buildings

MDPI

Article

A Historical Building Information Modeling-Based Framework to Improve Collaboration and Data Security in Architectural Heritage Restoration Projects

Cong Zhou [1], Xingyao Dong [1,*], Yiquan Zou [1,*], Hao Yang [2], Jingtao Zhi [1] and Zhixiang Ren [1]

[1] School of Civil Engineering, Architecture and Environment, Hubei University of Technology, Wuhan 430068, China; 102200781@hbut.edu.cn (C.Z.); 102100768@hbut.edu.cn (J.Z.); 102300832@hbut.edu.cn (Z.R.)
[2] Central South Architectural Design Institute Co., Ltd., Wuhan 430071, China; csadi153@outlook.com
* Correspondence: jjdxy123@gmail.com (X.D.); zouyq@mail.hbut.edu.cn (Y.Z.)

Abstract: With the increasing awareness of architectural heritage conservation and the development of digital technology, there is an urgent need in the field of architectural heritage restoration for a novel solution that can enhance data security, collaboration efficiency, and file management capabilities. This study proposes an Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) framework based on blockchain and IPFS technologies to address the above challenges. The AHR-DCDE framework significantly improves data security and collaborative efficiency in architectural heritage restoration projects by creating a decentralized collaborative design process that achieves data immutability, traceability, and efficient large-scale file processing capabilities. The AHR-DCDE framework significantly improves data security and collaborative efficiency in architectural heritage restoration projects by creating a decentralized collaborative design process that achieves data immutability, traceability, and efficient large-scale file processing capabilities. In this study, the practicality and effectiveness of the AHR-DCDE framework is verified by taking the heritage restoration design project of Pinghe Packing Factory in Wuhan, Hubei Province, as an example. Evaluation of the framework's network latency, throughput, and storage costs indicates that AHR-DCDE can meet the requirements of architectural heritage restoration projects, possessing efficient capabilities for handling and sharing project data. Furthermore, the implementation of the AHR-DCDE framework also facilitates efficient collaboration among interdisciplinary teams, providing robust technical support for the protection and restoration of architectural heritage.

Keywords: blockchain; collaborative design; HBIM; IPFS; smart contract

check for updates

Citation: Zhou, C.; Dong, X.; Zou, Y.; Yang, H.; Zhi, J.; Ren, Z. A Historical Building Information Modeling-Based Framework to Improve Collaboration and Data Security in Architectural Heritage Restoration Projects. *Buildings* **2024**, *14*, 1431. https://doi.org/10.3390/buildings14051431

Academic Editor: Irem Dikmen

Received: 27 March 2024
Revised: 10 May 2024
Accepted: 13 May 2024
Published: 15 May 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

International organizations, such as the International Council on Monuments and Sites (ICOMOS), the International Centre for the Study of the Preservation and Restoration of Cultural Property (ICCROM), and the United Nations Educational, Scientific and Cultural Organization (UNESCO), have issued a series of charters and conventions aimed at protecting architectural heritage [1–3]. The Venice Charter, issued in 1964, emphasizes the role of historical monuments as carriers of past information, which are precious and vivid witnesses for us [4]. As global recognition of shared human values grows, architectural heritage is increasingly viewed as a common treasure of all humanity. This further strengthens the awareness of our shared responsibility to protect these cultural heritage sites for future generations [5]. Therefore, how to restore and conserve these architectural heritages in a scientific and rational manner has become a problem that needs to be solved through extensive research and practice.

Historical Building Information Modeling (HBIM) is the application of Building Information Modeling (BIM) technology to cultural heritage conservation and restoration projects [6–8]. It enables team members, including restorers of historic buildings, to collaborate by sharing design attributes, changes, and issues. Currently, more and more academics and industry members are developing HBIM collaborative design methods and platforms. The concept of a Common Data Environment (CDE) has been introduced in the latest ISO 19650 [9,10] standard, which provides a single source of information for the entire project team to centrally collect, manage, and disseminate project-related documents, graphical models (BIM models), and non-graphical data (project documentation) [9]. The Common Data Environment (CDE), by defining standardized workflows, segments the project data lifecycle into four stages: Work in Progress (WIP), Shared, Publish, and Archived, offering clear definitions of data states and robust version control capabilities. This approach not only enhances the reusability of data but also significantly improves the efficiency of design collaboration, making the HBIM-based design collaboration process more streamlined [9,10].

The Common Data Environment (CDE) for architectural heritage restoration is specifically designed and developed to support architectural heritage restoration projects. Although existing CDE systems have made some advancements in BIM model conversion [11], collaboration, real-time synchronization [12], and efficient data access, these systems are primarily aimed at conventional BIM model design and not specifically developed for handling complex Historical Building Information Models (HBIMs). Furthermore, these systems generally utilize a central design structure, rendering them highly vulnerable to concerns related to data protection, notably regarding the manipulation of design information and susceptibility to denial-of-service (DoS) attacks. Even though protective strategies such as firewalls [13] and controls on database access [14] are employed to guard against threats from outside, these precautions frequently fail to completely safeguard against internal risks, particularly the alteration of data by ill-intentioned insiders [15]. Therefore, developing a secure, HBIM-based Common Data Environment (CDE) for architectural heritage restoration that can effectively address both internal and external security threats, ensuring data security and integrity, has become an urgent and important issue to resolve.

Blockchain is a novel application model that combines the uniqueness and innovation of computing technologies, such as distributed data storage, decentralized and independent peer-to-peer transactions, automatic and intelligent consensus mechanisms, a programmable smart contract, and dynamic encryption algorithms [16]. *The Economist* compares the blockchain with the “trust machine” and predicts that “the blockchain will redefine the world”. Exploring the adoption of distributed technologies, such as the blockchain, to enhance data security and transparency in CDE is a promising direction. Blockchain technology, as an innovative and promising solution, offers a decentralized, unchangeable, and traceable new approach to data storage and management [17]. Unlike traditional centralized systems, blockchain technology adopts a decentralized architecture, forming a peer-to-peer network [18]. In this network, each participating node can maintain a complete ledger. Through advanced technologies such as cryptographic algorithms, distributed ledgers, and consensus mechanisms, blockchain can ensure the irreversibility and authenticity of data transactions [19]. Currently, blockchain technology has been widely applied in the construction industry, such as in building quality information management [20], supply chain management, and payment management, among various fields [21]. Particularly noteworthy is the combination of blockchain with Historical Building Information Modeling (HBIM) technology, which is valued for its ability to provide a reliable source for HBIM data [22]. In the blockchain, the immutability and traceability of data make it an ideal platform for protecting HBIM data. While blockchain technology exhibits considerable promise for mitigating cybersecurity threats in collaborative projects using Heritage Building Information Modeling (HBIM), its integration into Common Data Environment (CDE) management for the restoration of architectural heritage encounters various obstacles. One major challenge is the uniqueness and complexity of historical buildings, which often leads to large and complex HBIM models. Additionally, the inherent block size limitations of the blockchain also pose a challenge for storing large-scale data, such as HBIM models and

CAD files. Attempting to directly store large HBIM models on the blockchain could lead to network delays and congestion.

The InterPlanetary File System (IPFS) is an innovative, peer-to-peer distributed file system, providing an ideal technological complement to the blockchain for securely storing and distributing large files [23]. Upon uploading a file to the IPFS, it assigns a unique and immutable cryptographic hash, known as the Content Identifier (CID), which is derived from the file's content. This mechanism allows recipients to access and verify the authenticity of files using the CID. Compared to storing large HBIM model files directly on the blockchain, a CID, as a mere 256-bit length string, can be easily distributed on the blockchain, thus facilitating data integrity verification and access control without consuming significant network resources. By integrating blockchain technology with the IPFS, HBIM models and architectural restoration–related files can be stored in a distributed manner, addressing the challenge of storing large-scale data files directly on the blockchain. This integrated approach not only optimizes the efficiency and reliability of data storage but also enhances data security and accessibility. Utilizing the IPFS, large HBIM model files are broken down into smaller blocks and stored across multiple nodes, while the blockchain records the CIDs of these files, ensuring data immutability and permanence.

Using blockchain and IPFS integration can effectively solve the data security issues in the traditional HBIM model storage and central design structure, as well the data storage challenges of combining HBIM and blockchain technologies. This novel blockchain-IPFS integration solution provides an efficient, secure, and decentralized approach for the storage, distribution, and verification of Historical Building Information Models (HBIM). Through this method, architectural heritage restoration projects can more effectively manage and share critical historical architectural data, while ensuring data security and integrity. To this end, this paper proposes the Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) framework, aimed at solving the problem of storing large files in blockchain-based secure collaborative design applications using HBIM. The three research objectives of this study are summarized as:

1. Develop an Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) framework that integrates the blockchain and IPFS.
2. Develop technical components that support the functionality of the AHR-DCDE.
3. Evaluate the performance of the AHR-DCDE.

