this channel. Since the organisation

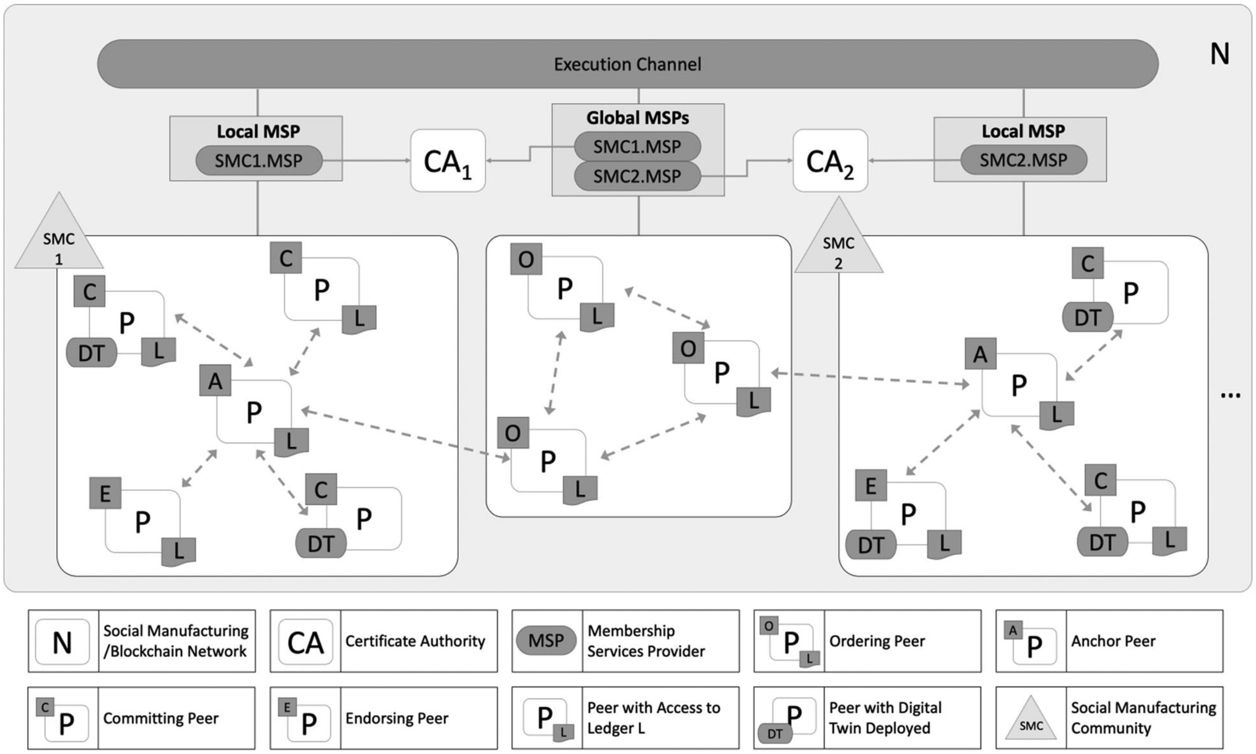


Figure 4. Network topology for digital twins

of digital twins in our study is for collaborative execution, only one execution channel is designed for this aim. Additional channels are also allowed to add in the same network for other business purposes. The MSP is the abstraction of identity operations for members to enrol in the SMN. It enables a series of cryptographic mechanisms and protocols behind issuing, validating and authenticating certificates based on certificate authorities. An SMN can be governed by one or more MSPs. Two levels of MSPs are designed considering the hierarchical structure in this network: channel MSPs and local MSPs. Each level of MSP lists roles and permissions at a particular level of administration in its corresponding hierarchy, and each SMC participating in a channel must have an MSP defined for it. In general, it is better to establish a one-to-one mapping between the SMCs and MSPs to facilitate privacy by segregating ledger data only to channel members.

There are three types of responsibilities for peers: data, behaviour and network responsibility. Data responsibility means that a peer should be capable of persisting with the local ledger. Behavioural responsibility is optional, and only required if a peer requires function extension. The digital twin could be regarded as a type of behaviour function to enable a peer to associate with specific SMRs to execute the manufacturing task. Network responsibility is necessary and can be further divided into four types. The detailed network responsibilities for each type are summarised as follows:

- Anchor peer

An anchor peer is specific to an SMC participating in a channel, and its main role is peer discovery so that peers from different SMCs can be aware of one another. Additionally, it is also responsible for receiving and broadcasting updates to the other peers in the SMC.

- Endorsing peer

An endorser peer refers to a peer that could validate and provide the seal of approval for a transaction. After the transaction is endorsed, it will be submitted to the blockchain for persistence. Smart contracts are installed on endorsing peers to make them capable of executing smart contracts and simulating the outcome of the proposed transaction for validation.

- Ordering peer

An ordering peer operates the mechanism by which the applications and peers interact with one another to ensure that every peer has the same consensus on its consistent ledger. To be more specific, it sorts the received transactions, assembles them into newly generated blocks and broadcasts these blocks to all members in the channel.

- Committing peer

Each committing peer maintains a current snapshot of the current state of the channel ledger. It performs a series of validations for ordered blocks of transactions and then commits these blocks to its replica of the channel ledger.

5.3. Workflow knowledge context

Petri nets with time and colour properties are proposed in this context to model the collaborative execution processes into formulised workflows that could be deployed in smart contracts for decentralised endorsement. To take advantage of boundedness and liveness, a Petri net-based workflow could enable a Turing-complete feature for the smart contract. Coloured tokens are adopted to facilitate generic representation for social manufacturing tasks and heterogeneous SMRs involved in a workflow.

Definition 1: The timed coloured Petri net could be formulised as a ten-tuple: $N = \{\Sigma, P, T, A, C, G, E, I, D, M_0\}$, satisfying the following requirements:

- Σ is a finite set of non-empty types, called colour sets
- P is a finite set of places, $P = \{p_1, p_2, \ldots, p_n\}$, *where n is the total number of places in a system*
- T is a finite set of transitions, $T = \{t_1, t_2, \ldots, t_m\}$, *where m is the total number of transitions in a system and* $P \cap^T = \emptyset, P \cup^T \neq \emptyset$
- A is a definite set of directed arcs satisfying $A \subseteq (P \times T) \cup^(T \times P), P \cap^A = T \cap^A \neq \emptyset$
- *C is a color function, which is defined as* $C : P \to \Sigma$ *or* $C : T \to \Sigma$
- G is a guard function, which is defined as $G : T \to Expressions$, *satisfying* $\forall t \in T, \exists[Type(G(t)) = Boolean \wedge Type(Var(G(t))) \subseteq \Sigma]$, *where* $Type(v)$*is the definite type of variable v, and* $Var(Exp)$ *is a variable set of expression Exp*
- E is the arc expression function, which is defined as $E : A \to Expressions$, *satisfying* $\forall a \in A, \exists[Type(E(a)) = C(p(a))_{MS} \wedge Type(Var(E(a))) \subseteq \Sigma]$,*where* $p(a)$ *is the place for N(a),* $C(p(a))_{MS}$ *denotes a set of all multi-sets of* $C(p(a))$
- I is the initiation function, which is defined as $I : P \to ClosedExpressions$, *satisfying* $\forall p \in P, \exists[Type(I(p)) = C(p)_{MS}]$
- D is the time-delay function, which is defined as $D : T \to R^+$, where $R^+$is the set of nonnegative real numbers. $D(t_i) = 0$ *represents that* t_i *is immediate transition, and* $D(t_i) > 0$ *represents that* t_i *is a timed transition. When a time transition is fired, the tokens will be removed immediately from the input and sent to the output after time* $D(t_i)(i = 1, 2, \ldots, m)$.
- M_0 *is the initial marking, which represents the initial number of tokens with colors in each place.*

