buildings

MDPI

Article

Software Architecture and Non-Fungible Tokens for Digital Twin Decentralized Applications in the Built Environment

Benjamin Teisserenc * and Samad M. E. Sepasgozar

Faculty of Built Environment, The University of New South Wales, Sydney, NSW 2052, Australia
* Correspondence: b.teisserenc@unsw.edu.au

Abstract: Blockchain technology (BCT) can enable distributed collaboration, enhance data sharing, and automate back-end processes for digital twin (DT) decentralized applications (dApps) in the construction industry (CI) 4.0. The aim of this paper was to propose a software architecture and to develop a framework of smart contracts for blockchain-based digital twin (BCDT) dApps throughout the lifecycle of projects in CI 4.0. This paper leveraged the existing literature and action research interviews to identify and validate the critical industry problems, functional requirements (FRs), and non-functional requirements (NFRs) to be addressed by BCDT dApps in CI 4.0. Basic use cases were developed to design a framework of smart contracts for BCDT dApps throughout the lifecycle of projects. The analysis of an online survey was used to identify the key requirements and enablers to propose a software architecture for BCDT applications and to validate the requirements for developing the framework of a smart contract for BCDTs. The findings were: (1) The identification of key problems in CI 4.0 for each BIM/BCDT dimension (3D, 4D, 5D, 6D, 7D, 8D, and contractual (cD)) and the related FRs and NFRs for BCDT applications. Additionally, key use cases were designed to address the problems identified. (2) The proposed BCDT architecture permitted us to narrow gaps in the literature on blockchain-based decentralized digital twins. Moreover, the proposed BCDT architecture and smart-contract framework addressed the main requirements in the literature on BCDTs. (3) The study leveraged the non-fungible token (NFT) standard to develop a framework for smart contracts that addressed the key use cases and the related industry problems and functional requirements that were identified. The study also considered the contractual dimension (cD) as an overarching dimension in relation to the other BCDT dimensions. (4) We also compared the costs of several public blockchains for executing the proposed smart-contract framework throughout the lifecycle of a medium-sized building project. The cost analysis permitted the development of criteria to evaluate the suitability of blockchain networks for BCDT applications in CI 4.0 depending on the principal blockchain networks' properties (security, decentralization, scalability, and interoperability). Finally, this study resulted in a novel framework that included software architecture, smart-contract use cases, and selection criteria among blockchain networks for BCDT dApps in CI 4.0.

Keywords: blockchain; construction industry; decentralization; digital twin; Ethereum; Industry 4.0; Internet of Things; non-fungible token; smart contract

Citation: Teisserenc, B.; Sepasgozar, S.M.E. Software Architecture and Non-Fungible Tokens for Digital Twin Decentralized Applications in the Built Environment. *Buildings* **2022**, *12*, 1447. https://doi.org/10.3390/buildings12091447

Academic Editors: Zhen Lei, SangHyeok Han and Hexu Liu

Received: 6 June 2022
Accepted: 6 September 2022
Published: 14 September 2022

1. Introduction

The key challenges in the construction industry are currently related to low efficiency, lack of trust, lack of collaboration due to adversarial behaviors, limited information sharing, and a fragmentation of the information value chain [1]. Blockchain technology (BCT) can overcome these challenges for Industry 4.0. A blockchain is a distributed network composed of peer-to-peer (P2P) nodes that validate transactional data that are recorded into a data structure comprising successive blocks that are cryptographically linked to one another. A blockchain network is distributed; each node stores a copy of the shared ledger, making it tamper-proof. BCT can guarantee the traceability and immutability of historical transactional data and ensure data integrity and trust in these data records.

Smart contracts are programmable transactions; the code of a smart contract is executed on the blockchain in a decentralized way and provides immutable and tamper-proof states. Smart contracts enable decentralized applications (dApps) to automate business logic and processes. BCT promises to improve efficiency through automation and to enhance trust, collaboration, data sharing, and efficiency in the construction industry (CI) 4.0 [2]. More generally, the main properties of BCT, such as auditability, transparency, immutability, and decentralization, are promising for the future of smart cities [3].

BCT and smart contracts are beneficial to enhance data security, data integrity, and process automation in the CI throughout the full lifecycle of projects for fundamental aspects such as modelling and design [4], construction supply chain [5], payments and cashflows [6–8], operation and maintenance [9], sustainability [10], safety [11], and contractual aspects [12]. Cybersecurity is essential for DT in the CI [13], and BCT is particularly interesting in terms of enhancing security and data integrity for digital twins (DTs) of smart infrastructures. Indeed, a blockchain can be leveraged as the core back-end layer to enhance data security, data integrity, trust, traceability, transparency, and immutability of information, as well as in data sharing to reduce data silos [1]. Decentralized DT applications leveraging BCT have a great potential to secure information throughout the lifecycle of projects and form a decentralized digital twin cycle (DDTC) [1] that contributes to the environment and the circular economy. Digital twins gather the various type of data from the physical world, such as live and dynamic data (BMS, BAS, smart meters, and IoT sensors), static data (reports, asset registers, and O&M manuals), and geospatial data (CAD, BIM, and GIS) [14]. Blockchain-based digital twins (BCDT) can strengthen information security throughout the lifecycle of projects by recording key project data related to the seven BCDT dimensions, which are the spatial (3D), time (4D), cost (5D), maintenance (6D), sustainability (7D), safety (8D), and contractual (cD) dimensions [2]. Furthermore, BCDTs have enabled a paradigm shift for CI 4.0 that includes P2P (distributed) collaboration, processes automation with smart contracts, and data sharing in a decentralized data value chain [2]. However, the development of smart contracts for blockchain-based dApps software comes with major challenges such as the cost of deployment, performance limitations, smart contract code vulnerabilities, substantial testing, and complexity of the related software architecture [15].

It is key to identify the industry-specific problems to address for each BCDT dimension and how these problems could be addressed by BCDT dApps. Additionally, to implement BCDT dApps, it is essential to define their software architecture. Moreover, it is essential to identify the functional and non-functional requirements for BCDT dApps in relation to the industry-specific problems. Finally, a suitable framework of smart contracts is required for BCDT dApps that considers the associated costs. Hence, this paper follows four objectives: (1) Identify the problems specific to the construction industry, FRs, and NFRs for each BCDT dimension. (2) Propose a software architecture for BCDT to address the requirements identified in the first objective and to narrow the literature gaps. (3) Develop a framework for smart contracts to address the key industry problems and FRs identified in the first objective for the BCDT dimension. (4) Carry out a cost analysis and develop criteria for evaluating blockchain protocols that can be leveraged in the proposed BCDT smart-contract framework.

To achieve these objectives, the paper firstly reviews specifically related works—as discussed in Section 2—to present the main findings and technical gaps in the literature. Section 3 presents the methodology followed for this study. Section 4 presents the results of an online survey to extract the dominant requirements for the BCDT architecture and smart-contract framework. Moreover, Section 4 presents the results from semi-structured action research interviews to identify the key problems in the CI that should be addressed by BCDT applications for each BCDT dimension. Section 5 presents the findings, including the software architecture and the framework of smart contracts for BCDT dApps. Finally, Section 6 discusses the findings and carries out a performance and cost analysis of the proposed smart-contract framework in order to further discuss and evaluate the framework.

2. Related Works

A previous study on DDTC [1] revealed five main technical gaps in the integration of BCT with DTs in CI 4.0. These gaps relate to the following central themes: technical requirements of BCT, integration of IoT with BCT, integration of BIM with BCT, integration of DT data with BCT, and the complexity of CI project lifecycles and integration with the CE. As mentioned in the introduction, the key data (from CI projects) required to be recorded on the blockchain is related to the seven BCDT dimensions (3D, 4D, 5D, 6D, 7D, 8D, and cD) [2]. Moreover, the essential NFRs for BCDT applications are privacy, security, data ownership, data integrity, interoperability, and the decentralization and scalability of data storage [2]. This paper leveraged the BCDT dimension framework and the key NFRs of BCDTs for design decisions regarding the architecture and the smart contracts. Interoperability between different blockchain networks is a critical requirement so that DDTC ecosystems can exchange value between each other by leveraging interoperable protocols for the private, consortium, and public blockchain networks [1,16].

The EtherTwin study [17] proposed a DT dApp for data sharing in Industry 4.0. It leveraged access-control mechanisms, encrypted off-chain data storage for privacy, and the Ethereum public blockchain for smart contracts. This paper also leveraged off-chain data storage and public blockchains. However, due to the high cost of smart-contract deployment on Ethereum, this paper aimed to explore other blockchains to achieve a greater cost efficiency, scalability, and interoperability. The Ethereum blockchain was leveraged in another study [18] that used smart contracts in the creation of DTs of real-world assets throughout their lifecycles (design, build, test, deliver) and to manage DT ownership rights. However, the smart contract developed did not follow any particular standard, which may present limitations for wider adoption by industries. Hence, this paper aimed to use smart-contract standards such as the non-fungible token (NFT) to tokenize real-world assets, manage data ownership, and facilitate the adoption of the proposed framework through accepted standards such as the Ethereum ERC-721 NFT standard [19].

Konashevych's paper on blockchain use for real estate and property rights [20] indicated that permission and private blockchain systems will not be enablers of the blockchain paradigm shift due to their lack of decentralization and hence lack of immutability and resistance to tampering. It appears natural that the BCT paradigm shift will essentially emerge from dApps leveraging public blockchains that are decentralized, immutable, and censorship-resistant. The software architecture of blockchain applications should address key challenges around digital identities, privacy, legal compliance, and scalability [20]. The tokenization of land titles is a promising concept that could be extrapolated to all kinds of tangible or intangible values, such as intellectual property (IP), data ownership, data records, and physical assets throughout the lifecycle of CI 4.0 projects.

Hunhevicz J. et al. [21] presented a novel BCDT concept and implementation that utilizes performance-based smart contracts for a smart building digital twin [21]. The model is very promising in terms of enabling new peer-to-peer (P2P) business models based on crypto-economic incentivization related to the performance of built assets throughout their lifecycle. However, the implementation did not come without technological challenges, such as data storage on the blockchain (on-chain), and in particular, the transaction execution cost (referred to as the gas cost on the Ethereum platform) for smart contracts on the Ethereum [22] public blockchain (layer-1 blockchain). Moreover, the study exposed the security challenges that arise from coupling centralized DT solutions with decentralized blockchain networks. Hence, it is recommended to explore ways to decentralize the software stack for BCDT dApps in order to reduce single points of failure and improve cybersecurity. Moreover, it is essential to investigate cost-efficient blockchain solutions such as cheaper layer-1 blockchains or layer-2 scalability solutions to achieve sustainable gas costs throughout the lifecycle of buildings' DTs. A layer-2 scaling solution refers to a blockchain network that is running on top of an underlying layer-1 blockchain network (such as Ethereum) in order to enhance its scalability performance.

3. Method

A qualitative method was designed to address the first three research objectives and a quantitative implementation permitted us to address the fourth objective. Ultimately, the proposed methodology permitted us to address the four research questions. Two previous studies on the DDTC [1] and BCDT [2] suggested a list of critical challenges, problems, and enablers in the CI for each BCDT dimension. Then, semi-structured action research interviews were carried out to validate the key problems and identify new relevant problems when applicable. We adopted the action research overall concept: interviews were conducted to validate key FRs and NFRs related to each problem identified for all BCDT dimensions. The action research process is described in the following section. The action research interviews permitted us to validate the first objective of the study and gather data to address further objectives as explained in the following paragraphs.

Step 1—Diagnosing: this initial step aimed to diagnose and validate the main problems in the CI that BCDT could address for each of the seven dimensions. As mentioned, the initial key problems were preliminarily extracted from the findings of our previous studies on the DDTC [1] and BCDT [2]. A total of seven interviews were carried out with 6 CI experts in the field of each BCDT dimension and one expert in blockchain for the review of the BCDT technical architecture. During the interviews, the context of the research was introduced to the interviewees in order for them to understand how BCDT adoption could be beneficial for the critical CI problems identified. Tables similar to the ones shown in Section 4 were presented to the interviewees so that they could firstly approve the relevance of the CI problems identified as well as their related FRs and NFRs. The interviewees were then asked to rank the problems by order of importance and to add problems, FRs, and NFRs if needed. These elements formed the main final outputs of the interviews and are presented in the tables in Section 4. The profiles of the participants in the action research interviews are presented in Table 1.

Table 1. Summary of interviewees' backgrounds.

BCDT Dimension/Theme	Participants Region	Industry Sector	Role	Participants Experience
3D/Spatial	Australia	Design engineering	Regional manager	Senior management
4D/Time	Australia	Construction	Site supervisor	Senior
5D/Cost	Australia/Canada	Infrastructure finance	Cost planning and control	Director
6D/Maintenance	Australia	Facility management	Property manager	Senior
7D/Sustainability	Australia	Environment	Founder	Director
8D/Safety [1]	N/A	N/A	N/A	N/A
cD/Contractual	Australia	Legal	Project manager	Director/PhD
BCDT/Architecture	Australia	IT/blockchain	Researcher	PhD

[1] Note: the 8D dimension was not included in this study due to a lack of data.

Step 2—Action planning: this step aimed to design a framework for smart contracts to resolve the problems identified in Step 1. Specific use cases were designed to be further able to address the problems with the smart-contract code for BCDT dApps. The proposed use cases aimed to satisfy the FRs identified for each problem (from Step 1). Hence, a basic use case was produced for each BCDT dimension.

Step 3—Action taking: this step consisted of implementing the use cases (defined in Step 2) in the smart-contract code developed using Solidity [23]. This is known as the most widely adopted programming language for smart contracts on the Ethereum blockchain [22] and other blockchains compatible with the Ethereum Virtual Machine (EVM) [24]. Hence, the developed smart contracts aimed to address the CI problems (defined in Step 1) and their FRs. The proposed smart contracts would run as back-end components for BCDT dApps for each BCDT dimension.

Step 4—Evaluation: the smart contracts developed in Step 3 were tested using the testing framework [25]. This allowed us to evaluate the correctness of the smart contracts developed and ensure that the programmed functions satisfied the FRs (identified in the Step 1). The testing with Hardhat [25] also permitted us to extract the gas consumption for each function and, hence, measures the total execution costs of the smart contracts and carry out a cost analysis of the proposed framework. This step also included the evaluation of several EVM-compatible blockchain systems that were compared for handling the proposed smart contracts framework.

Step 5—Learning: this final step consisted of identifying essential findings from the cost analysis and discussing the results from Step 4 to propose criteria for evaluating the suitability of blockchain networks for BCDT dApps. This step also included recommendations for future research works to refine and develop the proposed BCDT framework further.

The findings from the previous study on the DDTC [1] were leveraged to produce an online survey to identify key requirements for the development of software architecture and a smart-contract framework for BCDTs. The survey statements were designed to address gaps areas in the integration of BCT with DTs, as presented in Section 2. The online survey was distributed to key stakeholders in academia and in the CI, transport, IT, blockchain, finance, legal, and real estate industries. Table 2 illustrates the survey statements' themes and the participants' profiles (industries, seniority levels, digital and blockchain experiences). A total of 103 participants answered the survey; only the survey statements required for the design of the software architecture and the framework for the smart contracts were leveraged for this paper. The analysis of the survey results led to the design of the BCDT software architecture to validate the second objective. Furthermore, the CI-specific problems and their related FRs identified through the action research interviews (Step 1) were triangulated with the answers from the online survey to develop the smart-contract framework for all BCDT dimensions and hence address Objective 3. Finally, a cost analysis was carried out by leveraging the Autodesk Revit basic sample model [26] as a use case to evaluate the gas costs of the proposed smart-contract framework throughout the asset's lifecycle; i.e., for all BCDT dimensions. The cost analysis compared the various type of public blockchains that are EVM-compatible [24].

Table 2. Summary of the survey statement themes and the data profile including regions, roles, and experience.

Statement Themes	Participant Region	Participant Industry	Participant Experience	Digital Experience	Blockchain Knowledge
Project data Trus Privacy Data erasure Decentralization Data sharing Traceability Automation Data ownership Security	Australia (60%) UK (25%) Europe (6%) Middle East (3%) US (2%) Asia (2%) India (2%)	Construction Industry (CI) (60%) Transport (8%) Academia (4%) IT (10%) Blockchain (12%) Finance (2%) Legal (4%)	Director/CEO (23%) Doctor (PhD) (13%) Senior > 10 y (33%) Junior < 10 y (26%) Graduate < 3 y (5%)	Expert (16%) Advanced (35%) Intermediate (32%) Basic (17%)	Blockchain developer (2%) Expert (12%) Advanced (14%) Intermediate (14%) Basic (40%) Nil (18%)

The following section presents the results from the action research interviews.