The rest of the paper is organized as follows: Section 2 describes the technology development process through a literature review and highlights the research gaps. Section 3 details the development process of the AHR-DCDE framework. Section 4 validates the feasibility of the AHR-DCDE framework through a case study of Pinghe Packaging Plant. Section 5 discusses the innovativeness, potential impacts, and limitations of the AHR-DCDE framework. Finally, Section 6 presents conclusions and perspectives for future work.

2. Literature Review

2.1. HBIM-Based Collaborative Workflow

The decision-making process for the conservation and restoration of architectural heritage depends on interdisciplinary data, such as historical, diagnostic, and documentary information [24]. The application of BIM technology to the field of historic buildings, known as HBIM, greatly aids in information management, data exchange, and enhancing data accessibility [25]. In 2009, researchers such as Murphy first introduced the concept of "HBIM", aimed at clearly positioning its application within historical contexts, defining HBIM as a parametric object library concerning historical building data [25]. As research on related technologies deepened, HBIM has gone beyond the scope of traditional modeling, extending to the entire lifecycle management of historic buildings. The interdisciplinary nature of this field has attracted the attention of an increasing number of scholars and experts who have begun to focus on the application of HBIM in collaborative workflows. Scholars such as Ziyi Zhang further developed the definition of HBIM, viewing it as "a collaborative method based on digital technology [26]". Meanwhile, Jordan-Palomar and others focused on the application of HBIM in management, particularly in terms of multidisciplinary participation, monitoring, and forecasting, as well as information transmission, collection, and data standardization [27]. Nieto-Julian and colleagues developed a set of collaborative processes for identifying and classifying architectural components in architectural heritage based on cultural heritage restoration and conservation projects [28]. These studies not only enhanced the work efficiency of HBIM collaborative processes but also promoted ongoing collaborative communication. However, despite these collaborative methods significantly improving efficiency, issues such as complex workflows, poor model version management, and data redundancy and rework still persist in practice. This indicates that although HBIM provides significant support for architectural heritage restoration, the collaborative workflow still needs further optimization to improve the efficiency and outcomes of projects.

2.2. *Common Data Environment for Architectural Heritage Restoration*

To enhance the efficiency of BIM-based collaboration, the ISO 19650 [9] standard recommends the use of a Common Data Environment (CDE) for BIM project management. The CDE provides a structured framework for project management and collaboration, particularly important in the application to architectural heritage restoration projects. As shown in Figure 1, the Common Data Environment for architectural heritage restoration is composed of four main information containers: Work in Progress (WIP), Shared, Publish, and Archive. In the Work in Progress (WIP) container, the heritage restoration team uses specific domain software to develop models before and after restoration. At this stage, the data is in development and can only be accessed and edited by the internal project team. Subsequently, in the Shared stage, data verified through multidisciplinary collaboration by the team becomes shareable. At this point, data can be accessed but not edited, to ensure the accuracy of the information transmitted. The data in the Publish container is authorized by the client and ready for construction use, prepared to be released to project stakeholders. Finally, the Archive container holds a log of all published and shared information throughout the project, providing a complete historical record for the project. In the Common Data Environment (CDE), files circulated within each container must undergo review and confirmation by the client or project leader to ensure the accuracy and reliability of data. The CDE also specifies the definition of data states, rules for file naming, and regulations for version control, which are used to achieve efficient collaborative design [29].

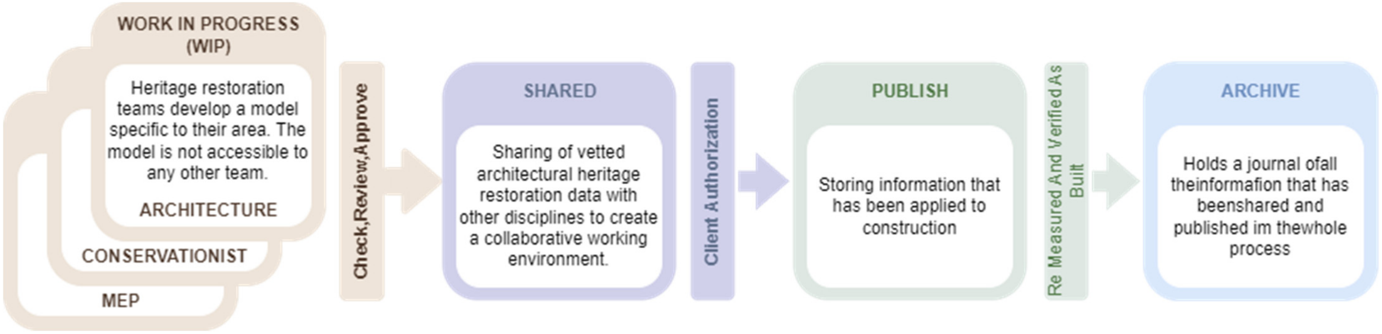


Figure 1. Common Data Environment (CDE) workflow.

The Common Data Environment (CDE) for architectural heritage restoration can be realized through various toolsets, including cloud servers, local databases, and file-based retrieval systems [30]. This flexibility permits project teams to choose the most appropriate technological platform to support their workflows, tailored to their specific needs and conditions. Researchers such as Cornelius Preidel [30] developed a cloud-based CDE repository, named the BIM Integration Framework (BIF), to enhance the convenience for project members in accessing and sharing BIM models. The development of this framework allowed project members to access and update BIM models in real-time, thus improving work efficiency. Stefan Mordue [31] proposed a CDE framework for HVAC design and engineering that, by optimizing the inspection, version control, and publication processes of information, reduced the time wasted in reissuing information, thereby enhancing the collaborative capabilities of the project team. Commercial Common Data Environment (CDE) solutions, such as BIM360 [13] and BIM cloud [32], facilitate the consolidation of models within a unified, cloud-based setting. This enables design teams to collaborate using the most current models, thereby demonstrating the practical benefits of CDE technology in the management of construction projects.

Although the introduction of the Common Data Environment (CDE) has played a key role in improving the efficiency of BIM-based construction project collaboration, its application in the field of architectural heritage restoration design has encountered specific challenges. Due to the uniqueness of architectural heritage restoration design, the CDE models developed for standard BIM projects do not fully meet its requirements. Furthermore, most existing CDE models are based on cloud servers or centralized databases, which to some extent increases network security risks, such as tampering with design data or the denial of access to data. Researchers such as Parn [33] have analyzed the network security threats facing the CDE, highlighting the importance of data throughout the building project lifecycle and the impact of data security on project success. Although measures such as firewalls or access controls have been implemented to reduce external threats, internal project personnel might exploit their access rights to the CDE database for data tampering, thus revealing potential network security vulnerabilities associated with relying on a single data source. Therefore, addressing the network security vulnerabilities associated with the use of a single data source in secure collaborative design for HBIM-based architectural heritage restoration projects has become an urgent issue. This requires the development of new CDE models or the enhancement of the security of existing models to ensure the integrity of data and the security of project information. This could include the introduction of more advanced data encryption technologies; the implementation of stricter access control mechanisms; the adoption of distributed data storage solutions to reduce the risks of centralized storage; or the exploration of emerging technologies, such as the blockchain, to provide immutable data records and enhanced security.

2.3. Blockchain in the Construction Industry

Blockchain technology, as a decentralized and distributed ledger technology (DLT), has been widely adopted in the Architecture, Engineering, and Construction (AEC) industry due to its traceability, immutability, transparency, and accountability. These characteristics help enhance transparency, improve information sharing, and solve trust issues [34]. For instance, Wang [20] and colleagues developed a blockchain-based framework for managing information in the prefabrication supply chain, which enhanced the traceability of the supply chain; Chin-Lin HSU [35] and others proposed a blockchain-based parametric machine learning approach that facilitated knowledge sharing in BIM collaboration; and Tao [36] and colleagues explored a blockchain-based confidentiality framework aimed at promoting design coordination and information confidentiality. Although there has been some research on the application of the blockchain in BIM design collaboration, its application in the collaborative design of Historical Building Information Modeling (HBIM) has been relatively scarce. This is mainly because the file size of HBIM is typically much larger than that of traditional BIM files, and the inherent limitation of the blockchain being suited for storing smaller files makes it unsuitable for directly storing large files. Therefore, methods for utilizing blockchain technology to protect HBIM files (before and after restoration) and project process files have not yet been widely developed in the construction industry.