Definition 2: The formalised definition of Σ could be further expanded as a five-tuple: $\Sigma = \{ID, F, B, i, E\}$

- ID is the only identification of a token in the system
- F is the activity feature of token. $OR \in Boolean$.The value of True or False is decided by the predecessor activity for expressing the synchronisation or asynchronization of the activity
- B is the Boolean value of a review result of tokens, $OR \in Boolean$
- i is the iteration number
- E is the set of timestamps

Definition 3: The timed coloured Petri net N is a timed coloured workflow net, iff

- $\exists p_i \in P,' p_i = \emptyset$ *(p_i is the starting place, $\cdot p_i$ is the pre-set of p_i, where i is a positive integer)*
- $\exists p_j \in P, p_j{}' = \emptyset$ *(p_j is the end place,$p_j\cdot$ is the post-set of p_j, where j is a positive integer, $i \neq j$)*
- $\forall x \in P \cup^T$, *where x is on an arbitrary path from p_i top_j*

Definition 4: *Enabling rules*: A transition t is enabled on (M_i, r) (r denotes time) with a binding b iff the following property is satisfied:

- $G(t)b = true$
- $\forall p \in P : [\sum_{(t,b) \in Y} E(p, t)b_r \leq M_i(p)]$
- $\forall e \in b : e \leq r$

Definition 5: *Fired rules*: if an enabled transition is fired on (M_i, r), the result of the firing can be defined as follows:

- $\forall p \in P : [M_{i+1}(p) = (M_i(p) - \sum_{(t,b) \in Y} E(p, t)b_r) + E(t, p)b_r]$
- $\forall e \in b : [e = r + D(t)]$

Based on these definitions, a simple workflow model of the smart contract is given in Figure 5. Different sets of colours are adopted to describe the user roles, service index, DT instances, DT commands and operation authorities. Seven places (denoted by ovals) are involved in this model, where each place has its own

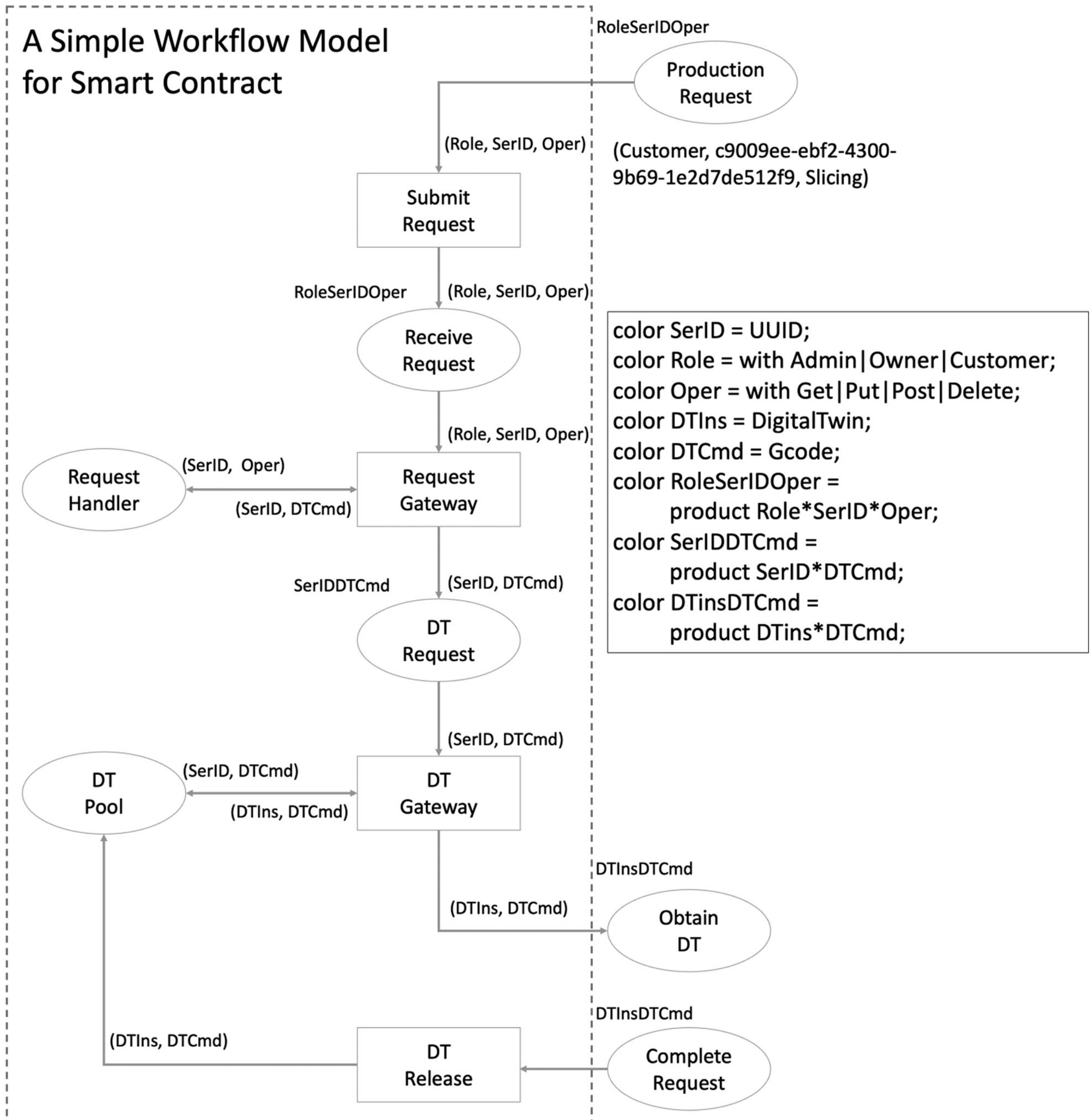


Figure 5. A simple workflow model of the smart contract

colour types and decides the corresponding data structures. During the execution of smart contracts, each place will have a number of tokens, and each token will associate and maintain its data, which conforms to the data structure of its places. For example, the place of Production Request has an initial token (Customer, ec9009ee-ebf2-4300-9b69-1e2d7de512f9, Slicing). This means that a customer placed a production order on the platform to request the slicing service with ID 'ec9009ee-ebf2-4300-9b69-1e2d7de512f9′ to perform the slice operation on its associated production file. There are five transitions (denoted by rectangles). For example, the transition 'DT Gateway' has three arcs with different arc expressions. This transition is activated when both SerID and DTCmd in its input arc expression are bound with definite values, and Definitions 4 and 5 are both satisfied.

6. A demonstrative case

A demonstrative case study is conducted in a laboratory environment under a 3D printing collaborative production scenario. First, several 3D printers at two universities are selected as a type of heterogeneous SMR to be integrated as digital twins into BcDTCP in a UPnP manner.