4. Emerging Requirements from the Action Research Interviews

As explained in the Section 3, action research interviews permitted us to identify the key problems in the CI and their related FRs and NFRs for each BCDT dimension. The findings from the interviews (Step 1) are presented in this section, which summarizes the CI problem themes, their order of importance according to the subject matter expert interviewed, and the FRs and NFRs related to each problem identified for each BCDT

dimension. The results of Step 2 are presented with the below use cases, which were designed to address the problems identified in Step 1.

Spatial context (3D): the key problem themes identified for the BCDT 3D (spatial) dimension were firstly the engineering checking and Q&A that should be completed so that checkers can validate the design independently; the related design records should be immutable, traceable, and facilitate accountability and compliance. Secondly, the site inspections and certification records should also be immutable, traceable, and facilitate accountability and compliance. Thirdly, the historical design data should be recorded and be immutable, traceable, and facilitate accountability, compliance, and privacy when required. Moreover, the system should enable sufficient storage capacity for the Big Data volume associated with the BCDT 3D spatial dimension. The fourth and last problem identified related to the requirement for recording BIM modeling changes in an immutable and traceable way while implementing accountability and privacy requirements. The system should be adequately scalable to cope with the large velocity and volume of BIM data.

The problem themes that were diagnosed permitted us to propose a use case to address the problems, FRs, and NFRs and form the outcome of the action-planning phase (Step 2). The proposed 3D use case recorded design history (importance rank 3) and BIM data (importance rank 4) with the possibility to check and approve the design from a Q&A standpoint (importance rank 1) and validate the inspection and certification processes (importance rank 2).

Time context (4D): the problems identified for the BCDT 4D (time) dimension are presented in order of importance as follows. Firstly, the construction supply chain and procurement information should be traceable so that construction goods and materials provenance can be traced and monitored in real time while maintaining regulatory compliance. These supply chain and procurement records should be immutable, traceable, and openly accessible, and the system should be sufficiently scalable for large volumes of supply-chain data. Secondly, the site inspections, installations, and certifications should be recorded periodically and approved before the packages are built. Hence, the project team should be able to audit the site conditions and the related construction logs should be recorded in an immutable and traceable way, enabling accountability and compliance. Lastly, the system should track as-built data in an immutable way, drive accountability, and enforce regulatory compliance and design compliance.

The main components selected for the development of the 4D smart-contract framework are presented as follows. The proposed 4D use case recorded supply-chain data (provenance, procurement, and delivery) (importance rank 1) and recorded approvals for inspection, installation, and certification (importance rank 2) of as-built states and their compliance (importance rank 3). It should be noted that the 3D interview revealed that the site installation should be inspected (but not approved) by the designer (3D) since there is no liability for the designer with regard to the site installation—the responsibility for the installation lies with the contractor (4D). This use case provided an acceptable framework for the 4D smart contracts to address the problems and FRs identified in the Step 1.

Cost context (5D): the problems identified for the BCDT 5D (cost) dimension are presented in order of importance as follows. Firstly, there is a requirement for an open, fair, and decentralized tendering process to avoid collusion with "prequalified" contractors. Secondly, the payment processes should be automated to enhance efficiencies, reduce bottlenecks, and improve cashflows. Automated-payment records should be traceable, immutable, and scalable, and should enable accountability and privacy when required. Thirdly, there is a requirement for decentralized project banks to improve cashflows and reduce financial bottlenecks. Currently, contractors obtain loans from banks at low interest rates to enable cashflow, but contractors can only pay subcontractors once the contractors are paid. Decentralized project banks could improve cashflow in order for everyone to stay afloat financially and would reduce payment delays. Authorized project stakeholders would be able to interact with project banks in a decentralized and transparent way.

Financial operations would be immutable, traceable, and openly auditable by key project stakeholders to enable accountability. The price values would be transparent, but the stakeholders' identities would remain private. The fourth theme is that digital assets should be leveraged by enabling the tokenization of data ownership, information, physical assets, and IP to enable incentivization mechanisms and exchange of value on decentralized marketplaces. As such, the value emanating from ownership of BCDT assets (physical, digital, or intangible) can be exchanged in decentralized, immutable, and traceable ways to enhance trust and data integrity. The fifth problem theme was that there is a requirement to notarize financial data to enable the traceability and immutability of cost records and guarantee a trusted financial audit of the supply chain. Currently, large consulting and auditing companies manage the audit process, which leads to bottlenecks and manual errors and reduces efficiency and trust. The notarization of financial data through BCDTs would make it easier to audit financial information and avoid unnecessary margins by making prices transparent. Finally, the sixth problem theme was to collect the correct cost data for the pricing of a project. Currently, it is difficult to price a project accurately due to incomplete modeling and inaccurate sources for price data. BCDTs could provide a single source for "frozen" BIM models to be priced by leveraging decentralized oracles to obtain accurate prices of materials from trusted data sources.

The components selected for developing the 5D smart-contract framework are presented as follows. The proposed 5D use case enabled decentralized open tendering (importance rank 1) with accurate pricing for BIM 5D (importance rank 6) while enabling automated payments (importance rank 2), and financial-data notarization (importance rank 5) through tokenized datasets (importance rank 4). The requirement for decentralized banks (importance rank 3) was not selected for the use case developed in this paper and should be the subject of future research work that is particularly focused on decentralized finance (DeFi) applications [27]. This use case provided an acceptable framework for the 5D smart contracts to address the problems and FRs identified in Step 1.

Maintenance context (6D): the problems identified for the BCDT 6D (maintenance) dimension are presented in order of importance as follows. Firstly, asset management should be improved by BCDTs so that facility managers can monitor building-asset information in real time and predict and automate maintenance while owners can manage their assets using their digital tokenized representation. Asset management enabled by BCDTs should ensure data security and traceability and enable decentralization and privacy as required. Secondly, BCDT should secure IoT management through decentralization to reduce single points of failure. Thirdly, the management of maintenance contractors could be automated by smart contracts to reduce manual tasks. The fourth problem theme was related to the requirement to notarize smart-building states and data records in order to enable authorized stakeholders to audit trustworthy historical states—and hence reduce risks—throughout the lifecycle of the smart-building asset. Finally, the fifth theme was related to the requirement to automate and decentralize the leasing processes to improve efficiency, traceability, and accountability.

The components selected for the development of the 6D smart-contract framework are presented as follows. The 6D use case proposed in this paper improved asset management (importance rank 1) with predictive and automated maintenance activated by smart-asset states obtained automatically from various sources such as sensors (energy, wastes, cost, temperature, and maintenance requirements) (importance rank 4). This framework also contributes to improving maintenance contractors' management (importance rank 3). The security of IoT network management (importance rank 2) is essential for facility-management organizations that are investing largely in this domain to enhance smart-building safety and improve cybersecurity.

Sustainability context (7D): Firstly, it is required to have visibility and trace energy usage patterns through the distributors and end users. For this purpose, tokens could be leveraged to buy and sell energy in a decentralized, trusted, and secure way for B2B and B2C business models. Similarly, the second problematic theme we identified was related

to improving the management and consumption of energy by leveraging P2P trading of energy surplus within smart grids in secure, scalable, immutable, and traceable ways. The third problem theme was related to the suitability and compliance of sustainability metrics to enable capital providers and regulators to track, flag, monitor, and audit the system, which provides a single open source of the truth of immutable historical sustainability metrics. In parallel, the fourth problem theme was related to the requirement to notarize environmental data so that the BCDT system could record environmental data such as air quality, temperature, weather data, and smart-infrastructure energy consumption in an immutable and traceable way. The fifth problem theme was related to the requirement to reduce construction wastes and facilitate reuse and recycling for the circular economy. For this purpose, BCDT systems require the identification of the provenance of materials and components and the carbon footprint at the source (i.e., from the manufacturer or distributor) and tracing these throughout the project's lifecycle along the entire value chain and until reuse and recycling. The sixth problem theme was related to the requirement to reduce the carbon footprint of infrastructures. BCDT systems could contribute to monitoring carbon emissions and facilitate secure P2P trading of carbon credits by leveraging open and scalable blockchain networks.

The components selected for developing the 7D smart-contract framework are presented as follows. The 7D use case proposed in this paper allowed us to capture environmental data from the information value chain to monitor green asset states such as energy usage and its distribution (importance rank 1). The system enabled data notarization of green assets (importance rank 4) through tokenization, which can enable P2P trading of energy (importance rank 2) and trace materials and reduce wastes through reuse and recycling for the circular economy (importance rank 5). Hence, the model contributed to reducing the carbon footprint (importance rank 6). The framework also enables regulators to audit and validate compliance (importance rank 3). This use case provided an acceptable framework for the 7D smart contracts to address the problems and FRs identified in Step 1.

Health and safety context (8D): Due to a lack of data and interviews for the 8D safety dimension, this study did not obtain problem themes, FRs, or NFRs for the BCDT 8D dimension. However, our previous study on BCDTs [2] indicated that BCT can enhance the 8D safety dimension by strengthening risk identification, reducing risk through transparency, and enforcing regulatory compliance for safety. Hence, the 8D use case proposed in this paper was a trusted, decentralized, immutable, traceable, and transparent risk register that enables regulators to verify compliance. The proposed use case allowed us to record projects' risks on the blockchain and enhanced safety through transparency, immutability, and traceability of the records.

Contractual context (cD): The problems identified for the BCDT cD (contractual) dimension are presented in order of importance as follows. Firstly, it is essential that BCDT systems guarantee data integrity by notarizing authenticated data in the blockchain to provide a single immutable source of truth of historical transactional data. Secondly, the system should enable smart legal contracts to automate specific agreements and contractual processes in a decentralized, transparent, and immutable way while providing secure and traceable records of contractual agreements. Two problems that were equally important came in third: the necessity to enforce policies and regulatory compliance and the necessity to prove accountability, data ownership, IP, and copyrights. Regulators should be able to audit the compliance of contracts. Accountability is key for BCDTs in order to prove responsibilities and liabilities. The identity of stakeholders should typically remain private; private blockchain, encryption mechanisms, or zero-knowledge proofs for self-sovereign identity (SSI) [28] may be appropriate for this purpose. The fourth problem theme was related to the requirement to verify and trace the identities of project stakeholders and devices in a secure and immutable way. Digital identities should be leveraged for access-control mechanisms; the BCDT applications would conceal certain data depending on the users' access rights. Finally, the fifth problem theme indicated that the governance of

BCDTs should be decentralized so that stakeholders can vote on governance decisions in an open and secure way.

The main components selected for the development of the cD smart-contract framework are presented as follows. The cD use case proposed in this paper was a smart legal contract to automate a maintenance contract (importance rank 2). The system guaranteed the integrity of the contractual data records, which were notarized on the blockchain (importance rank 1) and proved the accountabilities of the stakeholders involved (importance rank 3) and ensured regulatory compliance (also importance rank 3) via the 3D contract linked to the cD maintenance legal contract. The identity verifications (Problem 4) and decentralized governance (Problem 5) were not selected for the cD smart-contract use case proposed in this study and should be the subject of future research work. The proposed use case only focused on the three most important problems. This use case provided an acceptable framework for the cD smart contracts to address the problems and FRs identified in Step 1. To conclude this section, the above paragraphs provided a list of the key CI problems for each BCDT dimension. Moreover, they provided the related FRs and NFRs for each problem theme identified. Hence, this section addressed the first objective of the study.

Blockchain technical context: A technical action research interview was organized with a blockchain expert to discuss the technological framework proposed in this paper. The results from the online survey, combined with the outcomes from the interviews, allowed designing software architecture and smart contracts framework for BCDT applications. The elements discussed during the technical action research interview are presented as follows. Firstly, the initial version of the BCDT architecture was discussed with the blockchain expert. Secondly, the implications of the technical gaps areas identified in our previous study [1] and presented in Section 2 were discussed. And finally, the proposed smart contract framework was presented and discussed with the blockchain expert.

The outcomes of the technical action research interviews are presented as follows. It was suggested that the BCDT software architecture should contain two application programming interface (API) layers: one API for the BCDT software applications themselves and a second utility API layer such as web3.js [29] to connect these applications to the blockchain (Web 3) layer. In regard to the smart contract's components of the blockchain layer, it was also suggested to add a layer of generic smart contracts overarching all the BCDT dimensions to connect and articulate the BCDT dimensions of smart contracts throughout the lifecycle of projects. This overarching layer of generic smart contracts would contain smart-contract patterns such as a contract registry (to facilitate the updating of smart-contract addresses), a data contract (to store data in separate contracts), a factory contract (to contract a factory to generate contract instances), embedded permission (to enforce access-control conditions for some functions), and incentive execution (to reward smart-contract users) [30]. It was also discussed that the key management layer allowing smart-contract access should be separate from the storage layer and should be either centralized or preferably decentralized to enhance security. It was suggested that the IoT layer for collecting sensor data should be linked to the blockchain layer and key management layer to address the critical requirement in terms of IoT sensor identities and IoT data authentication. This would ensure that IoT devices can be authenticated with blockchain-based protocols [31] and secure elements [32]. In terms of scalability requirements, the volume of data anchored on public blockchains should be minimized due to BCT storage limitations [1]. Hence, off-chain storage systems should be used when possible. For example, important IoT data—such as deviations of sensor data from the mean, error logs, sensors alerts, and device-maintenance logs—would be required to be recorded on-chain, whereas less critical sensor data without meaningful historical value could just be stored off-chain. To manage what sensor data should be stored on-chain or off-chain (database or distributed storage), it is essential to include device agents between the IoT layer and the storage layer. In the IoT layer, the device agent services should run on dedicated machines such as master nodes (or non-resource-constraint IoT devices) equipped with

secure elements that provide hardware-generated keys. Finally, in terms of blockchain network governance, each network should have its own governance; for example, a private blockchain is governed by one organization and a consortium blockchain is governed by a subset of stakeholders and entities; the project itself cannot control a public blockchain. It was also suggested to leverage private blockchains for the data that are required to be confidential. The findings from the interviews improved the BCDT architecture and smart-contract frameworks proposed in Section 5.

5. Findings

5.1. BCDT Architecture

The survey results permitted the framing of the design of the BCDT architecture (presented in this section) to address the second objective of the paper. This section presents the relevant findings from the answers to the survey statements.

The proposed BCDT architecture naturally leveraged BCT to enhance data sharing within a decentralized data value chain empowered by the BCDT maturity Level 4 leveraging P2P collaboration [2]. The BCDT software architecture leveraged dApps to ensure data integrity and trust for digital twin applications in CI 4.0. Access-control mechanisms were integrated into the architecture and smart-contract framework to restrict the access to BCDT dApps to authorized project stakeholders only. Additionally, the BCDT architecture required privacy systems such as privacy protocols, encryption mechanisms, and private blockchains to satisfy the privacy requirements for confidential data—such as design data, some financial transactions, IP, and contracts—of smart infrastructure projects. The BCDT architecture and smart-contract framework allowed for a standardized data structure that is open source. Hence, the architecture contained a data-management component to organize the structured Big Data leveraged by BCDTs. The data-management layer cleaned, curated, and aggregated the data in a structured way. In terms of data storage, the BCDT architecture leveraged distributed storage systems to enhance cyber resilience. In the context of a Web 3.0-based CI 4.0, the BCDT architecture and smart-contract framework of the 5D (cost) dimension leveraged tokenization and trading on decentralized marketplaces to monetize IP and data ownership. Digital identity solutions [33] and particularly SSI solutions [34]—using BCT and smart contracts—were integrated into the BCDT architecture to enhance the traceability of professionals' identities throughout the project lifecycle in case of litigation. The governance of projects should be decentralized; the BCDT architecture considered this requirement by representing project stakeholders not only as BCDT dApps users, but also as blockchain validator nodes or staking nodes that were able to vote on decisions regarding the private, consortium, or public protocol governance. The BCDT architecture comprised private nodes within organizations running a private blockchain internally. The framework allows organizations to run nodes for the consortium blockchain(s) with which they are involved through partnerships such as joint ventures. The BCDT architecture ensured that organizations' blockchain applications can transact with public blockchains to record auditable data (e.g., carbon credits, certification, and safety records) on public blockchains. The BCDT architecture enables public and private organizations to run nodes for public blockchains on which they transact key data to enhance trust and data integrity. The blockchain trilemma signifies that a blockchain network needs to compromise either decentralization, scalability, or security [35]. In regard to the blockchain trilemma properties, the requirements of the CI are firstly the security to ensure resilience against cyber threats; secondly, decentralization to reduce single points of failure and ensure immutability and data integrity; and lastly, scalability/throughput to process large volumes of transactional data. Hence, since security is the most important requirement of the blockchain trilemma, the BCDT architecture also leverages public blockchains, which are more cyber resilient than private or consortium blockchains. Moreover, decentralization—which is the second most important requirement of the blockchain trilemma for the CI—also contributes to network security by reducing single points of

failure. Privacy mechanisms and key management can also contribute to security and should be considered adequately in the architecture.