The combination of blockchain technology with the InterPlanetary File System (IPFS) offers an innovative solution to the issue of the blockchain's inability to directly store large files. The IPFS is a distributed, peer-to-peer file system that allows for the storage of large files without relying on centralized servers. By storing these large files on an IPFS and uploading the resulting Content Identifiers (CIDs) to the blockchain network, it is possible not only to achieve storage, traceability, and authenticity verification of large datasets but also to ensure the security and integrity of the data. This method has been applied in fields such as medical data management and video storage. For example, Sun [37] and others combined blockchain and IPFS technologies to propose an encryption scheme that effectively achieves secure storage and querying of medical data, demonstrating the strong potential of this combined method in handling sensitive data. Yu [38] and colleagues introduced a digital rights management system based on blockchain and IPFS technologies, addressing the issue that the blockchain cannot directly store large files such as videos and audio. In the Architecture, Engineering and Construction (AEC) industry, this convergence of technologies also shows great promise for application. Tao [39] et al. propose a new solution for BIM co-design by combining blockchain and IPFS technologies. This innovation not only provides an effective solution path for large file storage, but also brings new possibilities for project management and team collaboration by ensuring data security and integrity.

Although the combination of blockchain and IPFS has shown great potential in other industries, such as AEC, application research in the field of architectural heritage restoration is relatively scarce, especially regarding how this integrated method can be applied to the Common Data Environment (CDE) for architectural heritage restoration, which remains a challenge. Therefore, this paper proposes an Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) framework, aiming to address the aforementioned issues.

3. The Development of the Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) Framework

The development of the Architectural Heritage Rehabilitation Distributed Common Data Environment (AHR-DCDE) faces three key challenges: (1) it needs to be clarified how to integrate the Architectural Heritage Rehabilitation CDE with blockchain and IPFS technologies in order to build a logical framework that can support a secure collaborative design process; (2) in order to support the AHR-DCDE functionality, a CDE-compliant transaction data model must be developed; (3) a corresponding smart contract needs to be written to support the AHR-DCDE functionality. Figure 2 shows the methodology of the AHR-DCDE framework development.

In Step 1, this research needs to propose the AHR-DCDE framework (introduced in Section 3.1). A detailed explanation is needed in this stage of how the logical problem of file storage in the architectural heritage restoration process and how multidisciplinary professionals can collaborate in a distributed CDE through an integrated blockchain-IPFS approach, resulting in a workflow, can be solved. Additionally, it is necessary to explain how the AHR-DCDE framework can ensure data security in the architectural heritage restoration process through distributed storage technology. Step 1 focuses on outlining the concept of the AHR-DCDE framework, and the technical components supporting the AHR-DCDE framework (transaction data models and smart contracts) will be detailed in Steps 2 and 3.

In Step 2, this study requires the development of transaction data models for recording the architectural heritage restoration process (as described in Section 3.2). The blockchain transaction data model mainly includes two requirements. First, it is necessary to consider the data model requirements of the blockchain platform. Second, it is necessary to meet the data model requirements for architectural heritage restoration collaborative design in the CDE standards. Transaction data models are developed based on these requirements to record architectural heritage restoration data within the AHR-DCDE framework.

In Step 3, this study requires the development of smart contracts for recording the sharing of architectural heritage restoration data files (as detailed in Section 3.3). The functionality of the smart contracts is determined based on the activities in the architectural heritage restoration process within the AHR-DCDE framework. The development of smart contracts aims to support the secure collaborative design process, enabling the sharing and querying of architectural heritage restoration data.

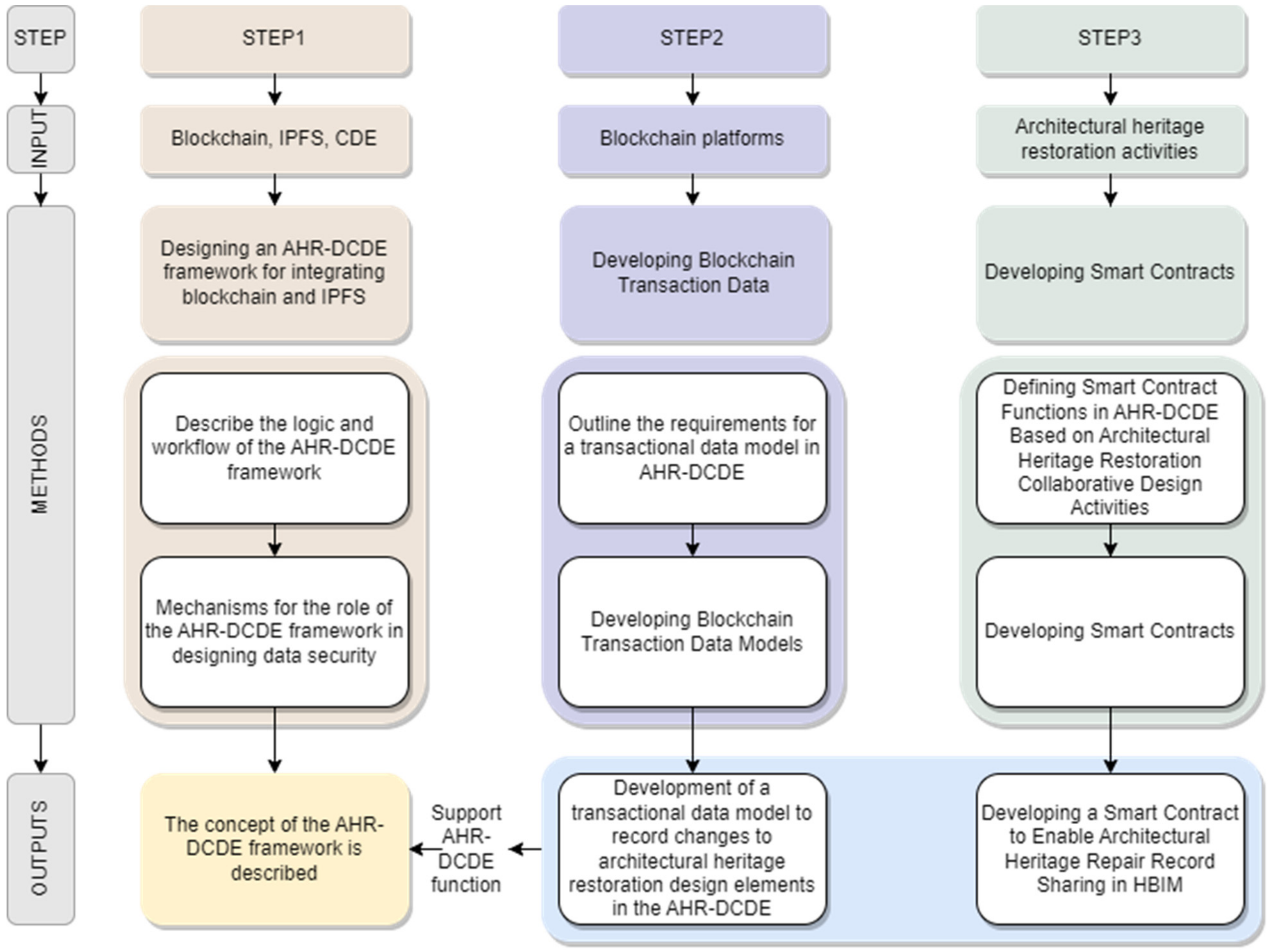


Figure 2. Methodology of AHR-DCDE framework development.

3.1. *Subsection Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) Framework*

During the development of the Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE), a series of steps was taken to establish an efficient and secure collaborative design process. This process, as depicted in Figure 3, begins with the uploading of pre-restoration related documents, involving key project team members such as architects, structural engineers, and conservators. Based on preliminary measurements and the data collected from documents, professional software tools are utilized to create HBIM models, structural analysis reports, and architectural assessment reports. Once approved by the team leader, these documents are uploaded to the IPFS network by the project members to facilitate further collaboration. Subsequently, IPFS automatically generates a unique Content Identifier (CID) for the uploaded files, making them indexable and retrievable within the network. Project members then use smart contracts to submit a special transaction to the blockchain network that includes key collaboration information, such as the file's CID, version number, and ownership information. This step ensures the security and reliability of the file-sharing process because the immutability of the blockchain guarantees the authenticity of the data. Project team members, such as historic building restorers and structural engineers, can download the materials they need from the IPFS network using the provided CID. Once they download a file, they become a secondary provider of that file, increasing its availability within the network. This demonstrates the advantages of decentralized storage, ensuring that project members can easily access data regardless of which node it is stored on in the network. When project members, such as structural engineers or clients, need to make suggestions for structural reinforcement or design modifications, they can download the required files from any provider using the same process. After making suggestions, they follow the steps previously described to upload feedback to the IPFS and record the transaction on the blockchain. This cyclical process, linked through CIDs in the IPFS and blockchain network, creates a distributed "shared" container that supports ongoing project collaboration. Upon completion of the design phase and final approval from the client, the status of the design documents is updated from "Shared" to "Publish", indicating that the files have been officially confirmed and are ready for use in the next phase of the project. This not only reflects an integrated collaborative design process but also shows how blockchain and IPFS technologies together provide a secure and reliable management model for architectural heritage restoration projects.