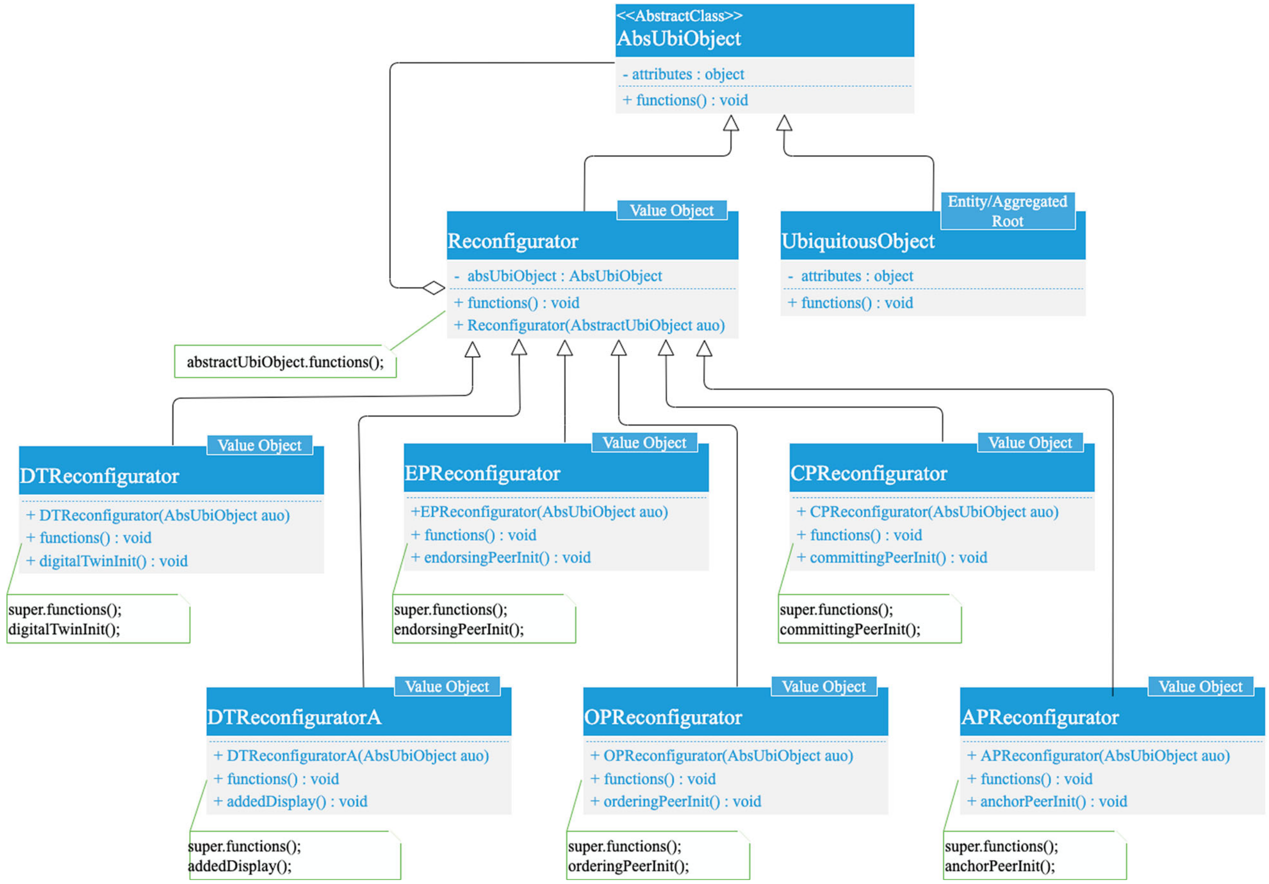


Figure 6. Simplified class diagram based on a ubiquitous object structure for peers

Second, two 3D printing clusters are built upon these 3D printers. This could simulate the gathering of nearby SMRs as a whole to enhance the overall service capacity. A cluster could compensate for the shortage where the service capacity of the single owner is limited, for example, the long time and inadequate size of a single 3D printer to perform a printing task with multiple components of a product. Third, a printing task for the shells of an industrial wearable is selected to simulate the collaborative execution among distributed 3D printers.

6.1. System implementation and demonstration

To verify and evaluate the BcDTCP using the above motivating scenario, a prototype is developed under the framework of Spring Boot Version 2.0.1. The development environment is based on JDK8, Lombok, IDEA and Gradle with necessary dependencies, including Netflix-eureka, Openfeign, Netflix-zuul, Apache Kafka, etc. Benefiting from the HDDD, each bounded context will be directly converted and developed as a microservice.

6.1.1. Object knowledge implementation

The object knowledge structure was initially proposed for digital twins. However, the generation needs of functional blockchain peers is, to some extent, similar to those of digital twins. Hence, this part merges both needs based on the expansion of the origin structure for two verification aims. One is to verify the effectiveness of addressing the heterogeneity of digital twins; the other aim is to evaluate the flexibility of the ubiquitous object structure based on the open–closed principle. The simplified class diagram is illustrated in Figure 6. The association relationship is taken to separate the heterogeneity of the environment from a ubiquitous object entity. Two factory classes are designed to further split the environmental heterogeneity from two dimensions: infrastructure and runtime. As a peer's network functions and digital twins are independent of one another, five concrete reconfigurators are designed to configure the object as required.

An open-source 3D printing host software, Pronterface, is selected as the virtual entity to realise the communication, control and management of physical printers. Another project, GCodeInfo, is taken as the core model of the digital twin to analyse the gcodes and simulate production. They are combined as a uniform digital twin and wrapped into a reconfigurator so that a digital twin for 3D printers can be constructed using this reconfigurator. Furthermore, the difference in the functions between

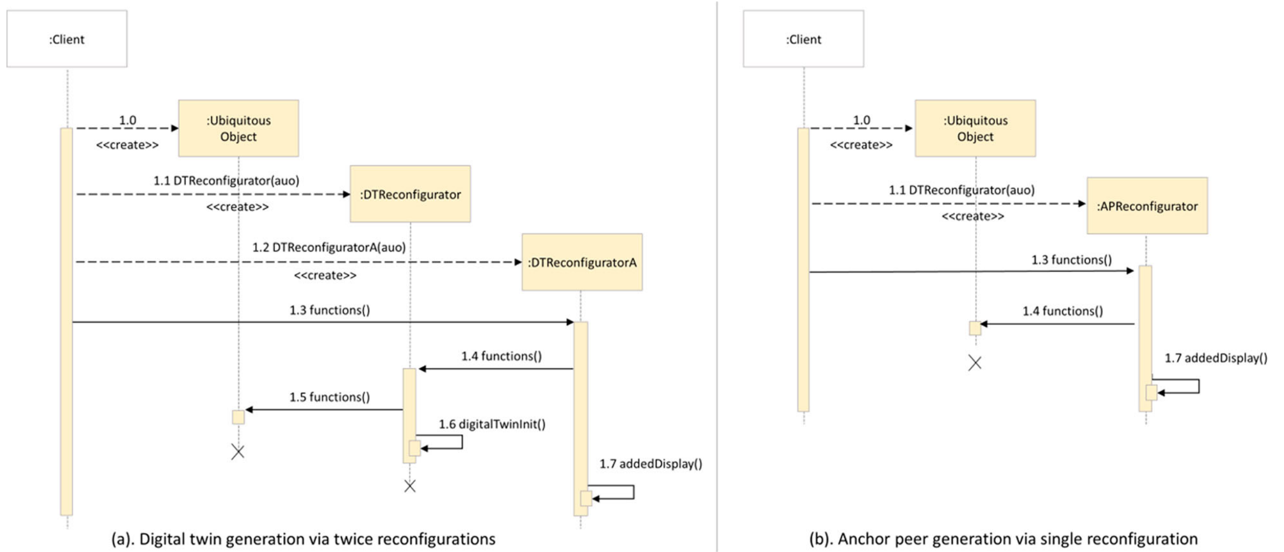


Figure 7. The sequence chart for two typical objects generation

3D printers is taken as a type of heterogeneity. Additional reconfigurators are designed for these differences to modify the functions for a specific 3D printer.