For the BCDT cD (contractual) dimension, legal smart contracts are required to automate regulatory processes. The BCDT architecture and smart contract required that key processes—council's DA approvals, payment processes, engineering checking/Q&A, certification processes, tendering processes, contractual processes, and asset-management processes (e.g., maintenance)—are automated with smart contracts. BCDT smart contracts required access-control mechanisms to restrict access to contract functionalities to authorized stakeholders only. Data ownership could be monetized on decentralized marketplaces to incentivize data owners to produce information. Hence, the BCDT architecture and smart contract required mechanisms to tokenize data ownership into smart contracts such as NFTs that are linked to the datasets to be monetized. Finally, the BCDT smart contracts should ensure that ownership of smart contracts containing data can be assigned to the relevant data owners as required.

Finally, the findings from the online survey and the interviews permitted us to propose the BCDT software architecture as shown in Figure 1. The proposed BCDT architecture contained five layers. Firstly, the application front-end layer comprised the software applications and dApps UI for all BCDT dimensions. The project stakeholders from the infrastructure BCDT ecosystem could interact with these applications. Project participants could collaborate and share data in a distributed way in accordance with the BCDT maturity Level 4 [2]. These applications were connected to software APIs and utility APIs to exchange data with the back-end layers including the blockchain layer and the computer layer. The second key layer was the blockchain layer, which included private, consortium, and public blockchains to address the various requirements in terms of privacy, data sharing, data integrity, and data notarization. Interoperability protocols were also considered to ensure that different blockchain networks from the blockchain layer could transact with each other. Privacy was achieved with private or consortium blockchains and by leveraging access control and encryption mechanisms. For public blockchains, privacy can be achieved to a certain extent with blockchain-based privacy protocols. The project stakeholders and organizations participating in the BCDT ecosystem could run private, consortium, and public blockchain nodes as required. The blockchain layer also included smart contracts to automate the key processes, business logic, and requirements identified in Section 4. The smart contracts colored in purple (as shown in Figure 1) typically contained private information and could run on private and consortium blockchains. The smart contracts colored in orange (as shown in Figure 1) typically contained non-confidential information and could run on open public blockchains. The smart contracts with colored borders matching the BCDT dimension color code (as shown in Figure 1) designated the smart contracts that automated the key use cases identified in Section 4 and to be developed in this study as presented in Section 5.2. The third layer of the BCDT architecture was the computer back-end layer containing the off-chain data storage, computing, and key-management components. The data storage leveraged decentralized cloud storage and distributed file-storage systems to enhance decentralization and reduce single points of failure. Similarly, the off-chain computing component leveraged distributed computing systems. Centralized cloud services were also included if required for specific tasks requiring a lower level of security. The fourth layer was the Big Data management layer, which included middleware and data-management solutions to clean, curate, and aggregate the data in a structured way and in accordance with the Gemini Principles [36]. The Big Data management layer included decentralized oracles for trusted data acquisition from external sources into the blockchain and reversed oracles to extract data from the blockchain to off-chain components when it was required to read and analyze specific blockchain data. Finally, the sensing layer contained sensors, RFIDs, and IoT devices that captured data at each phase of the lifecycle and for each BCDT dimension. The sensing layer also comprised servers at the edge to analyze the IoT data on the premises to enhance analytic efficiency and reduce

latency. The device agents managed what data was recorded on-chain and what data was stored off-chain.

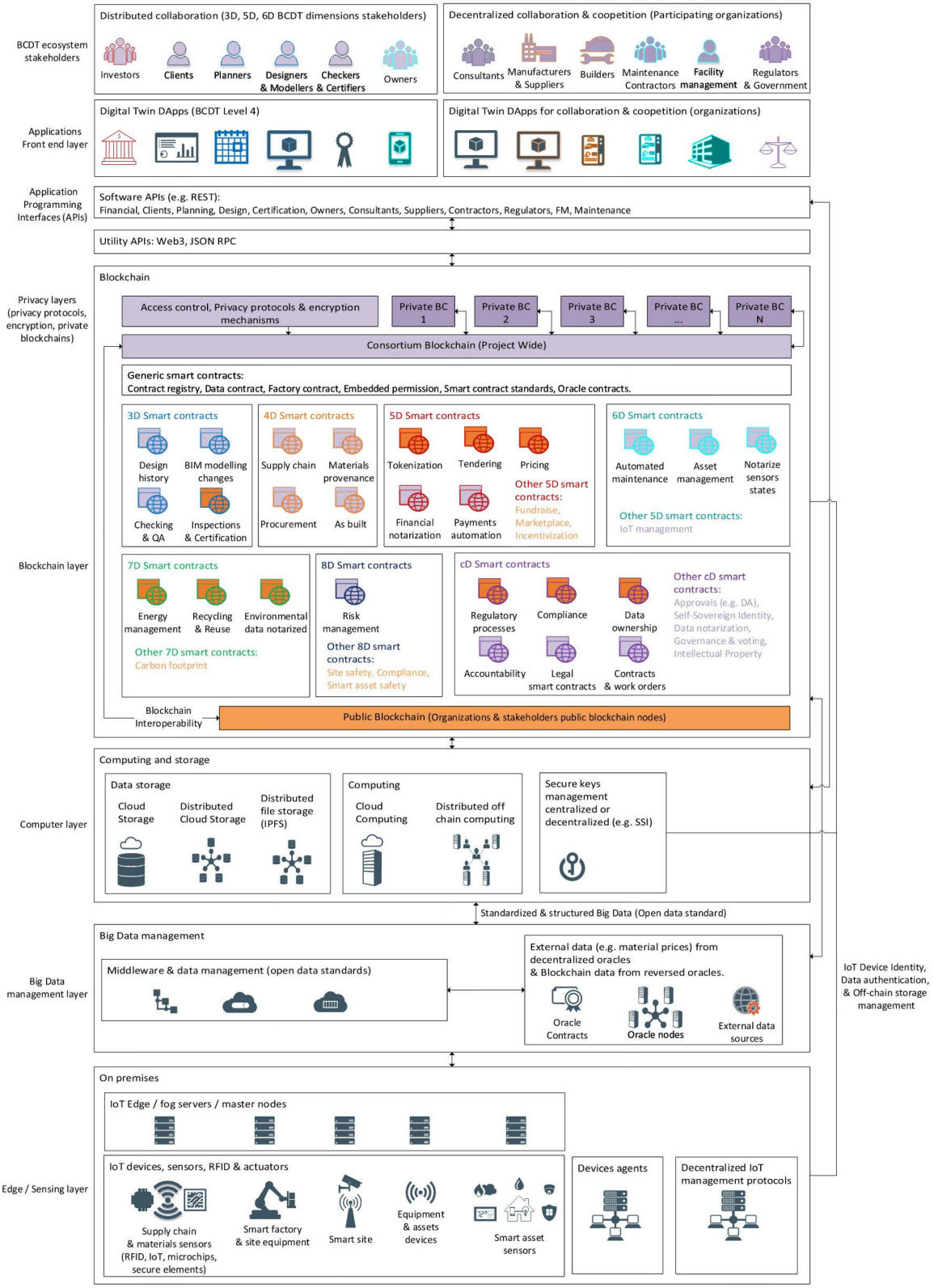

Figure 1. BCDT software architecture.

Finally, the proposed architecture aimed to address main gaps areas in the DDTC [1] such as the technical requirements of BCT (network governance, scalability limitations, decentralization, interoperability, protocol efficiency, and computational requirements), which will be discussed further in Section 6. In terms of integration of the IoT with BCT, the sensing layer of the architecture linked the IoT to the blockchain layer and included master nodes to address limitations of resource constraint of IoT devices, secure elements for data authentication and device identity, and blockchain-enabled IoT networks to improve security. In terms of integration of BIM with BCT, the BCDT architecture included BIM and ensured that the Big Data from the value chain of complex project lifecycles was stored in structured and format-agnostic ways and in accordance with the Gemini Principles [36]. The integration of the 7D (sustainability) BCDT dimension into the architecture enabled the integration of the circular economy into the proposed BCDT framework. Hence, the proposed architecture addressed the key gaps areas of the DDTC. Finally, the proposed BCDT software architecture addressed the second objective of this paper.

5.2. BCDT Smart Contracts (Using NFTs)

This section focuses on the development of a framework for smart contracts—for each BCDT dimension—to address the third objective of this study. The analysis of the survey results presented in Section 5.1 combined with the use cases derived from the action research interviews permitted us to define the design requirements for the BCDT smart contracts developed in this section. This study focused particularly on the key use cases identified in Section 4 throughout the action-planning phase (Step 2) of the action research process. To address these use cases, specific smart-contract themes were selected for each BCDT dimension, as shown in the BCDT architecture in Figure 1.

The following paragraphs present the smart-contract themes developed for this study for all BCDT dimensions. Thus, this section represents the action taking (Step 3) of the action research process and consists of implementing the use cases from Section 4 into smart-contract logic.

As identified in Section 4, the BCDT architecture and smart-contract frameworks enabled mechanisms to tokenize data ownership and IP to monetize them as digital assets that could then be traded on decentralized marketplaces. Therefore, data ownership—and more generally information as an asset—could be monetized on decentralized data marketplaces to incentivize data owners to produce information. To achieve this purpose, this study leveraged the non-fungible token (NFT) smart-contract standard to tokenize value into digital assets (tokens) that could be transferred on the blockchain. Hence, datasets that were specific to the BCDT dimensions could be tokenized into NFTs that could then be transferred—enabling the transfer of ownership—and traded on digital marketplaces. More generally, any type of value—such as datasets, IP, data ownership, or physical assets—can be linked to NFT metadata and consequently become tokenized into a digital asset that is unique and transferable on the blockchain in a secure, decentralized, and trusted manner.

To develop the BCDT smart-contract framework, the Solidity [23] programming language was used. Solidity is the main programming language of the Ethereum blockchain [22] and other blockchains that are compatible with the EVM [24] such as Avalanche [37], Fantom [38], Polygon [39], Binance Smart Chain [40], Arbitrum [41], Gnosis (xDAI) [42], Moonriver [43]. Solidity has the largest community of developers and hence was naturally chosen to develop the smart contracts for this study. The ERC721 [19] NFT smart-contract standard from OpenZeppelin [44] was leveraged for the BCDT smart contracts. Moreover, access-control requirements were integrated into the smart contracts via inheritance from the Ownable [45] smart contract from the OpenZeppelin framework. The BCDT smart contracts were developed using key tools such as Remix IDE [46] for initial implementation and the Hardhat [25] deployment environment for testing and deployment. The proposed smart-contract framework leveraged the Chainlink [47] decentralized oracle to obtain data from external sources through HTTP GET requests using the Chainlink API Consumer smart contract. For this study, the Chainlink oracle smart contracts were deployed on the

Ethereum Rinkeby testnet. Since some BCDT smart contracts' functions—for the 5D, 6D, and 7D dimensions—called the Chainlink oracle, it was required to deploy some smart contracts not only within the Hardhat environment, but also on the Rinkeby testnet to enable the interactions with the oracle smart contracts as needed. It should be noted that the BCDT project metadata datasets linked to the NFT smart contracts (via the token URI) should typically be recorded on decentralized storage systems such as IPFS [48]. However, the NFT metadata specifications were out of the scope of this study, which focused on the smart contracts of the blockchain layer.

The basic use cases identified for the BCDT dimension during the action-planning phase (Step 2) were discussed in Section 4. The action-taking phase (Step 3) presented in this section turned these use cases into smart-contract logic. The smart-contract logic developed for each BCDT dimension is presented in the following sections. The smart contracts were also programmed with Solidity [23] and the gas consumption of all functions throughout the lifecycle of a project were quantified as discussed further. An extract from the BCDT cD dimension smart contract developed with Solidity is presented in Figure 2.

```
pragma solidity ^0.8.7;

import "@openzeppelin/contracts/access/Ownable.sol";
import "./ZoneToken_6D.sol";

contract MaintenanceLegal is Ownable {

    address public maintenanceContractor;
    address public zoneTokenAddress;
    uint public smartAssetTokenId;

    // ZoneToken ERC721 (NFT) smart contract representing the smart asset being maintained
    ZoneToken zoneTokenNFT;
    // Boolean indicating if the smart asset requires maintenance (get value from zoneTokenNFT)
    bool public needsMaintenance;
    bool public maintenanceServiceCompleted = false;
    bool public maintenanceServicePaid = false;

    // maintenance fee is private and can only be seen by authorized stakeholders
    uint private maintenanceServiceFeeinWei;

    constructor(address _zoneTokenAddress, uint _smartAssetTokenId) {
        zoneTokenAddress = _zoneTokenAddress;
        zoneTokenNFT = ZoneToken(zoneTokenAddress);
        require(zoneTokenNFT.needMaintainance(_smartAssetTokenId) == true, "This smart asset does not require maintenance yet");
        smartAssetTokenId = _smartAssetTokenId;
        transferOwnership(zoneTokenNFT.ownerOf(_smartAssetTokenId));
        maintenanceContractor = zoneTokenNFT.maintenanceContractors(_smartAssetTokenId);
        needsMaintenance = zoneTokenNFT.needMaintainance(_smartAssetTokenId);
    }

    function setMaintenanceServiceFee(uint _maintenanceServiceFeeinWei)
        public
        onlyMaintenanceContractor
        returns (uint)
    {
        maintenanceServiceFeeinWei = _maintenanceServiceFeeinWei;
        return maintenanceServiceFeeinWei;
    }

    function getMaintenanceServiceFee()
        public
        view
        onlyOwner
        returns (uint)
    {
        return maintenanceServiceFeeinWei;
```

Figure 2. Solidity smart contract extract from the BCDT cD dimension contract.

The sequence diagram of the 6D maintenance dimension shown in Figure 3 illustrates the logic of the BCDT 6D smart-contract framework.

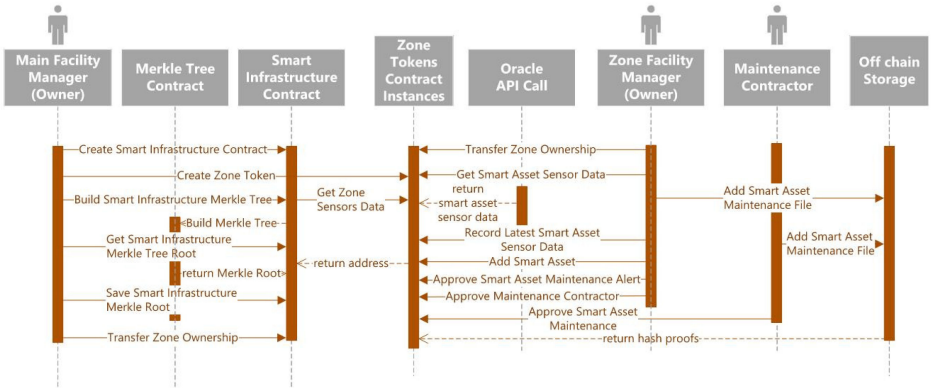

Figure 3. Sequence diagram of the BCDT 6D smart contract.

It should be noted that the MerkleTree smart contract [49] allowed us to store sensor data hashes in the leaf nodes of a Merkle tree data structure. Figure 4 shows the typical structure of the Merkle tree used for this study.

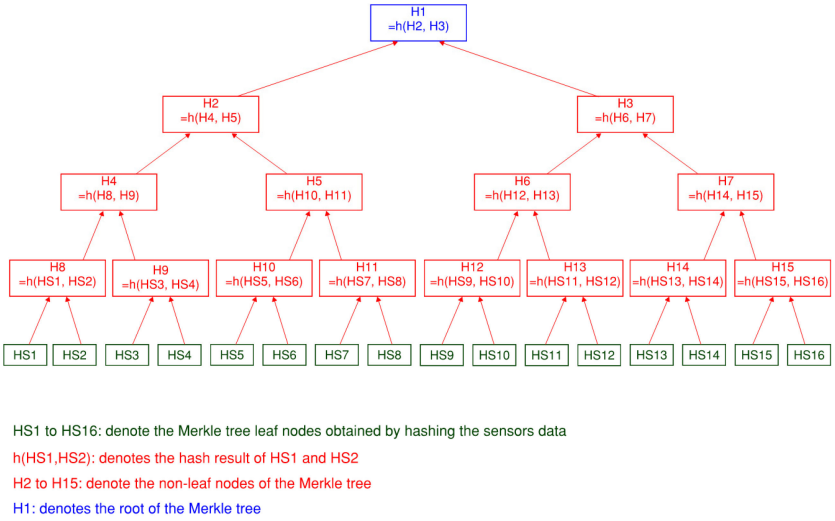

Figure 4. Merkle tree data structure for the sensor data.