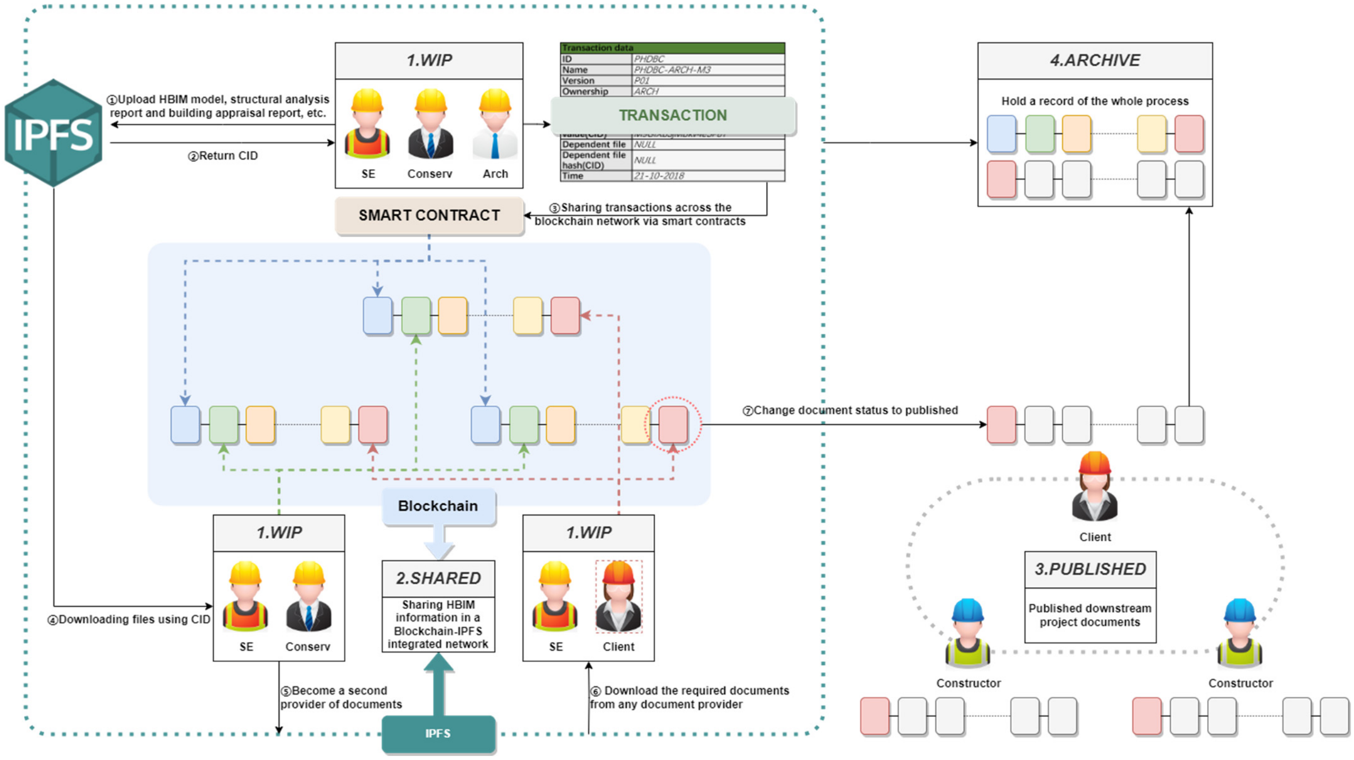


Figure 3. Architectural Heritage Restoration Distributed Common Data Environment (AHR-DCDE) framework.

Figure 4 displays the architecture of the AHR-DCDE, which consists of five main layers: User Layer, Work in Progress (WIP) Layer, Smart Contract Layer, Blockchain Network Layer, and Database Layer. In the User and WIP layers, project members utilize their local working environment to build various documents for the project, including pre- and post-restoration HBIM models, architectural inspection reports, and structural inspection reports. These documents support the design collaboration for the project, providing a platform for team members to update and share information in real time. The Smart Contract Layer provides functionalities for smart contracts and mechanisms for consensus, enabling participants in the project to execute various functions for uploading or retrieving transaction data. The introduction of consensus mechanisms ensures that all members receive transaction information in the same order, which is crucial for maintaining data consistency and accuracy. The Network Layer provides project members with the ability to register on the blockchain and IPFS platforms, serving as a key component of the architecture by connecting all project participants and providing them with access and a channel to share information. The Database Layer forms the core of the AHR-DCDE architecture, equipping each project member with a blockchain ledger and an IPFS database. At this level, the blockchain is responsible for recording transaction information, ensuring data immutability, while the IPFS database is used for storing and sharing large design files such as HBIM models and reports. Through this setup, the Database Layer becomes the centralized management point for project data, supporting secure sharing and persistent storage of data, ensuring that all members can efficiently access the information they need.

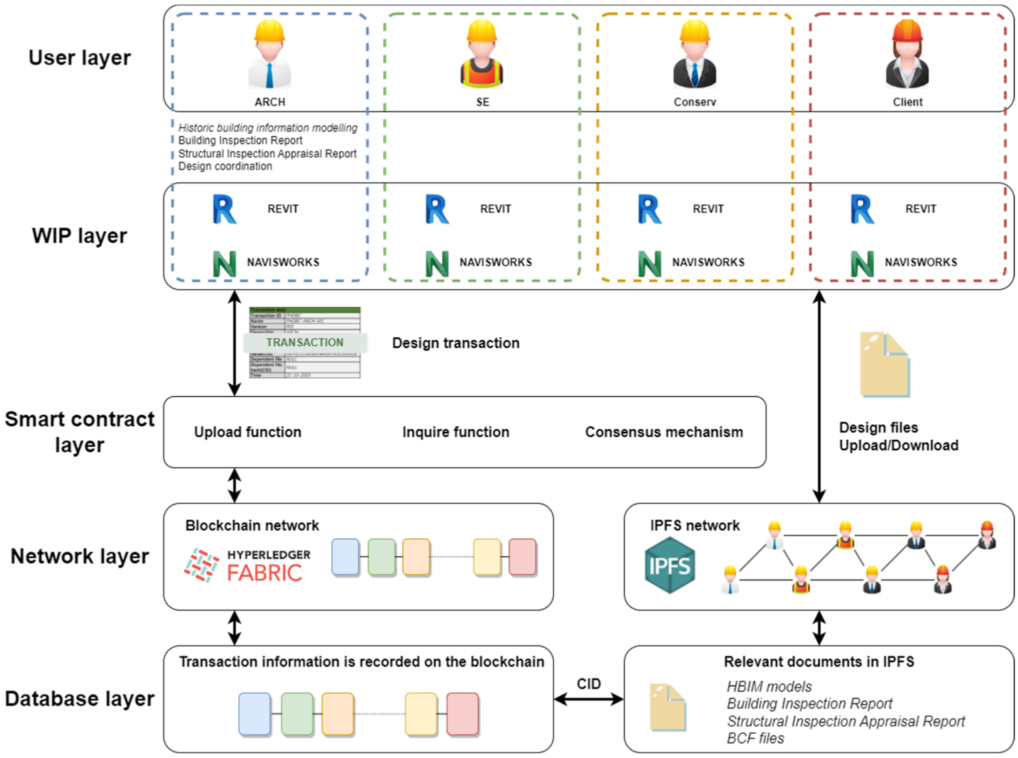


Figure 4. Architecture of AHR-DCDE framework.

Figure 5 illustrates the application of the AHR-DCDE architecture in addressing data security issues in collaborative design for architectural heritage restoration based on HBIM. Traditional approaches present data security risks in two main aspects: First, there are security issues with data changes in traditional HBIM architectural heritage restoration designs. Due to the complexity of the design process, traditional methods fail to comprehensively record every design change, leading to unclear responsibility attribution when designs that do not meet requirements emerge. Second, there are issues with data access denial. Traditional storage methods centralize data on a single server, which, if subjected to a cyber-attack, could result in data being altered or lost.

In contrast, the AHR-DCDE architecture effectively resolves these issues through the use of blockchain technology. Regarding the security issues associated with data alterations, all records of architectural heritage restoration designs that utilize Heritage Building Information Modeling (HBIM), referred to as transactions, are securely stored on the blockchain ledger. This ledger is both unchangeable and trackable, ensuring the integrity of the data. Furthermore, architectural heritage restoration data files encrypted with Content Identifiers (CIDs) generate a different CID, with every modification to the file content enhancing data security. Regarding the issue of data access denial, in the blockchain network, each node retains a complete set of architectural heritage restoration design records. Every node is equally important, eliminating the reliance on a single data storage point and thereby enhancing data security.