Multiple reconfigurations could be performed on a single object to dynamically update its responsibility. As the reconfigured functions may have a dependency relationship, the sequence of the calls for concrete reconfigurators needs to be strictly controlled. Hence, rule-based intelligence is adopted by Reconfigurator to manage the calls. The sequence chart for two typical objects is given in Figure 7 with their corresponding reconfiguration rules.

To bridge the communication gap between the digital twin and blockchain network, a dual-way adapter is designed and wrapped into a concrete reconfigurator to make the digital twin associate with the smart contract of endorsing peers. The extension of concrete reconfigurators is open, without an influence on the existing source code, which is fully in conformity with the open–closed principle.

6.1.2. Network knowledge implementation

The network implementation contains two phases. The first phase is the deployment of the BcDTCP platform. Several servers and Aliyun ECSs are used to deploy the microservices of BcDTCP. Two experiential SMCs located in Hong Kong and Zhuhai are taken for the second phase implementation. Necessary peers are generated through this platform for each community to construct the decentralised execution network in social manufacturing. A hyperledger fabric framework is adopted to achieve the SMN. Four MSPs are involved in this network. The first one is the BcDTCP. MSP is established on the root CA and configured to initialise the channel and genesis block. Two of these MSPs are then assigned to corresponding communities for their membership management. The last MSP is given to the group of users so that they can launch manufacturing demands in the network.

6.1.3. Workflow knowledge implementation

The workflow is implemented in a smart contract to support decentralised execution in the SMN. The overall implementation scheme follows the agent-based workflow management scheme in past studies (Huang, Huang, and Mak 2000; Zhang et al. 2010), although the specific execution process is different. The execution process is modelled into the workflow using the proposed timed coloured Petri net and built in the form of a smart contract. Then, a smart contract could be verified on the SMN for consensus, which means that the client agent could be granted permission to physically operate SMR through its digital twin. The specific workflow implementation is shown in Figure 8.

The seven steps to grant the execution permission of an SMR for a client agent are as follows:

(1) Client agent proposes a transaction proposal
The client agent generates a transaction proposal to invoke the smart contract with certain input parameters, with the intent of updating the ledger to obtain the use permission of an SMR. The transaction proposal will be packed into a properly architected format and signed with a unique signature produced by the user's cryptographic credentials.
(2) Endorsing peers execute the transaction proposal
The endorsing peers verify the received transaction proposal in terms of the format, signature, and

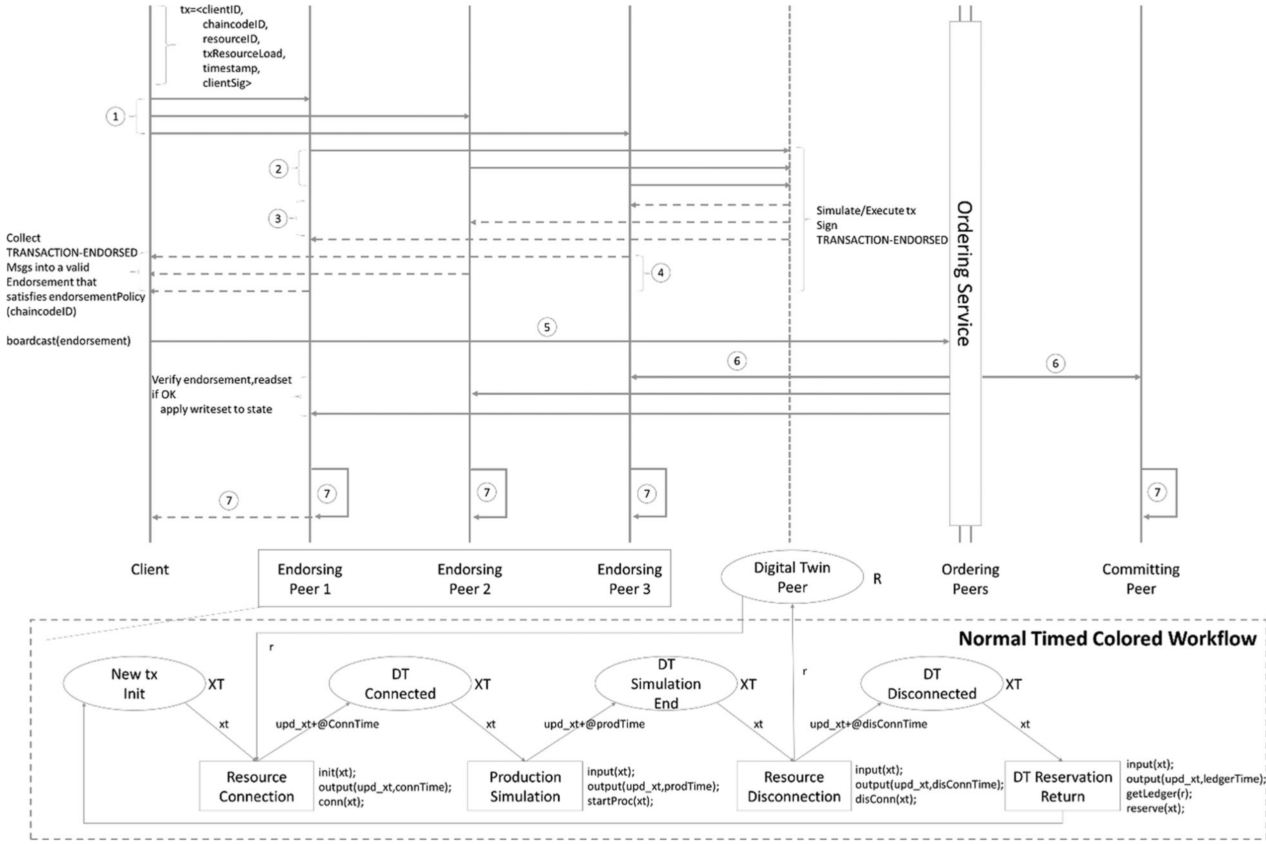


Figure 8. Workflow implementation for a smart contract

submitter's authorisation. Afterward, they take the transaction proposal inputs as arguments to invoke the smart contract. The smart contract is then executed against the current state database to produce a transaction result. This depends on the timed coloured Petri net-based workflow to invoke the SMR's digital twin for simulation.

(3) Digital twin simulates production process
The digital twin simulates the execution of the manufacturing task in the proposal and returns the production time for endorsing peers.

(4) Endorsing peers sign endorsements
The endorsing peers continue to execute the smart contract in order to generate the transaction result, which includes a response value, a read set and a write set in key/value forms to represent time slot allocation of the SMR for the submitter. In particular, the second-based simulated time will be rounded to the nearest hour to guarantee fuzzy consistency for different endorsement peers to verify the same task on the same SMR.

(5) Client agent assembles endorsements into a transaction
The client agent verifies the endorsing peer signature and compares the transaction results to determine if the results are the same. If so, the client agent will further check whether the specified endorsement policy has been fulfilled before submission (i.e. at least two endorsements collected). If passed, the client agent will assemble the transaction proposal and its endorsements into a transaction and then broadcast the transaction to the ordering peers. The ordering peers only receive transactions, order them chronologically, and create blocks of transactions without inspecting the entire content of a transaction to perform its operation.