The sequence diagram of the cD contractual dimension shown in Figure 5 illustrates the logic of the BCDT cD smart-contract framework

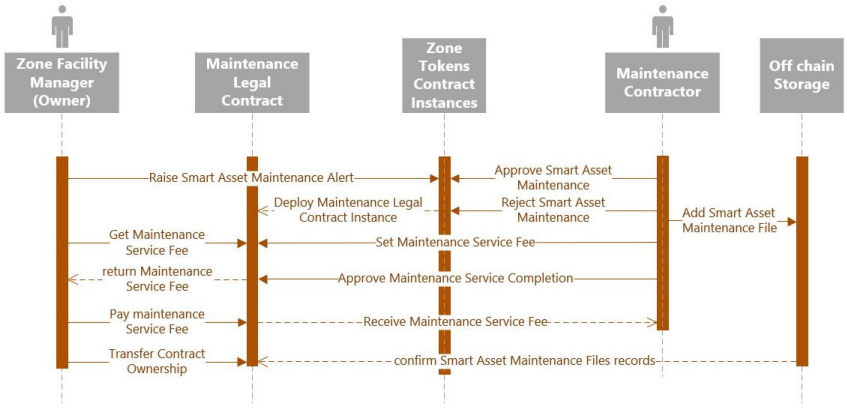

Figure 5. Sequence diagram of the BCDT cD smart contract.

Finally, the proposed smart-contract framework for the BCDT dimensions addressed most of the key requirements of the use cases identified in Section 4 for each dimension. The smart-contract framework addressed the critical problems and FRs identified in Section 4 as discussed further in Section 6. Consequently, the smart-contract framework proposed through the action taking (Step 3) of the action research process permitted us to validate the third objective of this study. The following section focuses on evaluating the cost performance of this smart-contract framework throughout the lifecycle of a simple medium-sized use-case project.

5.3. Cost Analysis

This section describes the evaluation phase (Step 4) of the action research process. The smart contracts proposed in Section 5.2 were tested using the Hardhat [25] framework. The tests were written with JavaScript and leveraged the Ethers.js [50] library to interact with the deployed smart contracts. Firstly, the testing phase permitted us to evaluate the correctness of the smart-contract functions and correct them as needed to ensure that all the tests were passed and hence strengthen the validation of the third objective. Secondly, the testing phase leveraged the Hardhat gas reporter plugin [51] to extract the gas consumption of the functions of each smart contract. Once the gas consumptions were identified for all key functions, the study evaluated how many times each function would need to be called throughout the lifecycle of a sample use-case project. This allowed evaluation of the total quantity of gas required throughout the lifecycle of a project by leveraging the proposed BCDT smart-contract framework for all dimensions. Finally, the total gas consumption was multiplied by the gas price of several EVM compatible blockchains: Ethereum [22], Avalanche [37], Fantom [38], Polygon [39] (layer-2 solution), Binance Smart Chain [40], Arbitrum [41] (layer-2 solution), Gnosis (xDAI) [42], and Moonriver [43]. The gas prices for each of these blockchains were obtained using the online tool CoinTool [52] and are summarized in Table 3. This approach permitted us to conduct a comparative cost analysis depending on the type of blockchain network used and to develop a criterion for evaluating the suitability of blockchain protocols for the proposed BCDT smart-contract framework. It should be noted that the cost of the decentralized oracle jobs for the HTTP GET requests tasks—leveraged for the 5D, 6D, and 7D smart contracts—were obtained from the Chainlink Market website [53]. Table 4 shows the results of the main cost analysis (for all BCDT dimensions) with and without the use of a decentralized oracle.

Table 3. Gas costs for the key EVM-compatible blockchain networks considered.

Blockchain Network	Ethereum	Avalanche	Fantom	Polygon [2]	BNB	Arbitrum [2]	xDAI	Moonriver
Gas price [1] (Gwei/gas)	70	26	196	40	5.1	1	2.1	1.5
Native cryptocurrency unit	ETH	AVAX	FTM	MATIC	BNB	ETH	XDAI	MOVR
Native cryptocurrency price [1] (USD)	3756.00	116.00	2.47	2.53	524.00	3756.00	1.00	202.00
Gas cost [1] (USD/gas)	262.9×10^{-6}	3.0×10^{-6}	484.1×10^{-9}	101.2×10^{-9}	2.7×10^{-6}	3.8×10^{-6}	2.1×10^{-9}	303.0×10^{-9}

[1] Data measured at the time of writing on 3 January 2022. [2] Denote layer-2 scaling solutions.

A sample use-case project was chosen in order to evaluate the number of function calls throughout the project lifecycle for each smart contract of the proposed BCDT framework. The use-case project selected was the advanced sample project obtained from the Autodesk Revit sample project files [26]. The quantities were derived from the Revit model as follows. Three main standard disciplines were selected: architecture (represented by the rac_ advanced _sample_project.rvt file), structure (represented by the rst_advanced_sample_project.rvt file), and building services (represented by the rme_advanced_sample_project.rvt file). The advanced sample model was chosen because

it is adequately representative of a medium-sized smart-building project. The number of packages and sub-packages identified for each discipline were extracted using Revit 2020.

Table 4. The total price of execution of the BCDT smart contracts for the use-case project lifecycle with a decentralized oracle (Chainlink) and without a decentralized oracle.

Blockchain Network	Ethereum	Avalanche	Fantom	Polygon [2]	BNB	Arbitrum [2]	xDAI	Moonriver
Without decentralized oracle [1]	USD 4,778,240	USD 54,812	USD 8798	USD 1839	USD 48,568	USD 68,261	USD 38	USD 5507
With decentralized oracle [1]	USD 4,782,815	USD 59,387	USD 13,373	USD 6414	USD 53,143	USD 72,836	USD 4613	USD 10,082

[1] Values derived from data measured at the time of writing on 3 January 2022. [2] Denote layer-2 scaling solutions.

For the smart contracts of the 3D spatial dimension, each design package was represented by a DesignToken ERC721 NFT contract; for each design package, sub-packages were minted as NFTs. For example, one ERC721 DesignToken contract were deployed for the structural walls package, and six NFTs were minted (from that DesignToken contract) so that each NFT represented a specific structural wall design tokenized entity. A total of 60 packages were created (i.e., DesignToken deployed) and a total of 8224 sub-package NFTs were minted. A similar approach was followed for the 4D time dimension contracts. For the 5D time dimension, it was assumed that five tenderers would apply to this project client tender and that the payments for the winning tenderer would be spread monthly over a construction period of 12 months. For the smart contracts of the 6D maintenance dimension, the number of function calls was measured throughout an O&M duration of 25 years, which is a standard service-design life for a building [54]. Due to a lack of data on the typical maintenance requirements for a medium-sized building, some additional questions were presented to a facility manager from the property-management team that was interviewed for the action research 6D interview. The key findings suggested an estimated average of about 213.5 inspections annually for all the main building-element categories such as vertical transport, hydraulics, electrical, access-control security, and fire and mechanical systems. Hence, an estimated 5338 maintenance inspections would occur for an O&M duration of 25 years. For simplicity, it was assumed that the sensor data were audited for each maintenance inspection and a smart maintenance alert was raised. Additionally, since there was no particular utility in recording the daily "well-functioning" states of smart assets on the blockchain, it was assumed that only abnormal states were recorded on-chain. Hence, the sensor data were only recorded on the MerkleTree data-store smart contract when there was a maintenance operation. Similarly to the 3D, 4D, and 5D dimensions, the application of the 7D sustainability dimension smart contract for the proposed use case led to a total of 60 green packages so that each discipline package was represented as a green package that could be shared, reused, or recycled at the end of the project lifecycle. The associated number of green assets (sub-packages) that was estimated was 8224. Finally, to ensure traceability, trading, reuse, and recycling of green assets throughout the project lifecycle, it was assumed that the ownership of each green asset was transferred about four times between key stakeholders in the project lifecycle (manufacturer, supplier, builder, facility manager, recycler). For the 8D BCDT smart contracts, one RiskToken contract was deployed for each package, hence a total of 60 RiskToken contracts were deployed for the use-case model. An average of 10 risks for each package was assumed, which led to a total of 600 risk NFTs to be minted. Finally, for the cD BCDT smart contract proposed in this study, a MaintenanceLegalContract instance was deployed for each smart asset requiring maintenance. Since there was a total of 16 smart assets (four smart assets in each of the four zones), the MaintenanceLegalContract smart contract was deployed 16 times.

Table 3 shows the gas prices identified for the EVM-compatible networks considered in the cost analysis. It should be noted that these networks are typically layer-1 blockchains

except for Polygon [39] and Arbitrum [41], which are layer-2 scaling solutions running on top of the underlying Ethereum [22] blockchain (layer-1). The gas price fluctuated in real time; the values shown in Table 3 were measured using CoinTool [52] at the time of writing. The costs of the native cryptocurrencies for these blockchain networks also were fluctuating; the values shown in the table were measured at the time of writing. Finally, the gas costs in USD for each of these blockchains were obtained by multiplying the gas price in Gwei by the native cryptocurrency price. Typically, 1 Gwei = 10–9 ETH, where ETH represents the native cryptocurrency of the Ethereum blockchain. This conversion also applied to the other EVM-compatible blockchains included in this cost comparison and has a different native cryptocurrency as listed in the second row of Table 3. Hence, Table 3 shows the gas prices (in Gwei/gas), the native cryptocurrencies units, the native cryptocurrencies prices (in USD), and the gas costs (in USD/gas) for the EVM-compatible public blockchain networks considered in the cost analysis.

In order to obtain the total cost, the gas cost (in USD/Gas as shown in the last row of Table 3) was multiplied by the total gas consumed by all functions from the BCDT smart contracts for all dimensions (3D, 4D, 5D, 6D, 7D, 8D, and cD). Two options were considered for the calculation of the total cost. One option excludes the use of the Chainlink [53] decentralized oracle, and the second option includes the cost of using the Chainlink decentralized oracle for the functions add5DpackagePricingFiles (5D), getSmartAssetSensorData (6D) and getGreenAssetData (7D) that query data from external online sources via HTTP GET requests. It should be noted that the cost for each HTTP GET request task done by the Chainlink API consumer contract is typically 0.01 LINK token as shown on the Chainlink Market website [53]. The LINK token price at the time of writing is 21 USD. Hence for the three functions mentioned above, the oracle price was added for each function call, and multiplied by the number of function calls in order to get the estimated price throughout the full lifecycle of the project. Finally, the results of the cost analysis for all the BCDT dimensions smart contracts—with and without a decentralized oracle—for the medium size smart building use-case project are shown in Table 4.

Table 4 shows that the use of a decentralized oracle led to a price increase of USD 4575.06 for all the networks compared. This represented a price increase of less than 10% for most networks such as Ethereum (+0.1%), Avalanche (+8%), BNB (+9%), and Arbitrum (+7%). However, it represented a more significant price increase for Fantom (+52%), Polygon (+250%), xDAI (+12,000%), and Moonriver (+80%) because these blockchains had cheaper gas costs. It should be noted that Table 4 shows the cost of deployment and execution on public blockchains for all the dimensions of the BCDT framework. However, since the online survey results revealed that most data might be required to be confidential with the exception of certification, safety, and environmental data, the total execution cost could be reduced by only anchoring the 7D, 8D, and some elements of the 3D dimensions onto public blockchain networks.

6. Discussion

This section firstly discusses how the proposed BCDT architecture and smart-contract framework narrowed some key gaps in our previous study on DDTC [1]; secondly, it addresses the maturity Level 4 and NFRs identified in our previous study on BCDTs [2]; and thirdly, it addresses the key industry problems and related FRs and NFRs identified in this paper for each BCDT dimension. Finally, this section discusses the cost-analysis results presented in Section 5.3 and develops criteria for evaluating the adequacy of blockchain protocols for the proposed BCDT smart-contract framework.

DDTC gaps areas: The proposed architecture aimed to address the gaps and areas identified in our previous study on DDTC [1]. Table 5 presents these gaps and areas and compares them with components of the proposed BCDT architecture and smart-contract framework to discuss how each gap was addressed. The themes of the gap area related to the technical requirements of BCT are further discussed in Section 6.2 with the NFRs of BCDTs in CI 4.0.

Table 5. Evaluation of the proposed architecture against the technical gaps.

Gap Areas	Themes	Comparison with Components of the Proposed Architecture
Technical requirements of BCT	Network governance	The BCDT architecture included private, consortium, and public blockchains. The governance methods of these blockchain types were different. Decentralized governance for BCDTs is discussed in Section 6.2.
	Type of blockchain (private, consortium, public)	The BCDT architecture included private, consortium, and public blockchains to allow an adequate degree of privacy in accordance with project requirements. Privacy is discussed in Section 6.2.
	Scalability limitations	The EVM-compatible public blockchains presented in Section 5.3 had different levels of scalability; adequate throughput should be leveraged for BCDT dApps as discussed in Section 6.2.
	Decentralization	The EVM-compatible public blockchains presented in Section 5.3 had different degrees of decentralization. Optimal decentralization should be leveraged for BCDT dApps as discussed in Section 6.2.
	Cross-chain interoperability	The EVM-compatible public blockchains presented in Section 5.3 either required cross-chain bridges for interoperability or leveraged native cross-chain interoperability as discussed in Section 6.2.
	Energy efficiency	The EVM-compatible public blockchains presented in Section 5.3 typically leveraged proof of stake (PoS)-based consensus mechanisms, which are considerably more energy efficient than proof of work (PoW) (mining) consensus mechanisms.
	Computational requirements	The BCDT architecture integrated decentralized computing systems (cloud and distributed off-chain computing) in the computer layer.
IoT and BCT	Resource-constraint devices	The BCDT architecture integrated master nodes in the edge layer to run blockchain nodes for resource-constraint IoT devices. IoT sensor data were computed preliminary at the edge (i.e., on the premises) to improve latency and efficiency.
	Data authentication and device identities	The BCDT architecture integrated microchips and secure elements in the sensing layer to authenticate device identities. Devices agents in the sensing layer managed device identities. Decentralized oracles in the Big Data management layer authenticated IoT data.
	Management of IoT networks	The BCDT architecture integrated decentralized IoT management protocols into the edge layer to reduce single points of failure.
Big Data (e.g., BIM, GIS, and dynamic) and BCT	Big Data storage and BCT storage limitations	The BCDT architecture allowed for Big Data (BIM, GIS, static, and dynamic data) to be stored off-chain in the computer layer by leveraging hybrid and decentralized data storage systems (cloud, distributed cloud, and distributed file-storage systems).
	Data structure requirements to integrate BIM into BCT	The Big Data management layer curated Big Data (e.g., from BIM, GIS, static, and dynamic) into standardized, structured, and format-agnostic data prior to storage and computation (on-chain or off-chain). Allowed for a standardized and structured Big Data stream between the Big Data management layer and the computer layer.
Digital twin data with BCT	Integration of DT data into BCT	The Big Data management layer filtered Big Data volumes in accordance with the Gemini Principles [36]. Structured data and metadata were recorded in the blockchain in a format-agnostic way. Integration of privacy protocols, encryption mechanisms, and private blockchains to satisfy the privacy requirements of DT.
Project lifecycle complexity	Lifecycle and circular economy	The BCDT architecture integrated dApps for all key projects stakeholders and the blockchain layer included smart contracts for all BCDT dimensions to cover the full lifecycle of projects. The requirements for the circular economy were addressed by the 7D dimension's smart contracts of the BCDT blockchain layer.