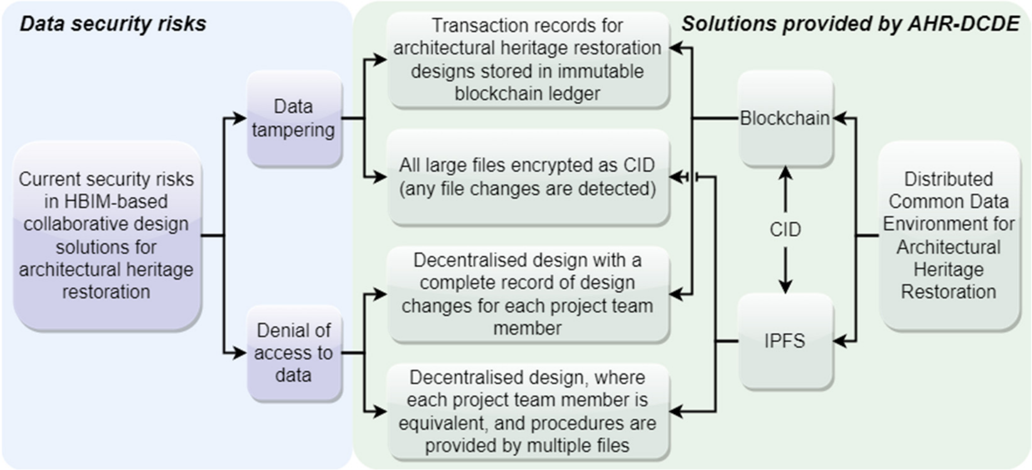


Figure 5. Proposed AHR-DCDE solution, ending data security issues.

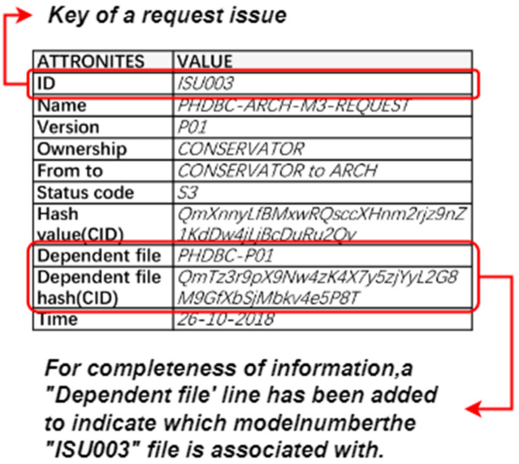
3.2. Blockchain Transaction Data Model Development

Blockchain transactions serve as the medium for sharing and recording necessary information on the ledger within the blockchain network. For the implementation of the AHR-DCDE architecture, developing a transaction data model suitable for architectural heritage restoration data exchange is a primary task. This study proposes a preliminary transaction data model, specifically designed for design collaboration based on HBIM, constructed around three core principles: compatibility, integrity, and interoperability.

To fulfill compatibility criteria, the transaction data model needs to be congruent with the selected blockchain infrastructure. For this investigation, the private blockchain network Hyperledger Fabric was selected as the foundational platform, primarily for its strengths in privacy safeguards, modular design, scalability, compatibility with various programming languages, and established practicality within industrial settings. The private nature and authorization mechanisms of Hyperledger Fabric are particularly suited for handling sensitive architectural heritage restoration data, ensuring data security and privacy [40]. Its modular design allows for customization according to project needs, meeting the specific requirements of the construction industry in collaborative design and architectural heritage restoration [41]. The support for multiple programming languages (Go, Java, and Node.js) simplifies the development process [42], making Hyperledger Fabric a flexible and user-friendly blockchain solution [43,44]. Given that Hyperledger Fabric primarily supports transactions in a key-value data model, we have developed a transaction processing method in a key-value format. For example, when a historic building restorer identifies a building component of special historical value that needs to be preserved, they can submit a problem request report to the AHR-DCDE. Figure 6 shows the information of the problem request report, with the key (file ID) being ISU003 and the value (attributes) including name, version, CID, etc.

To ensure integrity, the design of the transaction data model must guarantee that all necessary information is provided to project members throughout the architectural heritage restoration design process. For instance, a report on a specific historical building component that needs preservation should be closely linked with its related BIM model versions and supplementary materials, as shown in Figure 6. The concept of “dependent files” introduced here aims to clearly record the specific models and version information that the target report refers to. This method significantly improves the accuracy and efficiency of the collaborative design process.

To satisfy interoperability, the transaction data within the CDE utilizes formats that are compatible and easy to understand and exchange, adhering to the ISO 19650 [9,10] standard for data version management and status definitions. Table 1 ensures data consistency and interoperability by clearly defining the attributes in the transaction data model and their corresponding values, ensuring that data are effectively managed and communicated within the CDE workflow.



Key of a request issue

ATTRONITES	VALUE
ID	ISU003
Name	PHDBC-ARCH-M3-REQUEST
Version	P01
Ownership	CONSERVATOR
From to	CONSERVATOR to ARCH
Status code	S3
Hash value(CID)	QmXnnyLfBMxwRQsccXHnm2rjz9nZ1KdDw4iLjBcDuRu2Qv
Dependent file	PHDBC-P01
Dependent file hash(CID)	QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T
Time	26-10-2018

For completeness of information,a "Dependent file' line has been added to indicate which modelnumberthe "ISU003" file is associated with.

Figure 6. Illustration of the developed blockchain transaction data model.

Table 1. Explanation of transaction data model.

Attributes	Value
ID	The file ID functions as both the transaction key and a unique identifier for files, allowing project participants to access the current status of files within the blockchain network in real time. It is consistently used and remains unchanged throughout the collaborative design process of architectural heritage restoration. Unlike the Content Identifier (CID), which updates with changes to the file content, the file ID remains fixed, ensuring the consistency of file identity and the reliability of tracking.
Name	The ISO 19650 [9,10] standard provides conventions and optional fields for file naming, allowing project teams to construct file names based on project requirements, such as project name, originator, file type, etc. For example, in Figure 6, "PHDBC-ARCH-M3" represents a 3D (M3) model file in the architectural domain of the Pinghe Packing Factory Conservation project. The file "PHDBC-ARCH-M3-REQUEST" indicates that this is a request document regarding an architectural component issue.
Version	The ISO 19650 [9,10] standard establishes methods for data version control within a Common Data Environment (CDE). Modifications made to data within the Work in Progress (WIP) container are marked by minor version increments, for instance, transitioning from P01 to P01.1. Subsequent updates that move data to the Shared (SHARED) container necessitate an adjustment in the revision code to signify a major revision, such as altering from P01 to P02.
Ownership	Ownership of information in a CDE is defined in the ISO 19650 [9,10] standard as belonging to the creator of that information.
From to	Senders and recipients of documents.
Status code	Status codes are defined in the ISO 19650 [9,10] standard to describe the metadata for the content applicability of information containers. Several codes that will be used in this study are listed: in SHARED containers. S1: Suitability for coordination. The document can be shared with other disciplines. S3: Suitable for internal review and comment. The document is used to relay questions and requests. S4: Suitable for construction. The document is suitable for seeking client approval to proceed with construction in a PUBLISH container. A: Suitable for construction. The document is suitable for construction.
Hash value (CID)	CID returned by the IPFS network
Dependent file	Name of subsidiary document
Dependent file hash (CID)	CID of the attached file returned by the IPFS network
Time	Trading time

3.3. Smart Contract Development

Smart contracts, as computational programs capable of automatically executing the terms of agreements or contracts through software code and computational infrastructure, enable transactions to be carried out automatically without the need for intermediaries and to interact with a ledger or distributed ledger. Based on predetermined rules and conditions within the contract, smart contracts automatically trigger various operations, such as data updates or the broadcasting of new transactions. In this study, the development of smart contracts requires a clear understanding of the key collaborative design tasks in architectural heritage restoration under the AHR-DCDE architecture in order to determine the functionalities that need to be defined in the contract. This approach ensures that the smart contracts facilitate efficient and secure collaboration among project participants, aligning with the specific requirements and workflows of architectural heritage restoration projects.

Table 2 lists five main activities within the AHR-DCDE, four of which (numbered 1 to 4) require sharing the Content Identifier (CID) of files in the blockchain network. Therefore, the design of smart contracts includes an "UPLOAD" function to facilitate the smooth broadcasting of transactions. For the fifth activity, which requires querying existing transactions in the blockchain, the smart contract also includes an "INQUIRE" function for convenient information retrieval. This setup ensures that all necessary interactions with the blockchain, whether uploading new data or retrieving existing information, are efficiently managed through the smart contract's functionalities, thereby supporting the seamless execution of architectural heritage restoration projects within the AHR-DCDE framework.

Table 2. Based on the smart contract function for identifying architectural heritage restoration activities in the AHR-DCDE.