(6) Transaction is validated and committed
The blocks of transactions are delivered to all peers on the channel. The transactions within the block are validated to ensure that the endorsement policy is fulfilled and to ensure that there have been no changes to the ledger state for read set variables, since the read set was generated by the transaction execution. Then the transactions in the block will be tagged as either valid or invalid.

(7) Transaction is executed
All committing and endorsing peers will append the block to their chains, and for each valid transaction, the write sets are committed to the current state database and an event it emitted by each peer to

Figure 9. Demonstration of the object knowledge

notify the client agent that the transaction has been approved, which means the client agent is granted permission to this SMR.

6.1.4. Demonstration

Based on the platform implementation, three user cases are taken to demonstrate the object, network, and workflow knowledge. The first user case is from the view of the owners of SMRs. They could use the platform to promote their SMRs into digital twins that could be connected in the SMN so that these digital twins act as service-oriented carriers to undertake the received production demands and associate with the physical SMRs to fulfil these demands. There are three steps to accomplish the digital twin transformation, as shown in Figure 9. (1) Define an SMR with its basic information and select the digital twin model. (2) Configure the essential parameters and settings of the digital twin model according to the SMR. (3) Encapsulate the digital twin instance under the target running environment provided by the SMR owner. The heterogeneity of printers is shown in two aspects. One is the machinal capacity that is formulised as parameters, which is mainly taken for simulation and constrains for production demands. The other is the functional differences, which depend on the object knowledge to reconfigure. The last picture in Figure 9 shows the reconfiguration of the visualisation functions for a printer with a separated fan control and a preheating function.

The second user case is the network administration. The major responsibility for this user role is to monitor the SMN statues and manage the SMCs. A dashboard is designed in Figure 10(a) to illustrate the specific

Figure 10. Demonstration of the network knowledge

information for the entire SMN. Then, on the tab page of the social manufacturing community, the administration could add, edit, and delete the nodes of the SMN, including the community and the different types of peers. The specific procedures for adding a community are demonstrated in Figure 10(b), (c), and (d).

The third user case is customers using this platform for their production work. There are four steps for the customer to perform this task, as shown in Figure 11. First, the customer should initialise a task on this platform. Related parameters for this task should be set to help match tasks and SMRs. Second, the platform will retrieve available SMRs that could be eligible to perform this task. The customer then needs to select a preferred SMR to undertake this task. Third, the customer confirms the request for this SMR, and a transaction proposal will be generated in the backend to be processed, as mentioned in the last section. Fourth, if the request is approved, the task will be released to this SMR, and visualisation for task execution will be shown in this step until the end of task execution.

6.2. Discussion

The proposed BcDTCP is tentatively verified based on a real-life scenario. Digital twins for 3D printers with heterogeneous features are built to enact transparent sharing of SMRs between the resource owner and customers.

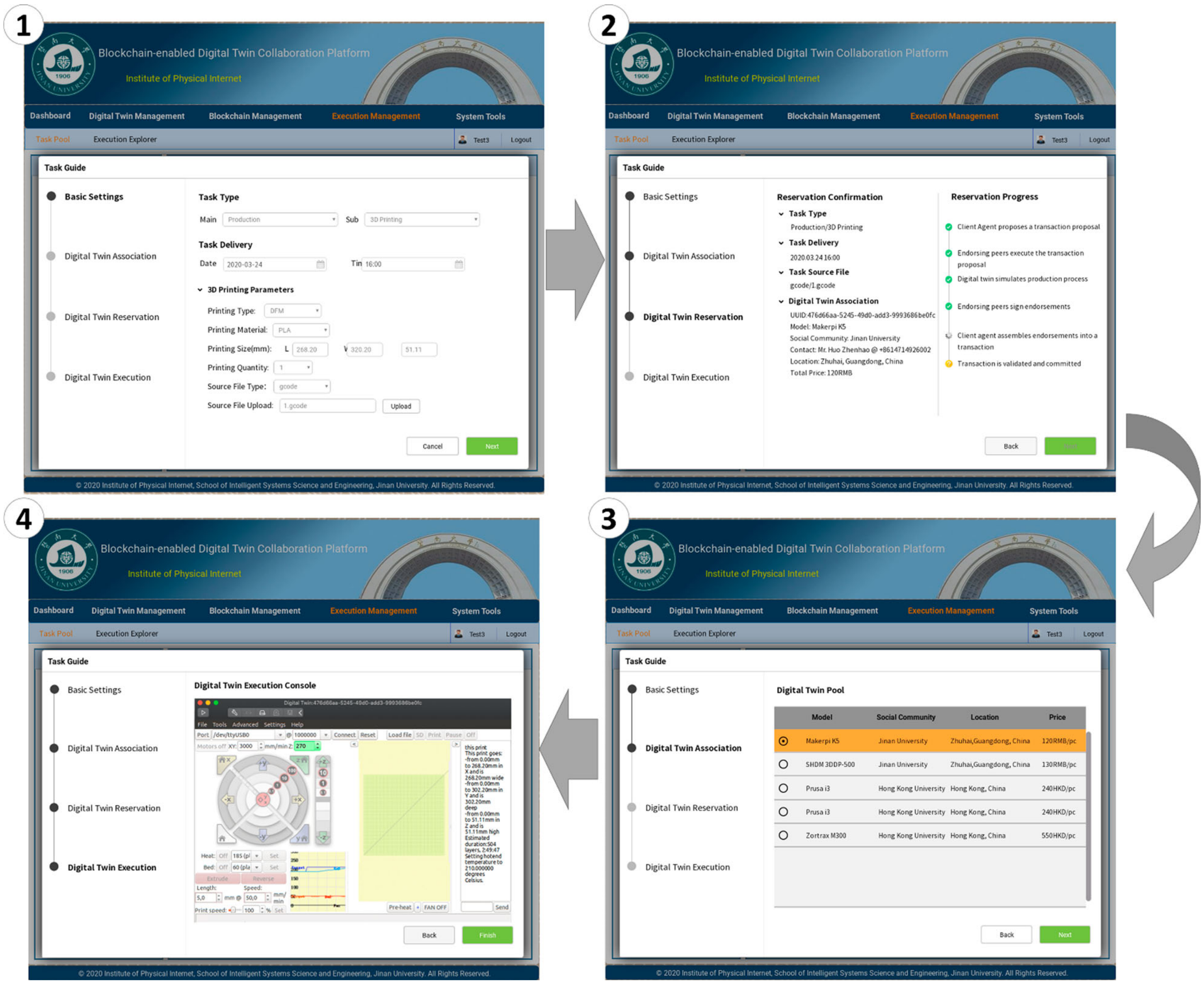


Figure 11. Demonstration of the workflow knowledge

Additionally, workflow knowledge enhances collaboration between different digital twins since a production demand may generate multiple slices, and each slice can only be executed by one printer at the same time. The simulation capacity of the digital twin is taken to evaluate the production time of different 3D printers to guarantee that the production demand can be fulfilled in time. A comparison is given in Table 1 to summarise the major differences in adopting BcDTCP in social manufacturing.

The performance for the decentralised execution is tested using Hyperledger Caliper with the basic settings listed in Table 2. Throughput (transactions/second) and transaction latency (second) are adopted as the key indicators for the performance measurement. The experimental results under the two scenarios are illustrated in Figure 12. In general, the performance of BcDTCP could satisfy the requirements of decentralised execution in social manufacturing. As the use of SMR follows the reserve-invoke rule, which decreases the dependency of

Table 1. Comparison between traditional resource organisation and BcDTCP-enabled resource organisations.