BCDT maturity Level 4: Our previous study on BCDTs [2] proposed the concept of maturity Level 4 for BCDTs to offer a paradigm shift that leveraged distributed collaboration through P2P networks, P2P data sharing with decentralized common data environments (DCDE), a decentralized data value chain (decentralization of data acquisition, data analysis, data curation, data storage, and data usage), and the automation of processes with smart contracts. The BCDT architecture in Figure 1 shows that the project stakeholders (investors, clients, planners, designers, modelers, checkers, certifiers, owners, consultants, manufacturers, builders, contractors, facility managers, and regulators) could interact with DApps that were by nature decentralized and leveraged P2P blockchain networks. Hence, the collaboration (and coopetition) of key project stakeholders (and organizations) operated in a P2P way, and data sharing was enhanced by open blockchain ledgers shared between project participants. This blockchain-based decentralized data sharing guaranteed the integrity and security of information by respectively providing a single source of truth of historical transactional data and by removing single points of failure. The incentivization mechanisms enabled by NFTs and the 5D dimension encouraged project participants to share information without losing the protection of their IP. Indeed, as explained further in the discussion in Section 6.2 on data ownership, the IP of data creators is protected by the tokenization of created datasets into NFTs. The data owner also owns the NFT and hence the value associated with this digital asset representing information as an asset. Hence, decentralized data sharing within BCDT ecosystems enhances collaboration and coopetition and reduces the current information-hoarding mechanisms that are slowing progress in CI 4.0. This paradigm shift toward open data sharing comes with a decentralization of the data value chain that leverages blockchain-based decentralized protocols for data acquisition, data analysis, data curation, data storage, and data usage. Indeed, the proposed smart-contract framework leveraged the Chainlink [47] decentralized oracle to acquire data from external sources such as IoT, RFID, and APIs for the 5D (price data), 6D (sensor data), and 7D (green asset data) BCDT dimensions. In terms of data analysis and data storage, the computer layer of the proposed BCDT architecture leveraged respectively distributed off-chain computing (e.g., iExec [55]) and distributed storage systems (e.g., IPFS [48]). The data curation was facilitated by the Big Data management layer of the BCDT architecture, which leveraged middleware solutions and open data standards to better manage data in accordance with the Gemini Principles [36]. Finally, the BCDT architecture also contributed to the decentralization of data usage through the use of dApps for the end users. Hence, the proposed BCDT architecture and smart-contract framework subscribed to the decentralization of the data value chain. Furthermore, the proposed smart-contract framework enabled the automation of processes for the basic use cases identified for each BCDT dimension as presented Section 4. The automation of key processes with smart contracts is discussed further in Section 6.1 for each BCDT dimension. To conclude, the proposed architecture and smart-contract framework for BCDTs embraced a paradigm shift represented by the BCDT maturity Level 4. Hence, the implication of the proposed framework was that it offered the technological ingredients for industry practitioners to implement BCDTs for CI 4.0 and embrace a paradigm shift toward the decentralization of the industry in accordance with the BCDT maturity Level 4.

6.1. BCDT Smart Contracts vs. Industry Problems and Functional Requirements

3D contracts: The 3D smart contracts offered a solution to record historical design data, BIM changes, site-inspection records, and certificates into NFTs. Hence, this approach provided immutable and traceable records for the key historical design data related to the 3D spatial dimension. Moreover, the 3D smart contracts enabled the automation of key processes of the design phase such as checking designs for Q&A, approving site inspections, and certifying designs.

4D contracts: The 4D smart contracts allowed historical recording of data from the construction supply chain (e.g., materials provenance data and delivery data) to better manage, monitor, and trace the procurement of construction goods. The 4D smart contracts

allowed the generation of NFT tokens for each construction sub-package. The metadata of these NFTs included key supply-chain information related to the materials and goods. Hence, the 4D smart contracts could be linked to construction materials supply-chain provenance and traceability.

5D contracts: The proposed 5D smart contracts offered a solution for a client to publish a tender and for tenderers to submit tender bids containing pricing data records such as the 5D BIM models, tendering bidding price, and all the metadata required to accurately price the tender. The proposed 5D framework also enabled the automation of tendering processes such as submitting a tender, pricing a 5D package with reliable cost data obtained with decentralized oracles, getting tenderers' bid prices, selecting a winning tender, and automating payments from the client to the winning tenderer. The 5D smart contracts aimed to price 5D BIM models accurately by automatically collecting pricing data with decentralized oracles. The 5D smart contracts enabled the tokenization of the BIM 5D pricing packages into NFTs for each component of the tendering project.

6D contracts: The 6D smart contracts offered a solution to monitor infrastructures' smart assets and record key historical states on the blockchain. The 6D framework suggested only recording abnormal and out-of-range sensor states on the blockchain to minimize the data to be stored on-chain and overcome the storage limitations of blockchain ledgers. Hence, the anchorage of out-of-range sensor data on-chain would provide proof of smart-asset malfunctions in case of litigation. Moreover, misfunctioning smart assets would trigger smart alerts for automated maintenance operations. The survey results suggested that smart-asset sensor data should be traceable throughout the lifecycle of projects. However, there was no apparent benefit to recording all the regular—"within range"—sensor data on-chain when these could be easily and efficiently monitored and recorded with centralized cloud storage and computing systems. However, it could be beneficial to have digital fingerprints of such "within range" data that could be anchored periodically on-chain to prove that smart assets were functioning correctly during periods of time. Future research works should explore this approach further and develop mathematical and computational models for efficient periodical on-chain recording of the digital fingerprints of well-functioning "within range" average sensor data. Layer-2 scaling solutions with ZK-rollups could be leveraged for this purpose.

7D contracts: The 7D smart contracts allowed the gathering of authenticated environmental data about green assets by leveraging a decentralized oracle. The framework also enabled the notarization of environmental data with the tokenization of green-asset metadata into NFTs. These 7D NFTs could relate to various types of green assets such as energy, recyclable materials, or reusable materials. Since NFTs can be traded on open markets, the framework enabled P2P trading of green assets such as energy trading within a smart grid and materials trading with infrastructures as material banks [56] for the circular economy. Since the blockchain provides immutability, transparency, and traceability of historical transactions, the proposed 7D smart contract framework allowed enhanced monitoring for the consumption patterns of tokenized green assets throughout their lifecycles; i.e., from production to distribution, usage, and reuse. The traceability and improved monitoring of green assets allowed for a reduction in energy consumption and materials wastes and hence contributed to reducing the carbon footprints of the sustainable BCDT infrastructures.

8D contracts: The 8D (safety) smart contracts developed in this study were in accordance with the basic use case proposed in Section 4. The solution allowed us to create a decentralized risk register and tokenize risks into NFTs owned by the risk owners. Risk data were embedded into the NFT metadata and hence were traceable throughout the duration of the project. The risks could be closed by the risk owner and risk manager once they had been resolved. Finally, the proposed 8D risk register would allow regulators to approve the safety compliance as required.

cD contracts and future research directions: The smart contracts for the contractual dimensions (cD) developed in this study were in accordance with the key use case proposed in Section 4. The smart-contract framework for the cD dimension enabled the automation

of maintenance work orders and therefore was required to be connected to the smart contracts of the 6D dimension. The cD smart contract called MaintenanceLegal could read conditions from the 6D ZoneToken contract to evaluate the smart assets' maintenance requirements and construct a legal smart contract when work orders were required. The MaintenanceLegal contract could then automate key processes such as the approvals from the maintenance contractors and the payments required from the asset owner to the maintenance contractor. However, this use case was limited, and future works should develop standardized legal smart contracts for the automation of regulatory processes and work orders with intelligent contracts [12] while considering regulatory compliance, accountability, data ownership and IP protection, SSI, and decentralized governance for all the BCDT dimensions.

Overarching smart contracts: The findings from the technical action research interview on BCT suggested adding a layer of generic smart contracts to the BCDT architecture overarching all the BCDT dimensions to connect and articulate all the BCDT smart contracts throughout the lifecycles of projects. The contract registry pattern was not implemented for this study because the use case evaluated in this paper was only deployed on test networks to evaluate the cost. However, this study included fundamental smart-contract patterns such as a data contract implemented via the Merkel tree contract to store the smart-asset sensor data (6D). The framework also included a series of factory contracts (DesignFactory, ConstructionFactory, ClientOfferTender, SmartInfrastructure, SustainableInfrastructure, and RiskFactory) to generate the child ERC721 NFT smart contracts. The framework also included a contract for embedded permissions—called Ownable—obtained from the OpenZeppelin framework [45]. Finally, the framework considered incentive execution smart contracts through NFTs that could be sold to incentivize data creators, data owners, and IP owners through the distribution of royalty percentages from the sales.

There were limitations to the NFT-based BCDT smart-contract framework proposed in this study. For example, if a tender (5D) was required for a large maintenance and repair operation (6D) requiring new design components (3D) and significant construction works (4D), the model would lead to multiple NFTs to be considered for the 3D, 4D, 5D, and 6D dimensions. The abundance of tokenized data in NFTs could make information management and data ownership management quite complex since several dimensions were overarching. Another limitation of the study was that it did not differentiate the CI context of each country, future research work could explore the BCDT framework with a country-based approach. Moreover, it should be noted that the outputs—importance ranking of CI problems—from the action research interviews were obtained from a limited number of subject-matter experts. Therefore, these outcomes could not be fully generalized, but provided valuable insights into the importance of current problems in CI that can be addressed by BCT. Future research works should further develop the identification of key CI problems that can be addressed by the BCDT framework.

6.2. BCDT Non-Functional Requirements

Our previous study on BCDTs [2] revealed that the key NFRs for BCDT applications in CI 4.0 were privacy, security, data ownership, data integrity, interoperability, and the decentralization and scalability of data storage [2]. The NFRs discussed in this section were organized according to the ISO/IEC 25010:2011 qualities model [57].

The security requirements are discussed as follows:

Integrity: The proposed BCDT framework strengthened the data integrity of the information value chain by leveraging BCT to enable traceability and immutability of authenticated datasets for all BCDT dimensions.

Privacy/Confidentiality: The survey results suggested that most data should be private except information related to certification, safety, and environmental data. The proposed BCDT architecture considered these privacy requirements by including private blockchains, encryption mechanisms, and privacy protocols. However, the developed smart-contract framework did not include encryption mechanisms or privacy protocols.

Hence, future research works should evaluate, implement, and test that privacy requirements can be practically addressed using privacy protocols, encryption mechanisms, and private blockchains networks.

Future works should focus on evaluating all the privacy requirements and how they would fit in the context of decentralization and Web 3.0 in CI 4.0. Future works should also identify what data exactly from organizations (governments, international organizations, and private or public companies) should be auditable on public blockchains. Furthermore, private organizations can run public blockchain nodes to increase their revenue stream (via blockchain mining or staking). Future works should identify how smart cities could leverage renewable energy and excess energy by running proof of work (PoW) mining and proof of stake (PoS) staking nodes to validate public blockchain transactions and generate value in sustainable ways. For example, green-energy excesses from smart grids could be used to mine or stake public cryptocurrencies such as Bitcoin [58] or Ethereum [22].

Accountability: BCDTs ensure the immutability and traceability of transactional data, which can guarantee accountability and non-repudiation throughout the lifecycle of the project and beyond. Indeed, NFTs allow the recording of project data from all BCDT dimensions and ensure that key information from the data value chain is tokenized onto the blockchain, which guarantees immutability and traceability. Additionally, when the NFTs are transferred from an owner to a new owner, the blockchain keeps records of ownership transfers, ensuring that accountability can be traced back from data owners to data creators throughout the lifecycle of projects. Blockchain-based digital identities contribute to improve accountability, and the proposed BCDT architecture integrated SSI solutions [34] to decentralize the management of digital identities and enable traceability of identities without compromising the privacy of individuals. However, SSI smart contracts such as the ERC-725 standard [59] were not included in the scope of this paper. Moreover, once the legal accountability of a project participant has expired—after a legally defined period of time—digital-identity systems can comply with the right to be forgotten and satisfy the General Data Protection Regulation (GDPR). The right to be forgotten leads to key challenges when it comes to BCT, which provides immutable open data records [60]. State-of-the-art privacy techniques leveraging SSI and zero-knowledge proofs [28] could be leveraged to prove accountability without revealing identities. Future research works should focus on developing adequate SSI frameworks for BCDT systems and, more generally, for CI 4.0 and society as a whole.

Authenticity: The BCDT architecture and smart-contract framework proposed in this study integrated technological components to guarantee data authenticity and avoid GIGO effects. Indeed, the edge layer of the BCDT architecture included microchips and secure elements to provide cryptographically secured unique digital identities for IoT devices and sensors. This layer of security increased resilience at the edge of the IoT network and ensured that the IoT data captured were genuinely coming from specific sensors. Future research works should implement and test BCDT experiments leveraging microchips and secure elements to enable devices identities at the edge layer of BCDTs. Moreover, oracles are recommended for data authentication that are integrated into the Big Data management layer of the BCDT architecture. Using a decentralized oracle solution such as Chainlink contributed significantly to improving data authenticity and hence data integrity and security of the link between the blockchain and the sensing layer. Moreover, the authentication of data by decentralized oracles contributed to limiting the GIGO effects and strengthening the data value chain for BCDTs.

Interoperability: The interoperability between blockchain networks is a key NFR for BCDTs [2]. There will not be only one blockchain, but instead many different coexisting blockchain networks with specific properties leveraged for different use cases. The BCDT architecture addressed this requirement in different ways. Firstly, the BCDT architecture allows private organizations to run internal blockchain nodes for their private blockchain network to enhance internally trusted data sharing. Moreover, these various private blockchains can connect to the consortium and public blockchain in interoperable

ways [16] to enable connectivity between ecosystems of DDTC [1]. As such coopeting organizations can share format agnostic transactional data with trust. This trusted data sharing can enhance collaboration in CI 4.0, reduce data silos, and enable blockchain-based incentivization mechanisms for stakeholders to share information and create value for projects—and for CI 4.0 as a whole—instead of holding information in an adversarial way. Finally, the proposed BCDT framework allows for interoperability protocols for public blockchain to be able to transact together. Indeed, the EVM-compatible public blockchain networks presented in Section 5.3 could interact with each other through bridges allowing transactions between two different networks. Moreover, the Moonriver [43] networks include interoperability by design, since they are respective parachains of the Kusama and Polkadot [61] networks, which allow for interoperability between all the parachains by design. Indeed, the Polkadot ecosystem comprises a core layer called the relay chain, which is responsible for the security, validation of transactions, and coordination of the network. The parachains are individual and interoperable blockchains networks that are connected to the core relay chain. It should be noted that this type of native interoperability offered by Kusama/Polkadot [61] (or by the Cosmos [62] network via the IBC technology) is preferred compared to bridges between different blockchains, since it is by design more efficient, cheaper, and more secure. Future works should practically implement and test interoperability protocols to evaluate the feasibility and measure their effectiveness for interoperable united ecosystems of BCDTs in CI 4.0.

Data ownership: The proposed BCDT smart-contract framework leveraged NFTs for the tokenization of IP and data ownership. This approach has great potential to enable incentivization for individual participants of a Web 3.0-based CI 4.0 in which stakeholders collaborate with BCDT Level 4 and are rewarded fairly throughout the lifecycle of projects. This study's smart-contract framework—for all BCDT dimensions—leveraged the ERC721 NFT smart-contract standard for data ownership. Firstly, the ERC721 contracts—from which multiple NFTs can be minted—can be assigned to a contract owner who is the main stakeholder allowed to call most functions from the contract. Therefore, data ownership is typically achieved with the ownership of the NFT token containing the datasets being produced by the data originator. More generally, the proposed framework leveraged information as an asset by tokenizing the key project data related to all BCDT dimensions into NFTs. Indeed, datasets recorded on distributed storage systems such as IPFS are linked to the NFT metadata that is recorded on the blockchain. The data creators and owners can then tokenize datasets into NFTs for the key data related to each BCDT dimension. Hence, a data creator becomes the data owner in the digital world by owning the unique NFT containing the information being tokenized. The data ownership can be transferred to other stakeholders by simply transferring the NFT to a new data owner on the blockchain. This transfer of ownership can either be monetized so that the information as an asset is sold as a service or be automatically transferred to a new owner after a legally defined period of time. The BCDT architecture and smart-contract frameworks proposed in this paper leveraged NFTs to tokenize data ownership and enable data owners to trade tokenized information on decentralized (blockchain-based) data marketplaces dApps. However, there were limitations to this approach, since the survey results on the monetization of IP and data ownership on decentralized data marketplaces were mitigated. Hence, further research should evaluate in detail the industry requirements regarding the monetization of IP and data ownership in a decentralized CI 4.0.

Decentralization: A previous study on the DDTC [1] identified decentralization as a critical gap area for BCDTs in CI 4.0. Moreover, the decentralization and scalability of data storage are key NFRs for BCDT applications [2]. They are essential to achieving decentralized blockchain protocols and decentralizing BCDT infrastructures. Decentralization is a core property of BCT that affects the blockchain trilemma [35], which signifies that a blockchain network needs to compromise either decentralization, scalability, or security [35]. The survey results from this study about the blockchain trilemma suggested that security was the most important, decentralization was the second most important, and scal-

ability was the least important NFR to the CI. Decentralization is key to guaranteeing data integrity and tamper-proof resistance of historical data records. The proposed BCDT framework embraced decentralization by leveraging decentralized architecture components such as dApps, public blockchains, distributed storage, distributed computing, decentralized oracles, and IoT protocols. Moreover, the proposed smart-contract framework leveraged NFTs and decentralized data storage systems such as IPFS [48] and Filecoin [63].