Number	Collaborative Design Activities in the AHR-DCDE	Explanation	Function in Smart Contract
1	Pre-restoration HBIM modeling and related file sharing	The sharing of Content Identifiers (CIDs) for HBIM models, structural inspection reports and building appraisal reports prior to the restoration of a building's heritage in a blockchain network enables project team members to access these key documents in order to carry out the design of structural reinforcement or to identify building components to be protected.	UPLOAD
2	HBIM model version update	Disseminating transactions that include the Content Identifier (CID) of an amended or updated Heritage Building Information Modeling (HBIM) model through the blockchain network grants every participant access to the most recent version of the model.	UPLOAD
3	Issue file sharing	Issue files cover BIM Collaboration Format (BCF) change files, Request for Information (RFI) files, and new design requirements files, among others, and the transactions contained in the issue file Content Identifiers (CIDs) are shared on the blockchain network. This approach allows other project members to have a clear picture of what changes were made, where they were made, and what new requirements were proposed.	UPLOAD
4	Sharing of contractual documents	When collaborating in a CDE, an updated contract needs to be shared between project members. The CID containing the contract will be shared across the blockchain network so that members are aware of the new contract terms.	UPLOAD
5	Information enquiry	All transactions can be queried in the blockchain network, which is used to prevent disputes from occurring.	UPLOAD

Figure 7 illustrates how the developed smart contracts facilitate the secure storage and convenient querying of architectural restoration transactions. When project members, such as architectural designers, wish to share the CID of a pre-restoration HBIM model of architectural heritage, they utilize the "UPLOAD" function of the smart contract. The process is divided into the following six steps:

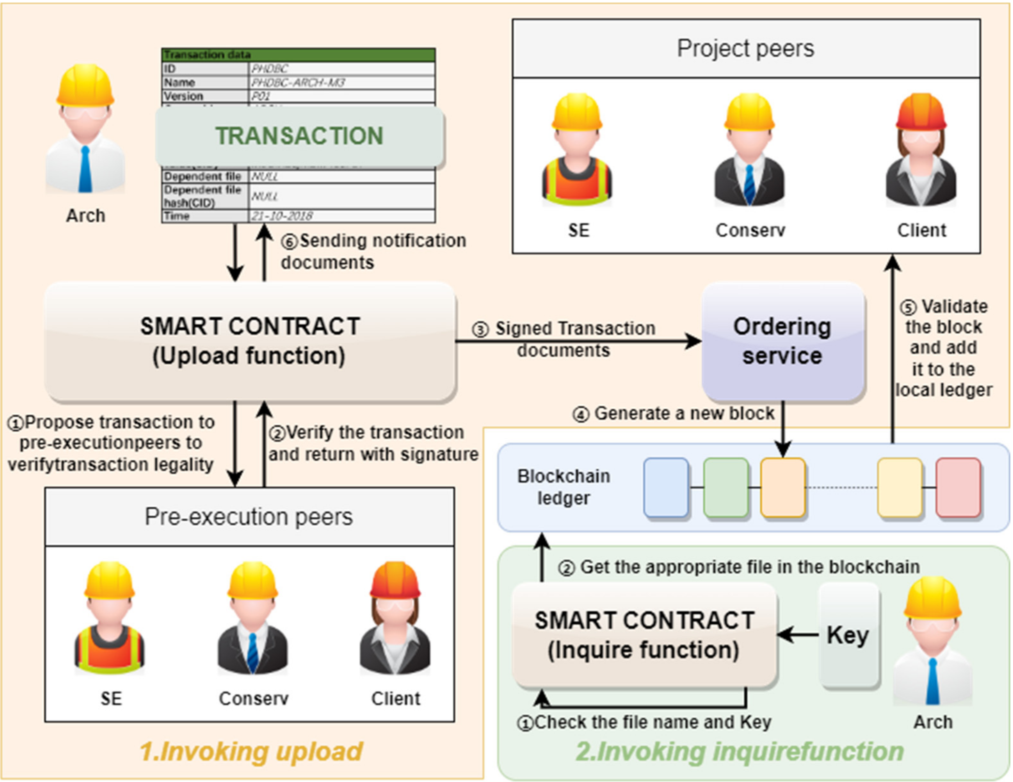


Figure 7. Implementing a shared and query transaction smart contract workflow in AHR-DCDE.

1. Upload Request: Project members initiate an upload request to share a CID. The smart contract receives and processes this request.
2. Pre-execution Verification: In the first step, the smart contract transmits the transaction to pre-execution members who review the transaction to verify its legitimacy. Transactions containing illegal parameters will be rejected.
3. Signing and Confirmation: Once verified, project members sign the transaction and return it to the smart contract.
4. Ordering and Broadcasting: The smart contract sends the signed transaction to the ordering service. The ordering service collects multiple transactions over a period, forms a new block, and broadcasts it to the blockchain network.
5. Validation and Recording: Other project members validate the block in the broadcast and record it in their local ledger, ensuring the transaction's legality and consistency.
6. Transaction Success Notification: The smart contract sends a notification to the transaction initiator, confirming the successful sharing and storage of the transaction.

When project members need to query a file on the blockchain, they simply call the query function, inputting the file name and its KEY. The smart contract independently confirms whether the input key is present in the blockchain ledger; if the key is found, it then retrieves and returns the results of the query. In summary, the smart contracts developed in this paper possess the functionalities required for the architectural heritage restoration field. These include uploading architectural heritage BIM files, structural analysis report files, and architectural historical materials, feedback on issue records, document sharing, and information querying. Participants in the project can partake in the design activities for architectural heritage restoration by activating various functions of the smart contract, thereby enabling a thorough and streamlined collaborative effort.

4. AHR-DCDE Applied Research

To verify the feasibility of the AHR-DCDE in collaborative design for architectural heritage restoration, this study selected the Pinghe Packing Factory heritage restoration design project located in Wuhan, Hubei Province, as a case study. The Pinghe Packing Factory, situated at No. 10 Qingdao Road in Wuhan, spans an area of 32,808 square meters (see Figure 8). It was constructed in 1905 by the Han Xie Sheng Construction Factory, marking it as one of the early significant industrial buildings. In 1993, the building was recognized as an excellent historical building in Wuhan, elevated to a Wuhan cultural relics protection unit in 2011, and further classified as Grade I industrial heritage in 2012. The Pinghe Packing Factory not only has a special place in history but, because of its long history, it has been repeatedly restored throughout its history. The huge amount of restoration data relating to the building has been lost or changed due to data storage, resulting in the need for re-measurement and revision of many parts of its restoration process which could not fully rely on existing data. This further highlights the importance of data security in the restoration process. Considering its age and the complexities involved in the design and collaboration process, as well as the need for long-term data security, the project decided to adopt the AHR-DCDE methodology for architectural heritage restoration co-design in order to address the collaborative difficulties and data security needs of the project.

Figure 8. Photos of the site after rehabilitation of the Pinghe Packing Factory.

4.1. Preliminary Preparation

To ensure the successful application of the AHR-DCDE architecture in the Pinghe Packing Factory project in Wuhan, the project team must carry out several key preparations: setting up the blockchain and IPFS networks, standardizing file formats, and becoming proficient in HBIM-related tools.

Firstly, by setting up the Hyperledger Fabric network and incorporating teams from architectural design, structural engineering, historic building restoration, and the client side, team members can upload documents through the IPFS desktop application, utilizing the generated Content Identifiers (CIDs) to achieve efficient file sharing. Secondly, the project adopts point cloud scanning and Revit 2020 software to develop HBIM models before and after restoration. By introducing the Industry Foundation Class (IFC) format and BIM Collaboration Format (BCF), file standardization is achieved. This not only facilitates information exchange between different software applications but also allows for precise issue localization through detailed problem descriptions in BCF files, including type, location, photos, and more. Lastly, project members need to become proficient in HBIM-related software. In this project, architects use Recap software to process radar scan data and employ BIM360 to import into Revit, constructing the pre-restoration HBIM model. Through BIM collab Zoom, team members can view the model and effectively communicate and resolve issues using the BCF issue functionality and Revit's BCF plugin.