	Traditional resource organisation	BcDTCP-enabled resource organisation
Resource acquisition	Via media, advertisement, etc.	Via platform
Resource owner	Professional companies	All types, even individual resource owner
Resource integration	None, or by development	Reconfigurable Digital Twin
Organizational forms	Loose management or centralised management and execution	Decentralised management and execution
Resource visibility	Not real-time, by communication (phone, email)	Real-time by digital twin

real-time features, the second-level latency and dozens-level throughput are acceptable. Particularly, the performance for a total of 6 peers under 2 and 3 SMCs are different because the N of N endorsement policy is enabled. Compared with 2 SMCs, one more endorsement is the

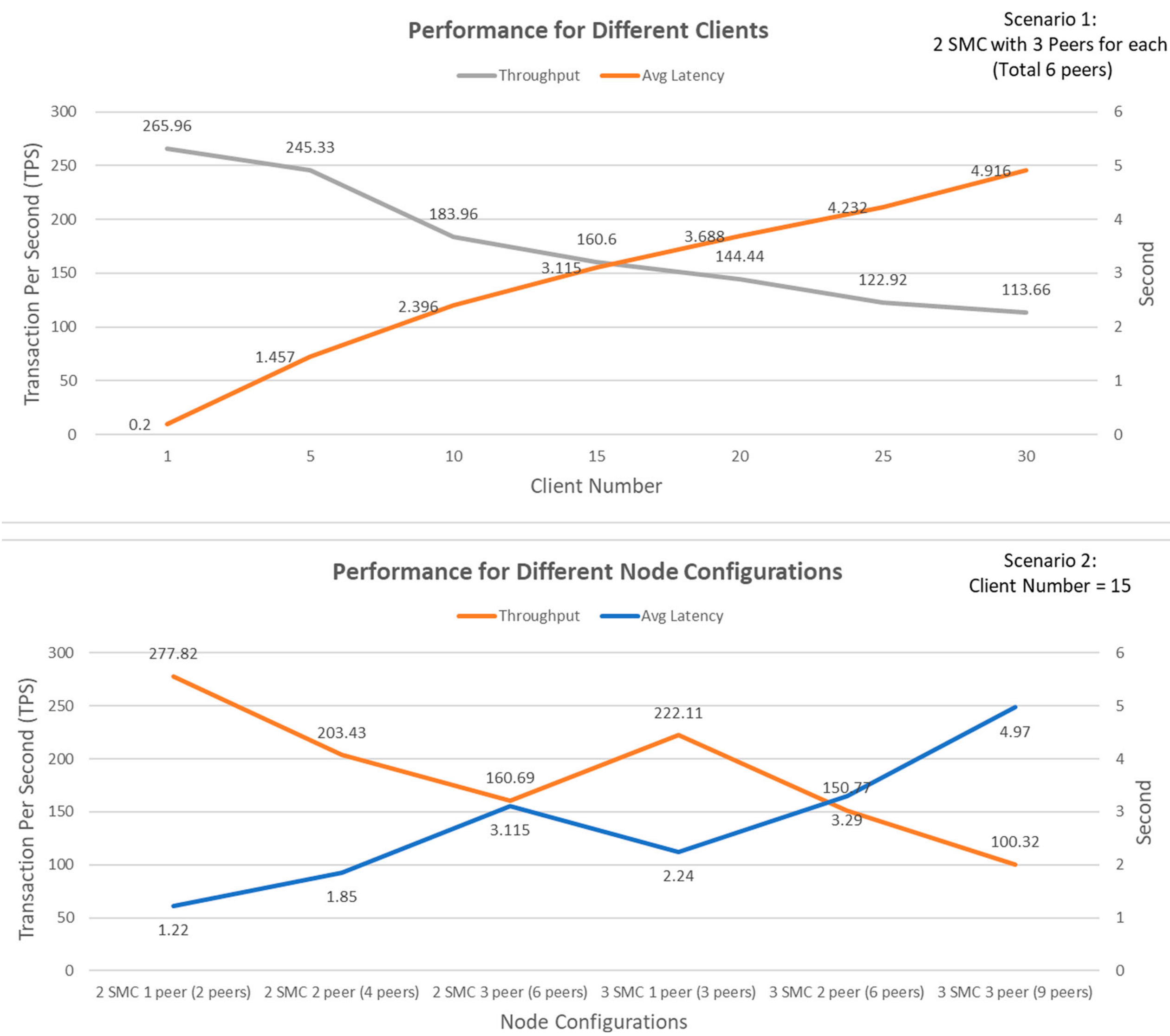


Figure 12. Performance results under two experimental scenarios

Table 2. Basic experiment settings.

Send rate	300 tps
Ordering type	Kafka
Database type	LevelDB
BatchTimeout	2 s
BatchSize: MaxMessageCount	500
BatchSize (AbsoluteMaxBytes)	10 MB
BatchSize (PerfereedMaxBytes)	512 KB
Endorsement policy	N of N[a]

[a] All endorsing peers in each SMC need to participate in approving a transaction.

requirement for 3 SMCs. Hence, in this case the expansion of SMCs should be more rigorous. It is better to add new peers to existing SMCs rather than to create a new SMC for those peers. This process also conforms to the natural generation of SMCs.

During the demonstrative case study, the key potential merits of BcDTCP are also identified. The first merit is the adoption of a digital twin for SMRs, which accelerates resource information sharing between the demander and supplier. The demander could operate remote SMRs as its local resources, such as configuring the 3D printing parameters, which could save considerable communication costs. In addition, benefiting from the simulation and optimisation abilities in the digital twin, advanced planning and scheduling mechanisms could be further applied for physical resource coordination.

The second merit is derived from the decentralised management and execution of SMRs. The platform enables a yellow page based on the decentralised management of SMRs, which follows the nature of SMNs. Additionally, decentralised execution decreases the dependency on a central organisation to conduct execution and improve transparency.

The third merit is the scalability of BcDTCP. The current prototype was only developed for research verification and evaluation. Many functions still need to be perfected for commercial use. Hence, scalability is also a necessity of BcDTCP. From the macro perspective, scalability reflects on the platform design and structure. Benefiting from HDDD, the relation between the bounded contexts is loosely coupled in terms of business logic, so that the bounded context in each layer can be easily added and removed in the form of microservices. Moreover, the scalability also manifests in the micro design of BcDTCP. The first is the reconfiguration mechanism of a ubiquitous object, which gives the digital twin the scalability for heterogeneities. The second microscalability shows the network design. The structure of the MSP makes it flexible for expanding SMCs to realise the fine-grained management of SMNs. Additionally, the execution channel could also be extended into multiple channels to maintain the additional distributed ledgers for other aims, such as sharing the self-designed digital twin models.

7. Conclusion

This paper presented the concept of BcDTCP as an overall solution to integrate heterogeneous SMRs with a decentralised execution network for social manufacturing. The primary aim of this platform is to reduce the dependency on centralised authorities for organising SMRs, which conforms to the nature of the structure in social manufacturing. Additionally, BcDTCP simplifies the integration of SMRs in a UPnP manner, which facilitates the sufficient exploitation and utilisation of idle and distributed SMRs.

The contributions of this paper are threefold. The most important contribution is the HDDD method derived from the concept of the divide-and-conquer method. It provides a problem-oriented, business-driven, and knowledge-support design method for large system construction with cross-disciplinary and multi-domain features. HDDD could be further applied to complex software system design, especially in interdisciplinary research, to facilitate the domain combination of business and knowledge.