The survey data revealed that the governance of projects should be decentralized and more democratic. The proposed architecture integrated a decentralized governance model for distributed ecosystems of BCDTs in which participants collaborated in accordance with the BCDT maturity Level 4 [2]. Despite the requirement to democratize governance, the project participants should be able to vote on governance decisions depending on their level of authority and governance weight.

Regarding the decentralization of digital identities, as mentioned in previously, digital-identity systems must remain decentralized to ensure that individuals are self-sovereign in their identities, finances, data, decisions, and human rights in general. The fundamental paradigm shift of BCT and Web 3.0 is to create trustless, decentralized systems that do not require trusting centralized third parties such as data custodians and centralized organizations in general. The Web 3.0 paradigm shift redistributes power and control to individuals instead of requiring trust in centralized third parties. Finally, Web 3.0 technologies enable a future CI 4.0 in which stakeholders can collaborate in P2P ways according to the BCDT maturity Level 4. BCT ensures that collaboration and coopetition in CI 4.0 can operate with trust, efficiency, and enhanced data sharing. CI 4.0 will comprise ecosystems of united BCDT dApps and DAOs that are interoperable and can exchange value throughout project lifecycles and enable better energy management and a decentralized circular economy through the reuse and recycling of materials.

Scalability: The survey results regarding the blockchain trilemma suggested that scalability was the least important NFR for BCDTs in CI 4.0 compared to security and decentralization, which were more crucial. It was clear that the BCDTs required the processing of large volumes of transactions for all the key transactional data related to all the BCDT dimensions throughout the project's lifecycle. The scalability requirements of this paper's use case can be addressed by efficient public blockchain protocols, which nowadays are able process up to several thousand transactions per second. For much larger projects in which the number of transactions required for BCDTs would be much higher than the numbers of this sample project, there would be ways to address the demand with layer-2 scaling solutions. Future research should evaluate the maximum scalability requirements that BCDT ecosystems require and prove that state-of-the-art BCT systems can address them. Considerations around scalability are discussed further in the cost-analysis discussion in chapter 6.3.

6.3. Cost Analysis, Performance, and Blockchain Criteria

This section describes the learning (Step 5) phase of the action research process. This final step consisted of identifying key findings from the cost analysis of the proposed BCDT smart-contract framework, discussing the results, and providing recommendations for future research to refine and develop the framework. Our previous study on the DDTC [1] revealed key gaps affecting BCT adoption for DT. As shown in Table 5, a gap area was related to the technological requirements for BCT: network governance, type of blockchain (private/consortium/public), scalability limitations, decentralization, interoperability, and energy efficiency. This section discusses the cost analysis of the use case presented in Section 5. The results were analyzed in conjunction with the gaps and the limitations imposed by the blockchain trilemma, which stipulates that a blockchain network needs to compromise either decentralization, scalability, or security [35]. With this in mind, this section aims to develop criteria to evaluate the adequacy of blockchain networks for BCDT dApps.

The results of the cost analysis were based on the gas cost at the time of writing. The gas cost—for the public blockchains included in the cost comparison—varied significantly up or down on a daily basis. Hence, the results of the cost analysis were only illustrative and indicative and should not be taken as constant definitive data. Future research works should focus on cost efficiency and further develop the cost analysis by providing some upper and lower price boundaries that should be considered for each type of blockchain network. As the costs were compared between the several public blockchain networks included in Section 5, a trade-off analysis was carried out to justify the prices differences, which varied significantly between different networks. The trade-off analysis considered the blockchain trilemma; for example, the Ethereum network led to a very high execution cost compared to the other networks. However, the Ethereum network is the most secure and decentralized among all the networks proposed, but has the slowest scalability in terms of transaction throughput. It is likely that the level of security offered by the Ethereum network is excessive for most of the data related to the BCDT project lifecycle. However, the level of security offered by a very resilient network such as Ethereum could be required for highly important transactional data involving large financial amounts, critical safety information, non-repudiable regulatory data, or high-importance environmental information. However, most of the other transactional data related to the BCDT dimensions could be transacted using more affordable blockchains such as Avalanche, Polygon, or Arbitrum, which provide a high level of security while currently being more scalable in terms of transaction throughput. There could also be a hybrid approach in which a BCDT ecosystem uses different blockchain networks for different purpose depending on the security and scalability requirements for each use case. Future research works should explore this hybrid proposition further and define the exact type of blockchain network specific use cases would require. However, a hybrid approach would cause friction in terms of interoperability to transact assets or data between different blockchains.

In terms of interoperability between blockchain networks, the public blockchains leveraged in this study typically require bridges to transfer assets or data from one blockchain to another. In terms of openness, the cost analysis of the proposed use cases essentially considered public blockchains to embrace the philosophy of a Web 3.0-based decentralized CI 4.0 leveraging open data sharing and a decentralized data value chain. However, future research works on BCDTs leveraging hybrid blockchain networks for privacy purposes should also consider the inclusion of private and consortium blockchains in the cost analysis.

This study aimed to develop criteria to better compare blockchain networks based on the requirements of BCDTs in terms of security, decentralization, scalability, and interoperability. The survey results on the blockchain trilemma revealed that security was the most important requirement, decentralization was the second most important, and scalability was the least important of the three for BCDT applications in CI 4.0. Hence, a technical content analysis was carried out to identify the security, scalability, and decentralization characteristics of the EVM-compatible public blockchain networks compared in Section 5 (Ethereum, Avalanche, Fantom, Polygon, BNB, Arbitrum, xDAI, and Moonriver). Key factors and metrics were identified to evaluate the security, scalability, and decentralization of these networks. The security characteristic was evaluated based on the resilience of a blockchain network consensus mechanism against a maximum number of malicious nodes (measured in percentage of a total number of nodes). The scalability characteristic was evaluated based on the blockchain network throughput in transactions per second (tps). The decentralization characteristic was evaluated based on the total number of validator nodes in the network. The results of this technical content analysis are given in Table 6. The proposed scales used in Table 6 to categorize throughput and decentralization—as shown in notes 2 and 3 of Table 6—were defined by comparing the characteristics of the key public blockchain networks, discussing them with blockchain experts, and referring to the technical literature on the performances of blockchain systems [64].

Table 6. Comparison of the public blockchain systems (presented in Section 5.3) in relation to the blockchain trilemma properties (security, throughput, and scalability) for BCDT.

Blockchain Network	Layer Level	Consensus Security [1]	Throughput [1,2] (tps)	Decentralization [1,3] (Nodes)
Ethereum	Layer-1	Highest security level as the main layer-1 blockchain. The current Ethereum 1.0 PoW chain requires more than 50% of honest nodes to reach a consensus. The upcoming Ethereum 2.0 requires 2/3 of the active validators to reach consensus.	Currently low throughput of about 14 tps [65] with Ethereum 1.0. Throughput will increase significantly with sharding on Ethereum 2.0.	High decentralization with currently 6354 [66] PoW nodes on Ethereum 1.0 and 303,591 [67] PoS nodes on the Ethereum 2.0 beacon chain.
Avalanche	Layer-1	The Avalanche subsampled voting consensus mechanism is highly resilient, with a safety threshold of 80% malicious validators required to attack the system [68].	Intermediate throughput with 4500 tps per public subnet [68].	The 1240 Avalanche validators give a high level of decentralization [69].
Fantom	Layer-1	The Fantom Lachesis consensus mechanism enables 1–2 s finality (vs. 10 min on Ethereum) and sSupports 1/3 of faulty nodes, including malicious behavior [70].	High throughput [70] with about 4500 tps.	The 78 Fantom validators (Lachesis consensus mechanism) give an intermediate level of decentralization [71].
Polygon	Layer-2	Polygon security layer provides the same level of security as the Ethereum layer-1 into which it is anchored. Polygon leverages a PoS system; if more than 2/3 of the active validators reach consensus, the checkpoint is submitted to the Ethereum mainnet [72].	The Polygon network layer offers a high throughput up to 10,000 tps on a side chain [72].	The 100 Polygon validators give an intermediate level of decentralization [72].
BNB	Layer-1	Delegated proof of stake or proof of staked authority (PoSA) consensus mechanisms [73].	Low throughput with 80 tps [74].	The 21 BNB validators provide a low level of decentralization [73].
Arbitrum	Layer-2	Optimistic rollup enabling trustless security rooted in Ethereum layer-1 with any one party able to ensure correct layer-2 results [41].	The contracts' computation and storage are taken off of the main Ethereum chain, allowing for a high throughput of up to 4500 tps [41].	The optimistic rollup ensures that only one honest active validator is needed [41]. A high level of decentralization is inherited from the Ethereum layer-1.
xDAI	Layer-1	Delegated PoS consensus mechanism that is Byzantine fault tolerant; i.e., consensus can be reached despite 1/3 of failed/malicious nodes [42].	Intermediate throughput with a theoretical maximum of 119 tps [42].	The 19 xDAI/Gnosis validators provide a low level of decentralization [42].
Moonriver	Layer-1	Leverage the security of the Kusama/Polkadot relay chain using a nominated proof of stake (PoS) consensus mechanism requiring 2/3 of honest validators [75].	High throughput [75] with about 1000 tps.	Currently running with 297 validators but aims to have 1000 validators [75]. Polkadot (will) provide a (high) intermediate level of decentralization.

[1] Data obtained at the time of writing from various sources (whitepapers, technical documentations, and other online sources as cited). These data will need to be revised as these blockchain protocols evolve over time. [2] Scalability/throughput categorization in transactions per second (tps): Low throughput < 100 tps; 100 tps ≤ intermediate throughput < 1000 tps; 1000 tps ≤ high throughput. [3] Decentralization categorization in number of nodes: Low decentralization < 30 nodes; 30 nodes ≤ intermediate decentralization < 1000 nodes; 1000 nodes ≤ high decentralization.

The evaluation and characterization—based on security, throughput, and decentralization—of public blockchain networks, as shown in Table 6, permitted the creation of nine key categories based on the level of decentralization and scalability. As the survey results on the blockchain trilemma revealed that decentralization was more important than scalability for BCDTs, the categories were numbered from 1 to 9 (where 1 was the most preferred configuration and 9 was the least preferred configuration for BCDTs), as shown in Figure 6.

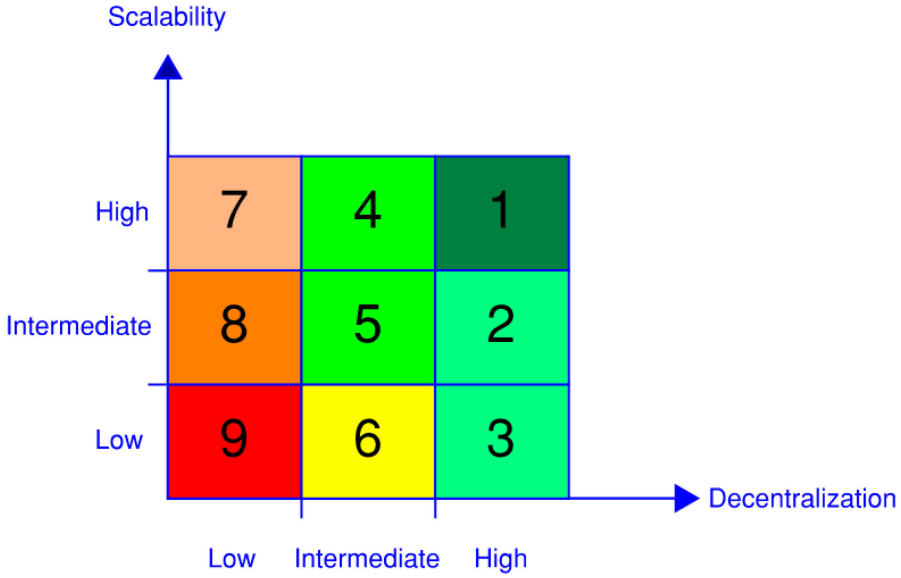

Figure 6. Categorization to evaluate blockchain networks' suitability for BCDTs.

The combination of the data from Table 6 and Figure 6 allowed us to populate the first column of Table 7, called the BCDT category, for each blockchain network based on its scalability and decentralization capabilities. Similarly, a numbering system was used to categorize key properties such as consensus security, interoperability, and cost. The consensus security was rated from 1 to 3, where 1 represented the most secured protocols. Ethereum [22] is the largest and most decentralized network, and its security score was inevitably set to 1. The Avalanche network [68] consensus security was also set to 1 due to the safety threshold of up to 80% for its subsampled voting consensus mechanism compared to the typical 33% for PoS and 50% for PoW. The security grade of the Arbitrum protocol was also set to 1 because only one validator was needed to witness the optimistic rollup that enabled trustless security rooted in the Ethereum layer-1 [41]. Other networks typically have their security score set to 2, and the networks with a low level of decentralization have their security score set to 3. Hence, this approach permitted us to populate the second column of Table 7. The interoperability of these networks was rated from 1 to 2, where 1 represented native interoperability, as with the Moonriver [43] network; and 2 indicated that interoperability with other networks could be achieved with bridges, as discussed in the previous chapters. Finally, the cost was categorized based on the distribution of the cost-analysis results in Table 4 and according to a 7-point Likert scale as presented in the notes of Table 7. Finally, the score numbers of the BCDT category, consensus security, interoperability, and cost were summed for each network to obtain the total BCDT suitability, as shown in the last column of Table 7.

Table 7. Comparison of the public blockchain use cases in relation to key NFRs for BCDTs.

Blockchain Network [1]	BCDT Category [2]	Consensus Security	Interoperability	Cost [4]	BCDT Suitability
Ethereum	3	50%/33% (1)	Bridges (2)	Very high (7)	13
Avalanche	1	80% (1)	Bridges (2)	Intermediate (4)	8
Fantom	4	33% (2)	Bridges (2)	Somewhat low (3)	11
Polygon	4	33% (2) [3]	Bridges (2)	Low (2)	10
BNB	9	33% (3)	Bridges (2)	Intermediate (4)	18
Arbitrum	1	100% (1) [3]	Bridges (2)	Intermediate (4)	8
xDAI	8	33% (3)	Bridges (2)	Very low (1)	14
Moonriver	4	33% (2)	Native (1)	Somewhat low (3)	10

[1] The data in this table were obtained from the content analysis of these state-of-the-art technologies at the time of writing (whitepapers, technical documentations, and other online sources as cited). However, these categories will need to be revised as these protocols evolve over time in terms of scalability, security, interoperability, cost, and decentralization. [2] The BCDT category was derived from the data presented in Table 6 and Figure 6. [3] Polygon, Arbitrum, and layer-2 scaling solutions were running on top of Ethereum and inherited from the security of Ethereum. However, Polygon still had a PoS BFT consensus mechanism requiring 2/3 of the nodes to be honest, whereas Arbitrum only required one honest node. An honest node refers to a node that validates transactions in accordance with the protocol's rules and without trying to forge malicious transactions. [4] The cost scale was categorized based on the cost-analysis data in Table 4 and according to a 7-point Likert scale. The price categories were created so that the blockchain networks analyzed in this paper with significant cost difference—greater than 35%—would fall in separate categories. Hence, the 7-point cost Likert scale chosen was as follows: USD 0 $\leq$ very low (1) <USD 100, USD 100 $\leq$ low (2) <USD 5000, USD 5000 $\leq$ somewhat low (3) <USD 10,000, USD 10,000 $\leq$ intermediate (4) <USD 75,000, USD 75,000 $\leq$ somewhat high (5) <USD 300,000, USD 300,000 $\leq$ high (6) <USD 1,000,000, USD 1,000,000 $\leq$ very high (7).