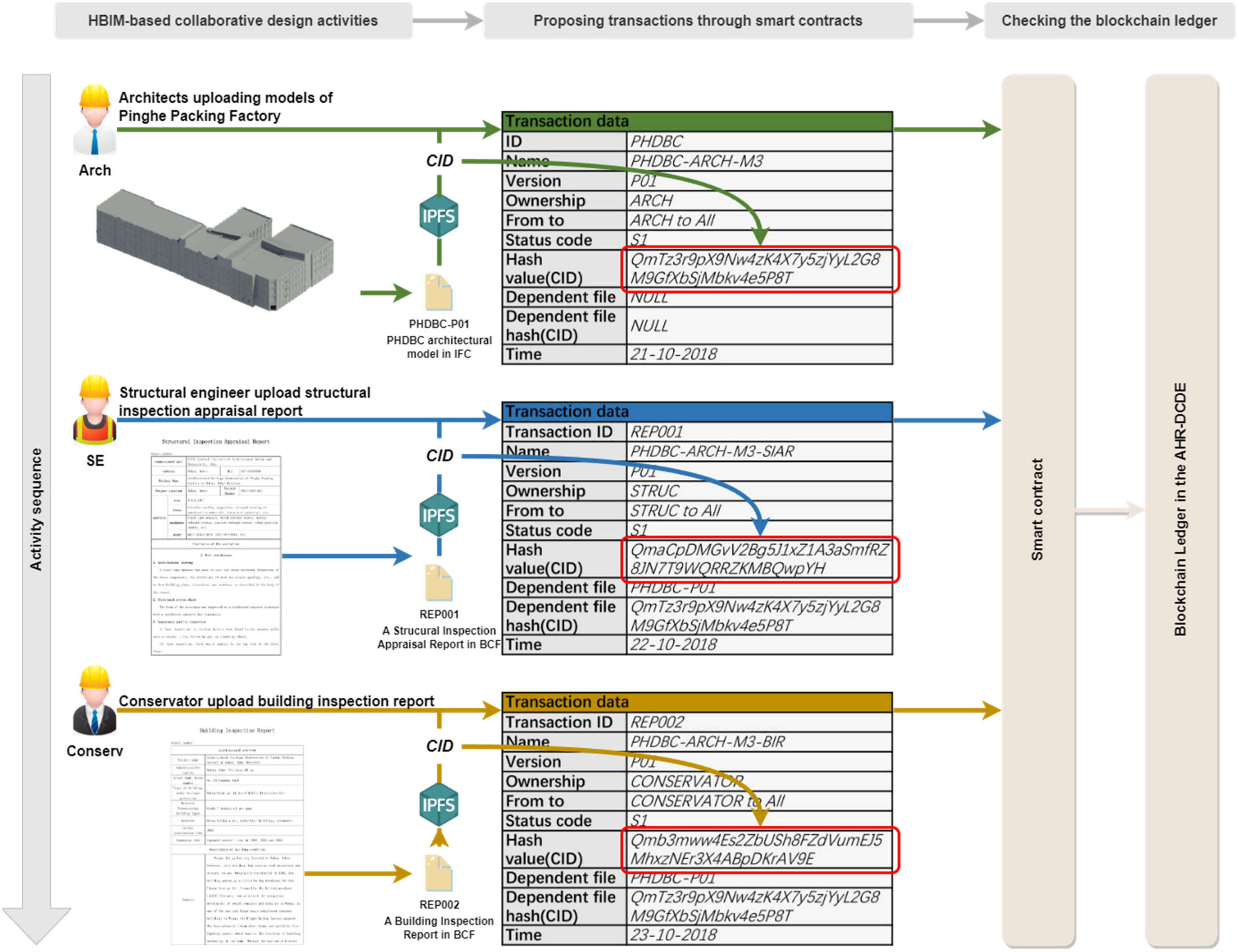
4.2. Project Applications

In this project, the application of the AHR-DCDE architecture is divided into three stages. In the initial stage, project team members shared existing architectural models, structural analysis reports, and historical materials by submitting transactions containing the Content Identifiers (CIDs) of these files to the blockchain network. Subsequently, the team entered a collaborative design stage based on HBIM, where they meticulously planned unchangeable architectural components and locations requiring reinforcement and repair by proposing and responding to BCF issues, in combination with historical documents and HBIM models. This allowed for precise adjustments and optimization of the design. Finally, after several rounds of iteration and approval, the client submitted approval documents, and the architect subsequently updated the HBIM model status to "PUBLISHED", marking the successful conclusion of the collaborative design stage of the architectural heritage restoration project. This process not only demonstrated the effectiveness of the AHR-DCDE architecture in promoting team collaboration but also ensured the precise implementation and efficient management of the heritage restoration design.

4.2.1. Sharing Information to Begin Collaborative Design of Architectural Heritage

In this stage, two steps are involved. First, the architect submits a transaction request to the AHR-DCDE network via smart contracts. Then, the system automatically verifies the execution of the smart contract to ensure all project members receive the new transaction information. As shown in Figure 9, the architect and project members upload relevant pre-restoration files of the Pinghe Packing Factory to IPFS, obtain the CID, and prepare the transaction request, with the status code being S1, indicating that the file is suitable for sharing with other project members. Subsequently, this information is uploaded through smart contracts, and the newly generated block is passed to all project members, allowing them to verify the transaction in the blockchain ledger and download the relevant files via CID. This process effectively promotes secure file sharing and collaborative work within the project team.

The results in Figure 10 demonstrate that the block containing transaction information has been successfully passed to every member of the project. Project members are able to review the transaction data in the blockchain ledger and use the Content Identifier (CID) stored within the block to download the required files. This process confirms the effective connectivity and application potential of the proposed AHR-DCDE framework in real projects.



Transaction data	
ID	*PHDBC*
Name	*PHDBC-ARCH-M3*
Version	*P01*
Ownership	*ARCH*
From to	*ARCH to All*
Status code	*S1*
Hash value(CID)	*QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T*
Dependent file	*NULL*
Dependent file hash(CID)	*NULL*
Time	*21-10-2018*

Transaction data	
Transaction ID	*REP001*
Name	*PHDBC-ARCH-M3-SIAR*
Version	*P01*
Ownership	*STRUC*
From to	*STRUC to All*
Status code	*S1*
Hash value(CID)	*QmaCpDMGvV2Bg5J1xZ1A3aSmfRZ8JN7T9WQRRZKMBQwpYH*
Dependent file	*PHDBC-P01*
Dependent file hash(CID)	*QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T*
Time	*22-10-2018*

Transaction data	
Transaction ID	*REP002*
Name	*PHDBC-ARCH-M3-BIR*
Version	*P01*
Ownership	*CONSERVATOR*
From to	*CONSERVATOR to All*
Status code	*S1*
Hash value(CID)	*Qmb3mww4Es2ZbUSh8FZdVumEJ5MhxzNEr3X4ABpDKrAV9E*
Dependent file	*PHDBC-P01*
Dependent file hash(CID)	*QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T*
Time	*23-10-2018*

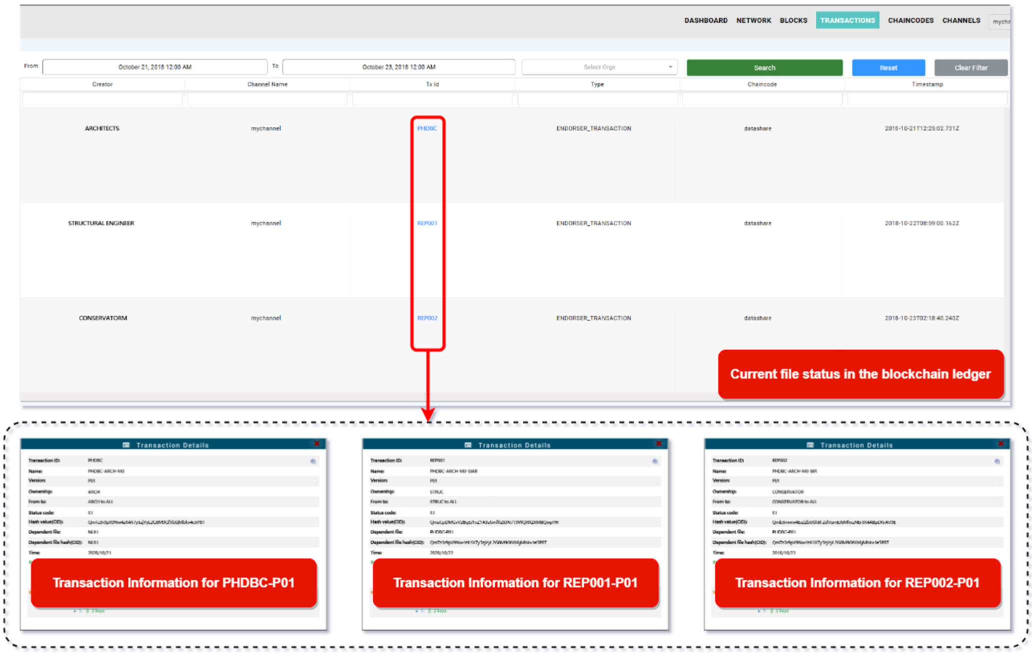
Figure 9. Stage 1 flowchart: sharing models and preliminary information to begin collaborative design.

4.2.2. HBIM-Based Collaborative Design for Architectural Heritage Restoration

The purpose of this stage is to facilitate information exchange, model revisions, and communication across disciplines, ensuring efficient collaboration among project team members. The flowchart presented in Figure 11 details the specific steps taken to achieve this objective within the Pinghe Packing Factory heritage restoration project.

Firstly, the structural engineer conducted a detailed analysis of the downloaded PHDBC-P01 model and the structural analysis report, identifying key nodes that required structural reinforcement. Necessary structural reinforcement designs were executed in the HBIM model, and a request for approval was submitted to the architect through an issue file (ISU001-P01). To share this information, the structural team created a new transaction, including the Content Identifier (CID) of the issue file and the related BIM model's CID, while also designating the issue file's status code as S3, indicating its purpose for internal review. The call to the smart contract ensured that all project members received this transaction notification and a corresponding block was generated in the blockchain.