Another contribution is the ubiquitous object structure. This presents an innovative method to resolve the heterogeneities of SMRs from the software structure aspect. This structure dynamically reconfigures the additional behaviour on a digital twin/peer, without affecting the instances from the same origins. It could be widely applied to construct more smart objects in manufacturing, logistics and other industries.

The third contribution is the timed coloured Petri-net-based smart contract. The timed coloured Petri-net workflow model enables the formalisation of complicated business logic for smart contracts. Additionally, the time and colour attributes enhance the collaboration of the dispersed digital twin in the time dimension. This could improve the adaptability of the blockchain for complex business scenarios, thus promoting wider applications for blockchain.

This research could be further extended in three aspects. First, this paper focused more on the construction of digital twins to address the heterogeneity of SMR. The current implementation of digital twins is based on open-source projects. However, most digital twins are both technology- and knowledge-intensive and may involve physical, optimisation and simulation models. The sharing of these digital twins should consider the protection of their inclusive intellectual properties. Both sharing methods and protection mechanisms need to be further explored. Second, the security and privacy issues for smart contracts are usually sensitive and a concern for commercial use. This paper only proposed and verified a novel method to model and formalise complex business logic in smart contracts. Its potential security and privacy issues should be specifically discussed before large-scale application of BcDTCP. Third, the incentive mechanism is not considered in the current implementation of BcDTCP. Apart from the peers contributed by platform users, more network responsibilities are necessary to manage and maintain the BcDTCP-enabled SMN, such as ordering peers, endorsing peers, and MSPs. A reasonable incentive mechanism should be designed to attract more participants in order to undertake these responsibilities in the near future.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (No. 52005218), the Guangdong Basic and Applied Basic Research Foundation (No. 2019A1515110296), Fundamental Research Funds for the Central Universities of China (No. 21620359) and the National Key R&D Program of China (No. 2019YFB1705401).

Disclosure statement

No potential conflict of interest was reported by the author(s).

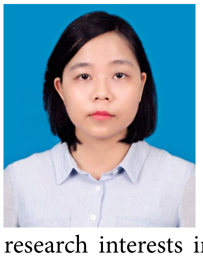
Funding

This work was supported by Natural Science Foundation of China: [Grant Number 52005218]; Natural Science Foundation of Guangdong Province, China: [Grant Number 2019A1515110296]; Fundamental Research Funds for the Central Universities of China: [Grant Number 21620359]; National Key R&D Program of China: [Grant Number 2019YFB1705401].

Notes on contributors

Dr. Ming Li is a lecturer at School of Intelligent Systems Science and Engineering, Jinan University (Zhuhai Campus). He received his PhD degree and master degree from Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong in 2018 and 2013 respectively. He also obtained two Bachelor degrees in computer science and finance respectively at South China University of Technology. Prior to joining JNU at 2019, he was a post-doctoral fellow at Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong. His research interests include multi-agent system, digital twin and blockchain technology.

Dr. Yelin Fu is an Associate Professor at School of Intelligent Systems Science and Engineering, Jinan University (Zhuhai Campus). Dr. Fu obtained two Ph.D. degrees in Management Science from City University of Hong Kong, and Management Science and Engineering from University of Science and Technology of China in 2017. His current research interests are Decision Support System, Blockchain Technology, Operations Management.



Ms. Qiqi Chen is a PhD candidate at Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong. She received his master and bachelor degree from School of Computer Science and Engineering, South China University of Technology in 2018 and 2013 respectively. Her current research interests include Cyber-Physical Systems and spare parts logistics.

Prof. Ting Qu is a full professor at School of Intelligent Systems Science and Engineering, Jinan University. He received his BEng and MPhil degrees from School of Mechanical Engineering of Xi'an Jiaotong University, and obtained PhD degree from the department of Industrial and Manufacturing Systems Engineering of The University of Hong Kong. After taking the positions of postdoctoral research fellow and research assistant professor at HKU, he was appointed as a full professor in 2010 and the department head of Industrial Engineering in 2014 at Guangdong University of Technology. In 2016, he moved to Jinan University. Prof. Qu's research interests include IoT-based smart manufacturing systems, logistics and supply chain management, and industrial product/production service systems.

ORCID

Yelin Fu http://orcid.org/0000-0003-4515-2211

References

Cheng, Jiangfeng, He Zhang, Fei Tao, and Chia-Feng Juang. 2020. "DT-II: Digital Twin Enhanced Industrial Internet Reference Framework Towards Smart Manufacturing." *Robotics and Computer-Integrated Manufacturing* 62: 101881.

Ding, Kai, Felix TS Chan, Xudong Zhang, Guanghui Zhou, and Fuqiang Zhang. 2019. "Defining a Digital Twin-Based Cyber-Physical Production System for Autonomous Manufacturing in Smart Shop Floors." *International Journal of Production Research* 57 (20): 6315–6334.

Evans, Eric. 2004. *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Boston: Addison-Wesley Professional.

Grieves, Michael. 2005. *Product Lifecycle Management: Driving the Next Generation of Lean Thinking*. New York: McGraw Hill Professional.

Helo, Petri, and A. H. M. Shamsuzzoha. 2020. "Real-Time Supply Chain—A Blockchain Architecture for Project Deliveries." *Robotics and Computer-Integrated Manufacturing* 63: 101909.

Huang, George Q, J. Huang, and Kai-Ling Mak. 2000. "Agent-Based Workflow Management in Collaborative Product Development on the Internet." *Computer-Aided Design* 32 (2): 133–144.

Ivanov, Dmitry, Christopher S Tang, Alexandre Dolgui, Daria Battini, and Ajay Das. 2021. "Researchers' Perspectives on Industry 4.0: Multi-Disciplinary Analysis and Opportunities for Operations Management." *International Journal of Production Research* 59 (7): 2055–2078.

Jiang, Pingyu, Kai Ding, and Jiewu Leng. 2016. "Towards a Cyber-Physical-Social-Connected and Service-Oriented Manufacturing Paradigm: Social Manufacturing." *Manufacturing Letters* 7: 15–21.

Kusiak, Andrew. 2021. "Universal Manufacturing: Enablers, Properties, and Models." *International Journal of Production Research*, doi:10.1080/00207543.2021.1894370.

Leng, Jiewu, and Pingyu Jiang. 2016. ""Mining and Matching Relationships from Interaction Contexts in a Social Manufacturing Paradigm." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47 (2): 276–288.

Leng, Jiewu, Pingyu Jiang, Kailin Xu, Qiang Liu, J. Leon Zhao, Yiyang Bian, and Rui Shi. 2019a. "Makerchain: A Blockchain with Chemical Signature for Self-Organizing Process in Social Manufacturing." *Journal of Cleaner Production* 234: 767–778.

Leng, Jiewu, Guolei Ruan, Pingyu Jiang, Kailin Xu, Qiang Liu, Xueliang Zhou, and Chao Liu. 2020. "Blockchain-Empowered Sustainable Manufacturing and Product Lifecycle Management in Industry 4.0: A Survey." *Renewable and Sustainable Energy Reviews* 132: 110112.

Leng, Jiewu, Douxi Yan, Qiang Liu, Kailin Xu, J. Leon Zhao, Rui Shi, Lijun Wei, Ding Zhang, and Xin Chen. 2019. "ManuChain: Combining Permissioned Blockchain with a Holistic Optimization Model as Bi-Level Intelligence for Smart Manufacturing." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50 (1): 182–192.