Avalanche and Arbitrum protocols came with the best BCDT suitability number of 8 due to their good BCDT category level and consensus security scores, which outweighed their intermediate cost. Indeed, Avalanche and Arbitrum had the highest BCDT category of 1 due to their high level of decentralization and throughput. Moreover, Avalanche and Arbitrum had a consensus security score of 1 due to the resilience of their consensus mechanisms, as discussed in the previous paragraph. Hence, we noticed that the best BCDT category (equal to 1) and, more generally, BCDT suitability (equal to 8) were typically related to blockchain networks with the highest level of security and decentralization—such as Avalanche and Arbitrum—which was naturally in accordance with the survey results. However, other networks with efficient cost and scalability performances—such as Polygon and Moonriver—also came with a competitive BCDT suitability (equal to 10) due to their low cost and a good level of security. Indeed, while the Polygon network layers leveraged their own PoS consensus mechanism, which was resilient against up to 33% failed nodes, the Polygon layer-2 periodically anchored its states in the Ethereum layer-1 blockchain and inherited from the security of Ethereum. The Moonriver parachain inherited from the security of the Kusama (or Polkadot) relay chain, which provided a high level of security due to the resilience and decentralization of the Polkadot consensus and network. The Moonriver network offered a "Somewhat low" execution cost that was more than the "Low" cost of Polygon. However, the native interoperability of Moonriver made its interoperability score better than Polygon, which required bridges to transact with all other blockchains. Networks with a low level of decentralization, such as BNB Smart Chain or xDAI/Gnosis chain, led to lower BCDT categories and hence lower BCDT suitability. The xDAI/Gnosis chain offered a very competitive cost-performance due to its low execution cost. However, its low level of decentralization still led to a low BCDT suitability of 14. Finally, Avalanche and Arbitrum had the best BCDT suitability due to their high level of security and decentralization. Hence, these networks could be the most suitable for BCDT projects requiring a high level of security, such as important infrastructure projects for hospitals, heritage, transport, defense, and more generally, government projects. Moreover, if we considered a medium-sized building similar to the advanced sample model studied in Section 5, the cost of execution with a decentralized oracle for Avalanche (USD 59,387) or Arbitrum (USD 72,690) appeared manageable over the lifecycle of a high-importance

government-funded infrastructure project. For smaller projects—such as residential or commercial buildings—cost-efficient and secure blockchain solutions such as Polygon, Moonriver, or Fantom could be leveraged. Similarly, if we considered a medium-sized building similar to the advanced sample model studied in Section 5, the cost of execution with a decentralized oracle for Polygon (USD 6414), Moonriver (USD 10,082), or Fantom (USD 13,373) appeared manageable over the lifecycle of the project. Finally, as mentioned previously, the Ethereum layer-1 blockchain in its current state appeared to be unacceptably too expensive (USD 4,782,815) for BCDT projects and would not be adequate, as it would in any case provide an excessive layer of security for most of the BCDT dimensions' transactional data. However, Ethereum could still be leveraged for crucial transactions required to be settled on the most decentralized and secure blockchain network. Furthermore, if a project leveraging BCDTs has a limited budget, it should be noted that not all BCDT dimensions' transactional data are required to be on a public blockchain due to privacy requirements. For example, if a project only requires anchoring the 7D and 8D transactional data onto a public blockchain, the final execution cost would be significantly less than if transactional data for all BCDT dimensions were recorded on the public blockchain (as shown on Table 4). It should be noted that Table 7 does not include privacy in the scoring system because all the EVM-compatible blockchains leveraged in this study were public and typically would require additional privacy layers to enable private transactions. However, the Avalanche [37] subnets can be a private, consortium, or public blockchain. Moreover, Polygon Edge and Polygon Nightfall [39] enable privacy for transactions. Future research works could test private or consortium blockchain systems for the BCDT operations that require privacy, such as confidential design data, confidential payments or contracts, or communications, as suggested by the survey results. However, using private blockchains would likely cancel the benefits of leveraging NFTs as open digital assets representing tokenized information for greater data sharing. Hence, public blockchains leveraging encryption mechanisms and/or privacy protocols such as the Phala Network [76] (on Polkadot) could also be used to fulfill the privacy requirements. Future research works should test the practical implementation of privacy layers and protocols for the requirements of BCDT applications. This study embraced the openness of public blockchains to improve data sharing in the context of the BCDT maturity level 4 [2]. Hence this study focused essentially on eight EVM compatible public blockchain networks, but future research works should examine other efficient layer-1 blockchain networks such as Solana [77], Cardano [78], Elrond [79], Algorand [80], Hedera [81], Internet Computer [82], and Tezos [83] and evaluate their suitability for BCDTs.

Table 4 reveals that the integration of a decentralized oracle added a constant price for all networks. This led to a moderate cost increase of less than 10% for the Avalanche and Arbitrum networks and a significant cost increase for the Polygon and Moonriver networks due to their low initial costs without an oracle. However, these cost increases were outweighed by the advantages brought by using a decentralized oracle in terms of data authenticity, data integrity, and a reduction in GIGO effects. Moreover, if a decentralized oracle system is widely used for a BCDT project, its price can become degressive through strategic partnering and sponsoring. Future research works should focus on comparing various decentralized oracles systems such as Chainlink [53], Band Protocol [84], API3 [85], and DIA [86] in terms of costs; security; and data authenticity, scalability, and decentralization.

There were limitations to the cost analysis proposed in this study. Firstly, the sample use-case model leveraged in Section 5 might not have been sufficiently accurate in quantifying the elements tokenized into NFTs. Hence, future research could leverage real-world projects for more realistic quantifications and cost analysis. Secondly, the proposed BCDT smart-contract framework mainly leveraged the ERC721 NFT standard [19], leading to excessive gas consumption, particularly due to the minting process that created new NFTs. The proposed smart-contract framework required minting one NFT for each element and for each discipline. Future research works should optimize the framework to reduce gas

consumption and potentially limit the use of NFTs for key assets only instead of leveraging NFTs for most of the BCDT dimensions. For example, a more cost-efficient way would be to have one single NFT for each element and cover as many disciplines and dimensions as possible with a single NFT. As a result, a single set of NFTs would cover several possible discipline packages for several BCDT dimensions and could be transferred accordingly to the relevant stakeholders of the BCDT dimension at each corresponding phase of the project lifecycle. For example, once the design phase is finished, the 3D design NFT of an element can be transferred from the designer (3D) to the next stakeholders of the project data value chain, such as the quantity surveyor (5D), the manufacturer (4D), the supplier (4D), the builder (4D), the sustainability manager (7D), and eventually the facility manager (6D). Each stakeholder would be responsible for updating the NFT metadata by adding the metadata relevant to their discipline and BCDT dimension. This approach would reduce the number of NFT smart-contract sets to be deployed and hence has the potential to reduce the gas consumption of the framework throughout the lifecycle of a project. However, this approach would require transferring many NFTs between multiple stakeholders using the transferFrom function, which would cost a significant amount of gas. It should be noted that the cost analysis carried out in this paper did not estimate the gas cost related to the transfer of NFTs after they were minted. Indeed, the transfer of NFTs between project stakeholders was too complex to estimate and hence was excluded from the scope of the cost analysis, which only focused on the deployment and key functionalities of the proposed smart-contract framework for BCDTs. However, for the 7D sustainability dimension, the cost of transferring NFTs was accounted for to explicitly trace dominant green assets such as materials and reduce wastes throughout the full lifecycle until they could be reused and recycled for the CE. Moreover, the transfer of NFTs would depend on specific project requirements and the gas cost to transfer the tokens would typically be covered by individual NFT owners and hence could fall out of the generic cost comparison scope carried out in this study. In addition, the cost analysis was not fully accurate, as some estimations had to be made for metrics such as the maximum number of zones, the maximum number of smart assets per zone, the assumed 25 years of service-design life, the number of risks per package, and the number of times the factory-contract ownership was transferred. Moreover, the cD legal smart-contract use case only considered maintenance contracts for the O&M phase, whereas in reality, there would be more legal contracts throughout each phase of the project and each of the BCDT dimensions, since the cD dimension overarches the other BCDT dimensions, as discussed in Section 6.1. Hence, the cD smart-contract gas consumption would potentially be significantly higher if all contractual processes of the project leveraged public blockchains. Another approach to reducing gas costs would be to leverage layer-2 scaling solutions for NFTs such as ImmutableX [87], which uses ZK-rollups technology. This would allow the inheritance of the layer-1 Ethereum blockchain's security and leverage the efficient scalability of the ImmutableX protocol, which can process more than 9000 transactions per second (tps). This would allow project participants to mint and trade NFTs without any gas cost and would reduce the carbon footprint of the BCDT ecosystem. However, to achieve this, the smart-contract format would require compliance with the API requirements of the layer-2 solution that is used. Another limitation of the cost analysis was that it did not include the IT infrastructure costs related to the adoption of the BCDT framework. Indeed, the study only evaluated the operation costs of deploying and interacting with the BCDT smart contracts. Hence, future research should evaluate the cost of deploying the BCDT framework for a project from an infrastructure point of view.

This study had positive implications for environmental sustainability. Indeed, Table 6 shows that the consensus mechanisms proposed by the public blockchain networks typically are proof of stake (PoS) consensus mechanisms that are considerably more energy-efficient than proof of work (PoW) mining. At the time of writing, Ethereum was migrating from a PoW- to a PoS-based consensus mechanism called Casper [88]. Hence, the framework of smart contracts and blockchain protocols used in this study addressed the DDTC [1] gap area regarding the energy efficiency of consensus mechanisms to positively contribute

to the environmental footprint of BCDT systems. Finally, another implication of this study was that it offered CI 4.0 practitioners a categorization tool to evaluate the suitability of blockchain networks for BCDT dApps. However, there were limitations with the categorization proposed by this paper to evaluate the suitability of blockchain networks for BCDT applications. Indeed, the categorization proposed by the study for security, decentralization, and interoperability was relatively simple, whereas the underlying blockchain properties were very complex. For example, decentralization depended not only on the number of nodes, but also on the geographic location of the nodes [89]. Hence, further research should refine the categorization criteria proposed in this paper and identify the acceptance thresholds to measure the exact degree of decentralization, security, and scalability required for BCDT projects.

Finally, Table 8 summarizes the suggested directions for future research works on BCDT applications.

Table 8. Summary of future directions for research works.

Future Research Themes	Future Research Directions	BCDT Dimension
BCDT framework	Develop the proposed BCDT framework further. Identify further design requirements for BCDT smart contracts.	All
BCDT architecture	Practical implementation of the other of the BCDT architecture layers (front end, computer, Big Data management, and the sensing layers).	All
	Identify further requirements about what sensors data exactly require to be recorded on-chain.	All
	Practical implementations of decentralized and scalable storage for large volumes of data.	All
	Hybrid architecture leveraging different blockchain networks adequate for specific use cases requirements.	All
	Explore other scalable layer-1 blockchain networks and evaluate their suitability for BCDTs.	All
	Compare various decentralized oracles systems in terms of cost, security, data authenticity, scalability, and decentralization.	All
Identity	Integration of SSI for the cD dimension to decentralized digital identities for stakeholders and devices from BCDT ecosystems.	cD
	Leverage existing decentralized identity frameworks (e.g., ERC-725 standard)	cD
Supply chain	Explore blockchain-based supply chain solutions for BCDTs in CI 4.0.	All
DeFi	Decentralized project banks solutions leveraging DeFi applications for BCDTs.	5D
	Smart contracts framework to automate financial services in CI 4.0.	5D
NFT	Explore resilient systems that leverage NFTs to represent physical assets.	4D, 5D, 6D, 7D
	Optimize the BCDT smart-contract framework to reduce gas consumption.	All
	Streamline smart-contract logic between BCDT dimensions to reduce the quantities of NFTs throughout the lifecycle of projects.	All
IoT	Develop models to minimize on-chain records of IoT data.	All
	Leverage multiple oracle nodes to strengthen data authentication and improve integrity.	All
	Leverage decentralized oracles to enhance data security and management for IoT systems.	All
	Implement systems to decentralize the management of IoT networks.	All
	Leverage microchips and secure elements to enable device identities at the BCDT edge layer.	All
Environment	Carbon-footprint analysis of the BCDT framework to measure the environmental benefits.	7D
	Leverage renewable energy and excess energy from smart cities to run blockchain nodes and validate public transactions for the public good and the environment, and to generate additional revenues and new business models.	5D, 7D

Table 8. *Cont.*

Future Research Themes	Future Research Directions	BCDT Dimension
Safety	Develop the BCDT smart-contract framework in relation to safety.	8D
Regulation	Automation of work orders with legal smart contracts.	All
Regulation	Integration of a regulatory framework to the BCDT cD smart contracts and provide industry practitioners and regulators with standardized legal smart-contract templates.	cD
	Development of smart-contract standards in the CI for the automation of regulatory processes.	All
	Develop the BCDT cD dimension smart-contract framework further to integrate key aspects such as regulatory compliance, accountability, data ownership, IP protection, SSI, and decentralized governance.	cD
	Decentralize regulatory compliance to improve efficiency and compliance.	All
Governance	Incentivization mechanisms to democratize governance and allow stakeholders to vote on governance decisions.	All
	Distributed collaboration and data sharing leveraging decentralized governance and decentralized autonomous organizations (DAO).	All
Access control	Develop access-control mechanisms for distributed collaboration in CI 4.0.	All
Privacy	Evaluate privacy protocols, encryption mechanisms, and private or consortium blockchain networks to address the privacy requirements of BCDTs.	All
	Explore, test, and evaluate privacy systems for BCDTs leveraging public blockchain networks.	All
	Identify further which data exactly from organizations should be auditable on public blockchains. Identify all the privacy requirements and how they fit with decentralization and Web 3.0.	All
Interoperability	Test interoperability protocols to achieve united ecosystems of BCDTs.	All
Scalability	Identify further the scalability requirements for BCDT systems.	All
Cost	Cost analysis to identify cost-efficient BCDT systems.	5D
	Cost analysis for BCDTs systems leveraging hybrid blockchain networks.	5D
	Exhaustive cost analysis for deploying a BCDT system for a project. Generate additional revenue streams for projects through tokenization and NFTs.	5D

7. Conclusions

This paper aimed to address critical CI problems, key requirements, and five essential literature gaps in the related works regarding BCDTs. The paper also aimed to propose a BCDT software architecture and smart-contract framework, as well as to carry out a cost analysis and develop criteria for evaluating blockchain protocols that can be leveraged for the proposed BCDT smart-contract framework. The interviews also permitted us to identify—for each of the problems—the related FRs and NFRs of BCDT applications. The problems identified for the BCDT dimensions were related to the notarization of key design data and automation of processes such as Q&A, inspections, and certification (3D); the traceability of the construction supply chain, as-built compliance, and automation of installation inspections, assessments, and certification (4D); the decentralization and automation of tendering, the notarization of the financial supply chain, the automation of payments, and the creation of digital assets through tokenization, incentivization mechanisms, and protection of data ownership and IP, and the authentication of pricing data (5D); the improvement of asset management and notarization of smart-asset records (6D); the management of energy, notarization of environmental data, and enablement of the circular economy (7D); the identification and mitigation of risks (8D); and the automation of regulatory processes and

enforcement of compliance with legal smart contracts (cD). Seven basic industry uses cases were designed for each BCDT dimension to address the problems identified. The analysis of data from an online survey permitted the development of a software architecture for BCDT dApps and to refine the design of smart contracts by leveraging NFTs to address the requirements of the developed use cases for each BCDT dimension. The present paper addressed the literature gaps related to the key technical requirements of BCT (governance, type of blockchain network, scalability, decentralization, interoperability, energy efficiency, and computational requirements), the integration of the IoT with BCT, the integration of BIM with BCT, the integration of DT Big Data with BCT, and the integration of BCDTs throughout complex lifecycles for the circular economy. Moreover, the proposed BCDT architecture and smart-contract framework addressed the requirements of BCDTs because it complied with the BCDT maturity Level 4 framework and addressed the key NFRs for BCDT applications. Indeed, the proposed framework contributes to enabling distributed collaboration in CI 4.0 by leveraging resilient, open blockchain networks, facilitating the automation of key processes with smart contracts for all BCDT dimensions and enhancing trusted data sharing in a decentralized data value chain. This paper also presented a gas cost comparison analysis between different EVM-compatible public blockchains (Ethereum, Avalanche, Fantom, Polygon, BNB, Arbitrum, xDAI, and Moonriver) in order to measure their cost differences in the execution of the proposed BCDT smart-contract framework throughout the lifecycle of a medium-sized building project. The results of the cost comparison analysis permitted us to further compare blockchain networks in relation to the fundamental properties of the blockchain trilemma (security, decentralization, and scalability). Finally, the criteria developed in this study revealed that secure and decentralized networks such as Avalanche and Arbitrum could be leveraged to secure BCDT transactional data for sensitive infrastructure projects such as defense, healthcare, heritage, or transport, whereas less decentralized but more cost-efficient networks such as Polygon, Moonriver, or Fantom could be leveraged for smaller-scale projects such as residential or commercial buildings.

Author Contributions: Conceptualization, B.T.; methodology, B.T. and S.M.E.S.; software, B.T.; validation, B.T.; formal analysis, B.T.; survey design, B.T. and S.M.E.S.; investigation, B.T.; resources, B.T.; data curation, B.T.; writing—original draft preparation, B.T.; writing—review and editing, B.T. and S.M.E.S.; visualization, B.T.; supervision, B.T. and S.M.E.S.; project administration, B.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used is presented in this paper.