Li, Zhi, Ali Vatankhah Barenji, and George Q Huang. 2018. "Toward a Blockchain Cloud Manufacturing System as a Peer to Peer Distributed Network Platform." *Robotics and*

Computer-Integrated Manufacturing 54: 133–144. doi:10.1016/j.rcim.2018.05.011.
Li, Ming, Zhi Li, Xidian Huang, and Ting Qu. 2021. "Blockchain-Based Digital Twin Sharing Platform for Reconfigurable Socialized Manufacturing Resource Integration." *International Journal of Production Economics* 240: 108223.
Li, Zhi, Xinlai Liu, W. M. Wang, Ali Vatankhah Barenji, and George Q Huang. 2019. "CKshare: Secured Cloud-Based Knowledge-Sharing Blockchain for Injection Mold Redesign." *Enterprise Information Systems* 13 (1): 1–33. doi:10.1080/17517575.2018.1539774.
Li, Ming, Leidi Shen, and George Q. Huang. 2019. "Blockchain-Enabled Workflow Operating System for Logistics Resources Sharing in E-Commerce Logistics Real Estate Service." *Computers & Industrial Engineering* 135: 950–969. doi:10.1016/J.CIE.2019.07.003.
Lim, Kendrik Yan Hong, Pai Zheng, and Chun-Hsien Chen. 2019. "A State-of-the-Art Survey of Digital Twin: Techniques, Engineering Product Lifecycle Management and Business Innovation Perspectives." *Journal of Intelligent Manufacturing* 31: 1–25.
Liu, X. L., W. M. Wang, Hanyang Guo, Ali Vatankhah Barenji, Zhi Li, and George Q Huang. 2020. "Industrial Blockchain Based Framework for Product Lifecycle Management in Industry 4.0." *Robotics and Computer-Integrated Manufacturing* 63: 101897.
Lu, Yuqian, Chao Liu, I. Kevin, Kai Wang, Huiyue Huang, and Xun Xu. 2020. "Digital Twin-Driven Smart Manufacturing: Connotation, Reference Model, Applications and Research Issues." *Robotics and Computer-Integrated Manufacturing* 61: 101837.
Nakamoto, S. 2008. "Bitcoin: A Peer-to-Peer Electronic Cash System." https://bitcoin.org/bitcoin.pdf.
Núñez-Merino, Miguel, Juan Manuel Maqueira-Marín, José Moyano-Fuentes, and Pedro José Martínez-Jurado. 2020. "Information and Digital Technologies of Industry 4.0 and Lean Supply Chain Management: A Systematic Literature Review." *International Journal of Production Research* 58 (16): 5034–5061.
Rademacher, Florian, Jonas Sorgalla, and Sabine Sachweh. 2018. "Challenges of Domain-Driven Microservice Design: A Model-Driven Perspective." *IEEE Software* 35 (3): 36–43.
Rodríguez-Espíndola, Oscar, Soumyadeb Chowdhury, Ahmad Beltagui, and Pavel Albores. 2020. "The Potential of Emergent Disruptive Technologies for Humanitarian Supply Chains: The Integration of Blockchain, Artificial Intelligence and 3D Printing." *International Journal of Production Research* 58 (15): 4610–4630.
Shang, Xiuqin, Zhen Shen, Gang Xiong, Fei-Yue Wang, Sheng Liu, Timo R Nyberg, Huaiyu Wu, and Chao Guo. 2019. "Moving from Mass Customization to Social Manufacturing: A Footwear Industry Case Study." *International Journal of Computer Integrated Manufacturing* 32 (2): 194–205.
Tao, Fei, Qinglin Qi, Ang Liu, and Andrew Kusiak. 2018a. "Data-Driven Smart Manufacturing." *Journal of Manufacturing Systems* 48: 157–169.
Tao, Fei, Qinglin Qi, Lihui Wang, and A. Y. C. Nee. 2019a. "Digital Twins and Cyber–Physical Systems Toward Smart Manufacturing and Industry 4.0: Correlation and Comparison." *Engineering* 5 (4): 653–661.
Tao, Fei, Fangyuan Sui, Ang Liu, Qinglin Qi, Meng Zhang, Boyang Song, Zirong Guo, Stephen C-Y Lu, and Andrew YC Nee. 2019. "Digital Twin-Driven Product Design Framework." *International Journal of Production Research* 57 (12): 3935–3953.
Tao, Fei, Meng Zhang, Yushan Liu, and A. Y. C. Nee. 2018. "Digital Twin Driven Prognostics and Health Management for Complex Equipment." *Cirp Annals* 67 (1): 169–172.
Vatankhah Barenji, Ali, Zhi Li, Wai Ming Wang, George Q Huang, and David A Guerra-Zubiaga. 2020. "Blockchain-Based Ubiquitous Manufacturing: A Secure and Reliable Cyber-Physical System." *International Journal of Production Research* 58 (7): 2200–2221.
Wang, Fei-Yue. 2010. "The Emergence of Intelligent Enterprises: From CPS to CPSS." *IEEE Intelligent Systems* 25 (4): 85–88.
Wang, Yingli, Meita Singgih, Jingyao Wang, and Mihaela Rit. 2019. "Making Sense of Blockchain Technology: How Will It Transform Supply Chains?" *International Journal of Production Economics* 211: 221–236.
Xiong, Gang, Fei-Yue Wang, Timo R Nyberg, Xiuqin Shang, Mengchu Zhou, Zhen Shen, Shuangshuang Li, and Chao Guo. 2017. "From Mind to Products: Towards Social Manufacturing and Service." *IEEE/CAA Journal of Automatica Sinica* 5 (1): 47–57.
Yu, Chunyang, Xuanlin Jiang, Shiqiang Yu, and Cheng Yang. 2020. "Blockchain-Based Shared Manufacturing in Support of Cyber Physical Systems: Concept, Framework, and Operation." *Robotics and Computer-Integrated Manufacturing* 64: 101931.
Yu, Chunxia, Luping Zhang, Wenfan Zhao, and Sicheng Zhang. 2020. "A Blockchain-Based Service Composition Architecture in Cloud Manufacturing." *International Journal of Computer Integrated Manufacturing* 33 (7): 701–715.
Zhang, YingFeng, George Q Huang, Ting Qu, and Oscar Ho. 2010. "Agent-Based Workflow Management for RFID-Enabled Real-Time Reconfigurable Manufacturing." *International Journal of Computer Integrated Manufacturing* 23 (2): 101–112.
Zhang, Hao, Qiang Liu, Xin Chen, Ding Zhang, and Jiewu Leng. 2017. "A Digital Twin-Based Approach for Designing and Multi-Objective Optimization of Hollow Glass Production Line." *IEEE Access* 5: 26901–26911.
Zhang, W. Y., S. Zhang, Y. G. Chen, and X. W. Pan. 2013. "Combining Social Network and Collaborative Filtering for Personalised Manufacturing Service Recommendation." *International Journal of Production Research* 51 (22): 6702–6719.
Zheng, Ting, Marco Ardolino, Andrea Bacchetti, and Marco Perona. 2021. "The Applications of Industry 4.0 Technologies in Manufacturing Context: A Systematic Literature Review." *International Journal of Production Research* 59 (6): 1922–1954.
Zheng, Pai, Yuan Lin, Chun-Hsien Chen, and Xun Xu. 2019. "Smart, Connected Open Architecture Product: An IT-Driven Co-Creation Paradigm with Lifecycle Personalization Concerns." *International Journal of Production Research* 57 (8): 2571–2584.
Zheng, Pai, and Abinav Shankar Sivabalan. 2020. "A Generic Tri-Model-Based Approach for Product-Level Digital Twin Development in a Smart Manufacturing Environment." *Robotics and Computer-Integrated Manufacturing* 64: 101958.