Acknowledgments: The authors of this paper acknowledge the support of the Australian Government in this research. The authors express their gratitude to the participants who agreed to be interviewed and answered the online survey, and who kindly shared their valuable insights on the research subject. The authors thank the researchers at CSIRO's Data61 for their very valuable technical advice and for the "software architecture for blockchain applications" course at the University of New South Wales that provided valuable insights in the preparation of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

API	Application programming interface
B2B	Business-to-business
B2C	Business-to-consumer
BAS	Building automation system
BIM	Building information modeling
BMS	Building management system
BCT	Blockchain technology

BCDT	Blockchain-based digital twin
CAD	Computer-aided design
CDE	Common data environments
CE	Circular economy
CI	Construction industry
DAO	Decentralized autonomous organizations
dApps	Decentralized applications
DCDE	Decentralized common data environments
DeFi	Decentralized finance
DT	Digital twin
EVM	Ethereum virtual machine
FM	Facility manager
GIGO	Garbage in garbage out
GIS	Geographic information system
IPFS	InterPlanetary File System
IT	Information technology
NFT	Non-fungible token
O&M	Operation and maintenance
P2P	Peer-to-peer
PoS	Proof of stake
PoW	Proof of work
Q&A	Quality and assurance
RFID	Radio-frequency identification
SSI	Self-sovereign identity
UI	User interface
USD	United States dollar

References

1. Teisserenc, B.; Sepasgozar, S. Adoption of Blockchain Technology through Digital Twins in the Construction Industry 4.0: A PESTELS Approach. *Buildings* **2021**, *11*, 670. [CrossRef]
2. Teisserenc, B.; Sepasgozar, S. Project Data Categorization, Adoption Factors, and Non-Functional Requirements for Blockchain Based Digital Twins in the Construction Industry 4.0. *Buildings* **2021**, *11*, 626. [CrossRef]
3. Bhushan, B.; Khamparia, A.; Sagayam, K.M.; Sharma, S.K.; Ahad, M.A.; Debnath, N.C. Blockchain for smart cities: A review of architectures, integration trends and future research directions. *Sustain. Cities Soc.* **2020**, *61*, 102360. [CrossRef]
4. Zheng, R.; Jiang, J.; Hao, X.; Ren, W.; Xiong, F.; Ren, Y. bcBIM: A Blockchain-Based Big Data Model for BIM Modification Audit and Provenance in Mobile Cloud. *Math. Probl. Eng.* **2019**, *2019*, 5349538. [CrossRef]
5. Tezel, A.; Papadonikolaki, E.; Yitmen, I.; Hilletofth, P. Preparing construction supply chains for blockchain technology: An investigation of its potential and future directions. *Front. Eng. Manag.* **2020**, *7*, 547–563. [CrossRef]
6. Construction Blockchain Consortium. Blockchain & Construction Cash Flow. Available online: https://static1.squarespace.com/static/58b6047520099e545622d498/t/5fdb6089ad5a0604f7feaf5e/1608212649913/CBC2020-WP1_Cashflow.pdf (accessed on 6 January 2022).
7. Guerar, M.; Verderame, L.; Merlo, A.; Migliardi, M. Blockchain-based risk mitigation for invoice financing. In Proceedings of the 23rd International Database Applications & Engineering Symposium, Athens, Greece, 10–12 June 2019; pp. 1–6.
8. Hamledari, H.; Fischer, M. Role of blockchain-enabled smart contracts in automating construction progress payments. *J. Leg. Aff. Disput. Resolut. Eng. Constr.* **2021**, *13*, 04520038. [CrossRef]
9. Li, J.; Greenwood, D.; Kassem, M. Blockchain in the built environment and construction industry: A systematic review, conceptual models and practical use cases. *Autom. Constr.* **2019**, *102*, 288–307. [CrossRef]
10. Lund, E.H.; Jaccheri, L.; Li, J.; Cico, O.; Bai, X. Blockchain and sustainability: A systematic mapping study. In Proceedings of the 2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), Montreal, QC, Canada, 27 May 2019; pp. 16–23.
11. Ye, Z.; Yin, M.; Tang, L.; Jiang, H. Cup-of-Water theory: A review on the interaction of BIM, IoT and blockchain during the whole building lifecycle. In Proceedings of the 2018 International Symposium on Automation and Robotics in Construction, Berlin, Germany, 20–25 July 2018; pp. 1–9.
12. McNamara, A.J.; Sepasgozar, S.M. Intelligent contract adoption in the construction industry: Concept development. *Autom. Constr.* **2021**, *122*, 103452. [CrossRef]
13. García de Soto, B.; Georgescu, A.; Mantha, B.; Turk, Ž.; Maciel, A. Construction Cybersecurity and Critical Infrastructure Protection: Significance, Overlaps, and Proposed Action Plan. *Preprints* **2020**, 2020050213. [CrossRef]
14. Autodesk. Digital Twins for a Physical World. Available online: https://www.autodesk.com/autodesk-university/class/Digital-Twins-Physical-World-2019 (accessed on 23 January 2022).

15. Vacca, A.; Di Sorbo, A.; Visaggio, C.A.; Canfora, G. A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges. *J. Syst. Softw.* **2021**, *174*, 110891. [CrossRef]
16. Ali, M.S.; Vecchio, M.; Pincheira, M.; Dolui, K.; Antonelli, F.; Rehmani, M.H. Applications of blockchains in the Internet of Things: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 1676–1717. [CrossRef]
17. Putz, B.; Dietz, M.; Empl, P.; Pernul, G. Ethertwin: Blockchain-based secure digital twin information management. *Inf. Process. Manag.* **2021**, *58*, 102425. [CrossRef]
18. Hasan, H.R.; Salah, K.; Jayaraman, R.; Omar, M.; Yaqoob, I.; Pesic, S.; Taylor, T.; Boscovic, D. A Blockchain-Based Approach for the Creation of Digital Twins. *IEEE Access* **2020**, *8*, 34113–34126. [CrossRef]
19. OpenZeppelin. ERC721.sol. Available online: https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/token/ERC721/ERC721.sol (accessed on 13 February 2022).
20. Konashevych, O. Constraints and benefits of the blockchain use for real estate and property rights. *J. Prop. Plan. Environ. Law* **2020**, *12*, 109–127.
21. Hunhevicz, J.J.; Motie, M.; Hall, D.M. Digital building twins and blockchain for performance-based (smart) contracts. *Autom. Constr.* **2022**, *133*, 103981. [CrossRef]
22. Vogelsteller, F.; Buterin, V. Ethereum whitepaper. Available online: https://ethereum.org/en/whitepaper/ (accessed on 15 June 2020).
23. Ethereum. Solidity, the Smart Contract Programming Language. Available online: https://github.com/ethereum/solidity (accessed on 28 January 2022).
24. Wood, G. Ethereum Yellow Paper. Available online: https://github.com/ethereum/yellowpaper (accessed on 28 January 2022).
25. Hardhat. Ethereum Development Environment for Professionals. Available online: https://hardhat.org/ (accessed on 28 January 2022).
26. Autodesk. Revit Sample Project Files. Available online: https://knowledge.autodesk.com/support/revit/getting-started/caas/CloudHelp/cloudhelp/2020/ENU/Revit-GetStarted/files/GUID-61EF2F22-3A1F-4317-B925-1E85F138BE88-htm.html (accessed on 20 January 2022).
27. Gudgeon, L.; Werner, S.; Perez, D.; Knottenbelt, W.J. Defi protocols for loanable funds: Interest rates, liquidity and market efficiency. In Proceedings of the 2nd ACM Conference on Advances in Financial Technologies, New York, NY, USA, 21–23 October 2020; pp. 92–112.
28. Pauwels, P. zkKYC: A Solution Concept for KYC without Knowing Your Customer, Leveraging Self-Sovereign Identity and Zero-Knowledge Proofs. Cryptology ePrint Archive, Paper 2021/907. Available online: https://eprint.iacr.org/2021/907.pdf (accessed on 13 February 2022).
29. Ethereum. Ethereum JavaScript API web3.js. Available online: https://github.com/ethereum/web3.js (accessed on 13 February 2022).
30. Xu, X.; Weber, I.; Staples, M. *Architecture for Blockchain Applications*; Springer: Berlin/Heidelberg, Germany, 2019.
31. SLock.it. INCUBED Protocol Documentation. Available online: https://in3.readthedocs.io/en/develop/intro.html (accessed on 8 March 2020).
32. Riddle & Code. The Blockchain Technology Company. Available online: https://www.riddleandcode.com/ (accessed on 17 May 2021).
33. Litentry Foundation Ltd. A Cross-Chain Identity Aggregator. Available online: https://www.litentry.com/ (accessed on 2 September 2021).
34. Preukschat, A.; Reed, D. *Self-Sovereign Identity: Decentralized Digital Identity and Verifiable Credentials*; Simon and Schuster: New York, NY, USA, 2021.
35. Bez, M.; Fornari, G.; Vardanega, T. The scalability challenge of ethereum: An initial quantitative analysis. In Proceedings of the 2019 IEEE International Conference on Service-Oriented System Engineering (SOSE), San Francisco, CA, USA, 4–9 April 2019; pp. 167–176.
36. Cdbb. Gemini Principles. Available online: https://www.cdbb.cam.ac.uk/system/files/documents/TheGeminiPrinciples.pdf (accessed on 18 May 2020).
37. Kevin Sekniqi, D.L.; Buttolph, S.; Sirer, E.G. Avalanche Platform. Available online: https://www.avalabs.org/whitepapers (accessed on 13 February 2022).
38. Fantom Foundation. Fantom Whitepaper. Available online: https://fantom.foundation/research/wp_fantom_v1.6.pdf (accessed on 13 February 2022).
39. Polygon Technology. Ethereum's Internet of Blockchain. Available online: https://polygon.technology/lightpaper-polygon.pdf (accessed on 13 February 2022).
40. Binance. Binance Smart Chain Whitepaper. Available online: https://github.com/binance-chain/whitepaper/blob/master/WHITEPAPER.md (accessed on 13 February 2022).
41. Offchain Labs. Inside Arbitrum. Available online: https://developer.offchainlabs.com/docs/inside_arbitrum (accessed on 13 February 2022).
42. Gnosis Chain (xDAI). Gnosis Chain. Available online: https://www.xdaichain.com/ (accessed on 13 February 2022).
43. Moonbeam Network. Moonbeam Docs. Available online: https://docs.moonbeam.network/ (accessed on 13 February 2022).
44. OpenZeppelin. The Standard for Secure Blockchain Applications. Available online: https://github.com/OpenZeppelin (accessed on 13 February 2022).

45. OpenZeppelin. Ownable.sol. Available online: https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/access/Ownable.sol (accessed on 13 February 2022).
46. Remix. Remix IDE. Available online: https://remix.ethereum.org/ (accessed on 13 February 2022).
47. Ellis, S.; Juels, A.; Nazarov, S. Chainlink: A decentralized oracle network. *Retrieved March* **2017**, *11*, 2018.
48. IPFS. IPFS Docs. Available online: https://docs.ipfs.io/ (accessed on 5 February 2022).
49. Smart Contract Engineer. Merkle Tree. Available online: https://solidity-by-example.org/app/merkle-tree/ (accessed on 17 February 2022).
50. Ethers. ethers.js. 2022. Available online: https://docs.ethers.io/v5/ (accessed on 13 February 2022).
51. Hardhat Gas Reporter. Eth-Gas-Reporter Plugin for Hardhat. Available online: https://github.com/cgewecke/hardhat-gas-reporter (accessed on 20 February 2022).
52. CoinTool. Gas Price. Available online: https://cointool.app/gasPrice/ (accessed on 20 February 2022).
53. Chainlink Market. Data Provider Nodes. Available online: https://market.link/ (accessed on 22 February 2022).
54. *AS/NZS 1170.0:2002*; Structural Design Actions–General Principles. Standards Australia (SA): Sydney, Australia, 2002.
55. Fedak, G.; Wassim, B.; Eduardo, A. iExec: Blockchain-Based Decentralized Cloud Computing. Available online: https://iex.ec/wp-content/uploads/pdf/iExec-WPv3.0-English.pdf (accessed on 3 November 2019).
56. Rose, C.; Stegemann, J.A. Characterising existing buildings as material banks (E-BAMB) to enable component reuse. *Proc. Inst. Civ. Eng.-Eng. Sustain.* **2019**, *172*, 129–140. [CrossRef]
57. Estdale, J.; Georgiadou, E. Applying the ISO/IEC 25010 quality models to software product. In Proceedings of the European Conference on Software Process Improvement, Bilbao, Spain, 5–7 September 2018; pp. 492–503.
58. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 23 October 2019).
59. Vogelsteller, F. ERC-725 Ethereum Identity Standard. Available online: https://erc725alliance.org (accessed on 13 February 2022).
60. Finck, M. *Blockchain and the General Data Protection Regulation*; European Parliament: Strasbourg, France, 2019.
61. Wood, G. Polkadot: Vision for a Heterogeneous Multi-Chain Framework. Available online: https://polkadot.network/PolkaDotPaper.pdf (accessed on 10 November 2019).
62. Kwon, J.; Buchman, E. Cosmos Whitepaper. Available online: https://v1.cosmos.network/resources/whitepaper (accessed on 10 November 2019).
63. Protocol Labs. Filecoin: A Decentralized Storage Network. Available online: https://filecoin.io/filecoin.pdf (accessed on 4 September 2021).
64. Dinh, T.T.A.; Liu, R.; Zhang, M.; Chen, G.; Ooi, B.C.; Wang, J. Untangling blockchain: A data processing view of blockchain systems. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 1366–1385. [CrossRef]
65. Etherscan. The Ethereum Blockchain Explorer. Available online: https://etherscan.io/ (accessed on 5 March 2022).
66. Ethernodes.org. Ethereum Mainnet Statistics. Available online: https://ethernodes.org/ (accessed on 5 March 2022).
67. Beaconcha.in. Open Source Ethereum 2.0 Beacon Chain Explorer. Available online: https://beaconcha.in (accessed on 5 March 2022).
68. Avalanche. Avalanche Multiverse. Available online: https://www.avax.network/ (accessed on 5 March 2022).
69. Avalanche. Avalanche Explorer. Available online: https://explorer.avax.network/ (accessed on 5 March 2022).
70. Fantom. Fantom Foundation. Available online: https://fantom.foundation/ (accessed on 5 March 2022).
71. FTMScan. Fantom Blockchain Explorer. Available online: https://ftmscan.com (accessed on 5 March 2022).
72. Polygon Technology. Polygon Documentation. Available online: https://docs.polygon.technology/ (accessed on 13 March 2022).
73. Binance. Binance Smart Chain Documentation. Available online: https://docs.binance.org/ (accessed on 13 March 2022).
74. BscScan. BNB Smart Chain Explorer. Available online: https://bscscan.com/ (accessed on 5 March 2022).
75. Polkadot. Polkadot Documentation. Available online: https://wiki.polkadot.network/docs/getting-started (accessed on 13 March 2022).
76. Yin, H.; Zhou, S.; Jiang, J. Phala Network: A Confidential Smart Contract Network Based on Polkadot. Available online: https://files.phala.network/phala-paper.pdf (accessed on 5 February 2021).
77. Yakovenko, A. Solana: A New Architecture for a High Performance Blockchain v0. 8.13. Available online: https://solana.com/solana-whitepaper.pdf (accessed on 4 December 2021).
78. IOHK. Cardano. Available online: https://www.cardano.org/ (accessed on 14 March 2022).
79. Elrond. The Internet Scale Blockchain Is Live. Available online: https://elrond.com/ (accessed on 14 March 2022).
80. Algorand. *Algorand Whitepaper*; Algorand: Boston, MA, USA, 2019.
81. Hedera. *Hedera Papers*; Hedera: Richardson, TX, USA, 2021.
82. Hanke, T.; Movahedi, M.; Williams, D. Dfinity technology overview series, consensus system. *arXiv* **2018**, arXiv:1805.04548.
83. Goodman, L. Tezos-White Paper. 2014. Available online: https://tezos.com/whitepaper.pdf (accessed on 13 March 2022).
84. Band Protocol. Secure, Scalable Blockchain-Agnostic Decentralized Oracle. Available online: https://bandprotocol.com/ (accessed on 13 March 2022).
85. API3. The Web3 API Economy. 2022. Available online: https://api3.org/ (accessed on 13 March 2022).
86. DIA. Cross-Chain, Open-Source Oracles for Web3. Available online: https://www.diadata.org/ (accessed on 13 March 2022).

87. Immutable. Immutable X Documentation. 2022. Available online: https://docs.x.immutable.com/docs/developer-faq/ (accessed on 13 March 2022).
88. Buterin, V.; Griffith, V. Casper the friendly finality gadget. *arXiv* **2017**, arXiv:1710.09437.
89. Lee, J.; Lee, B.; Jung, J.; Shim, H.; Kim, H. DQ: Two approaches to measure the degree of decentralization of blockchain. *ICT Express* **2021**, *7*, 278–282. [CrossRef]