Following this, the client suggested changes in line with the restoration design needs, such as altering the door types. They pinpointed the exact location and drafted a request issue in BCF format, thereby generating a new transaction that encompassed the CID of the issue file along with the associated model. This transaction was disseminated among all project participants through smart contracts, leading to the creation of a corresponding block on the blockchain.



Creator	Channel Name	Tx Id	Type	Chaincode	Timestamp
ARCHITECTS	mychannel	PHDBC	ENDORSER_TRANSACTION	datashare	2018-10-21T12:25:02.731Z
STRUCTURAL ENGINEER	mychannel	REP001	ENDORSER_TRANSACTION	datashare	2018-10-22T08:59:00.162Z
CONSERVATORM	mychannel	REP002	ENDORSER_TRANSACTION	datashare	2018-10-23T02:18:40.240Z

Figure 10. Transactions in stage 1 have been successfully shared.

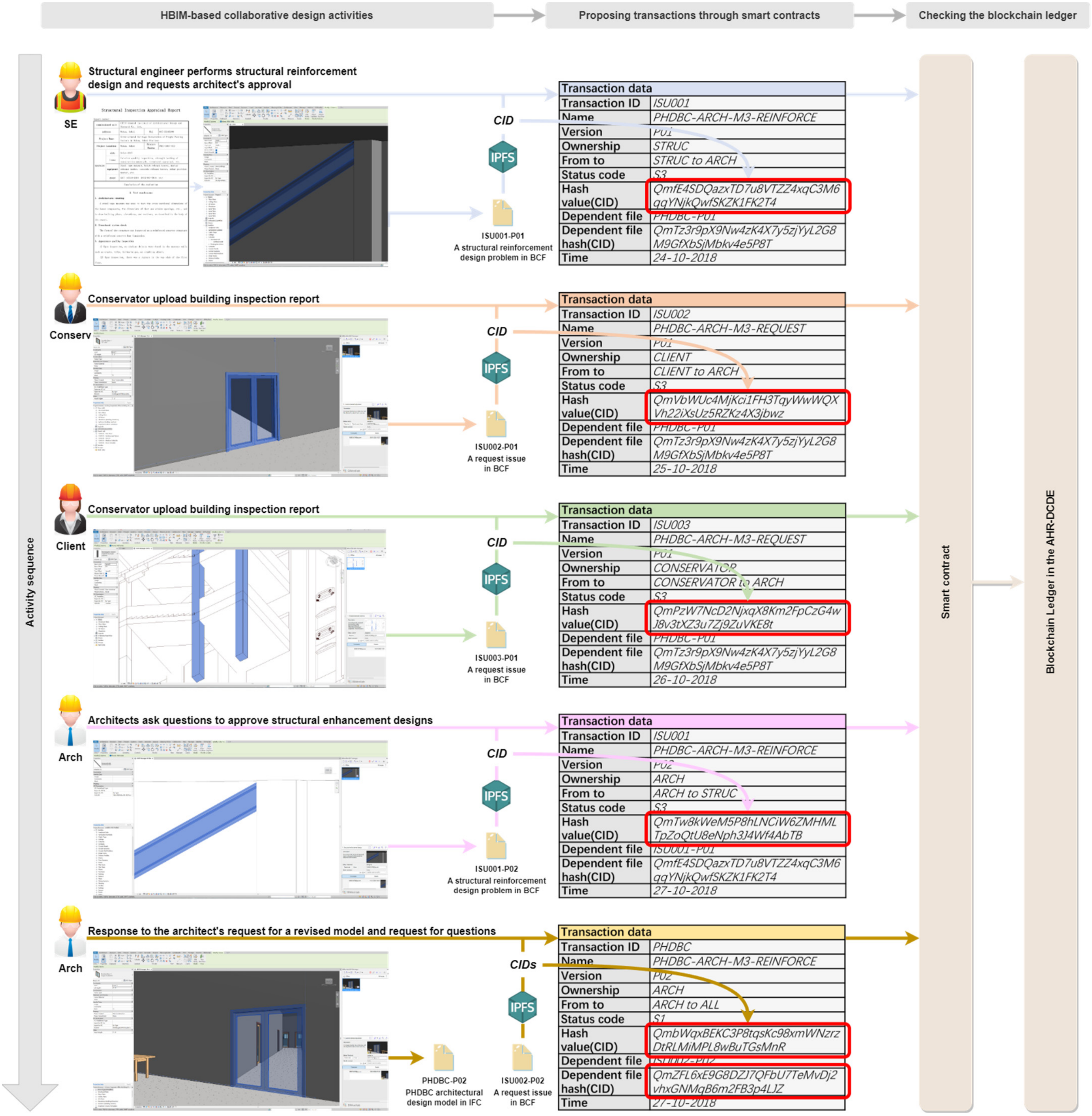
Similarly, based on the architectural appraisal report and professional analysis, the historic building restorer marked locations requiring special protection and prepared a request issue in BCF format. By creating a new transaction that included the CID of the issue file and the related model's CID and calling the smart contract to share this transaction, the historic building restorer ensured that all project members were promptly notified and the corresponding block was generated in the blockchain.

Upon obtaining insights from the structural engineer, the architect retrieved the BCF file (ISU001-P01) from the IPFS network and accessed it through the BCF Manager plugin within Revit. This tool helped the architect quickly and accurately locate the nodes that needed structural reinforcement. The architect responded to the issue file (ISU001-P01), and, after approval, a new version of the issue file with the architect's approval (ISU001-P02) was generated. The architect then created transaction information containing the updated issue file's CID and shared this transaction through the smart contract, ensuring real-time updates and information sharing among all members of the project.

When addressing the client's request for a change in door type, the architect updated the door type from Single-Flush0101 to Single-Flush0205 and responded to the issue file (ISU002-P01), indicating that the requirement had been met. A new transaction was then generated, containing the CID of the revised model (PHDBC-P02) and the updated request issue file (ISU002-P02). Through the call of the smart contract, this transaction was shared with all project members to notify them of the latest change in door type within the HBIM model.

In response to the requirements posed by conservator experts, the architect locked the components marked for special protection within the BIM model and confirmed that the requirements had been met through the issue file (ISU003-P01). This series of communication and update processes, efficiently implemented through the AHR-DCDE architecture, significantly improved the efficiency of project collaboration. It ensured rapid responses to design changes and strengthened communication and collaborative work among project team members.

Figure 12 shows that at this stage, all transactions have been successfully shared, meaning new blocks (or transactions) have been smoothly delivered to every project member. This allows the project team to review transaction information in their respective blockchain ledgers, thereby confirming that the AHR-DCDE framework can provide a secure and transparent communication channel and collaborative design environment for architectural heritage restoration projects. This mechanism not only enhances the efficiency and accuracy of project information management but also significantly improves the reliability and effectiveness of cross-disciplinary team collaboration.



Transaction data	
Transaction ID	ISU001
Name	PHDBC-ARCH-M3-REINFORCE
Version	P01
Ownership	STRUC
From to	STRUC to ARCH
Status code	S3
Hash value(CID)	QmfE4SDQazxTD7u8VTZZ4xqC3M6qqYNjkQwfSKZK1FK2T4
Dependent file	PHDBC-P01
Dependent file hash(CID)	QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T
Time	24-10-2018

Transaction data	
Transaction ID	ISU002
Name	PHDBC-ARCH-M3-REQUEST
Version	P01
Ownership	CLIENT
From to	CLIENT to ARCH
Status code	S3
Hash value(CID)	QmVbWUc4MjKci1FH3TqyWwWQXVh22iXsUz5RZKz4X3jbwz
Dependent file	PHDBC-P01
Dependent file hash(CID)	QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T
Time	25-10-2018

Transaction data	
Transaction ID	ISU003
Name	PHDBC-ARCH-M3-REQUEST
Version	P01
Ownership	CONSERVATOR
From to	CONSERVATOR to ARCH
Status code	S3
Hash value(CID)	QmPzW7NcD2NjxqX8Km2FpCzG4wJ8v3tXZ3u7Zj9ZuVKE8t
Dependent file	PHDBC-P01
Dependent file hash(CID)	QmTz3r9pX9Nw4zK4X7y5zjYyL2G8M9GfXbSjMbkv4e5P8T
Time	26-10-2018

Transaction data	
Transaction ID	ISU001
Name	PHDBC-ARCH-M3-REINFORCE
Version	P02
Ownership	ARCH
From to	ARCH to STRUC
Status code	S3
Hash value(CID)	QmTw8kWeM5P8hLNCiW6ZMHMLTpZoQtU8eNph3J4Wf4AbTB
Dependent file	ISU001-P01
Dependent file hash(CID)	QmfE4SDQazxTD7u8VTZZ4xqC3M6qqYNjkQwfSKZK1FK2T4
Time	27-10-2018

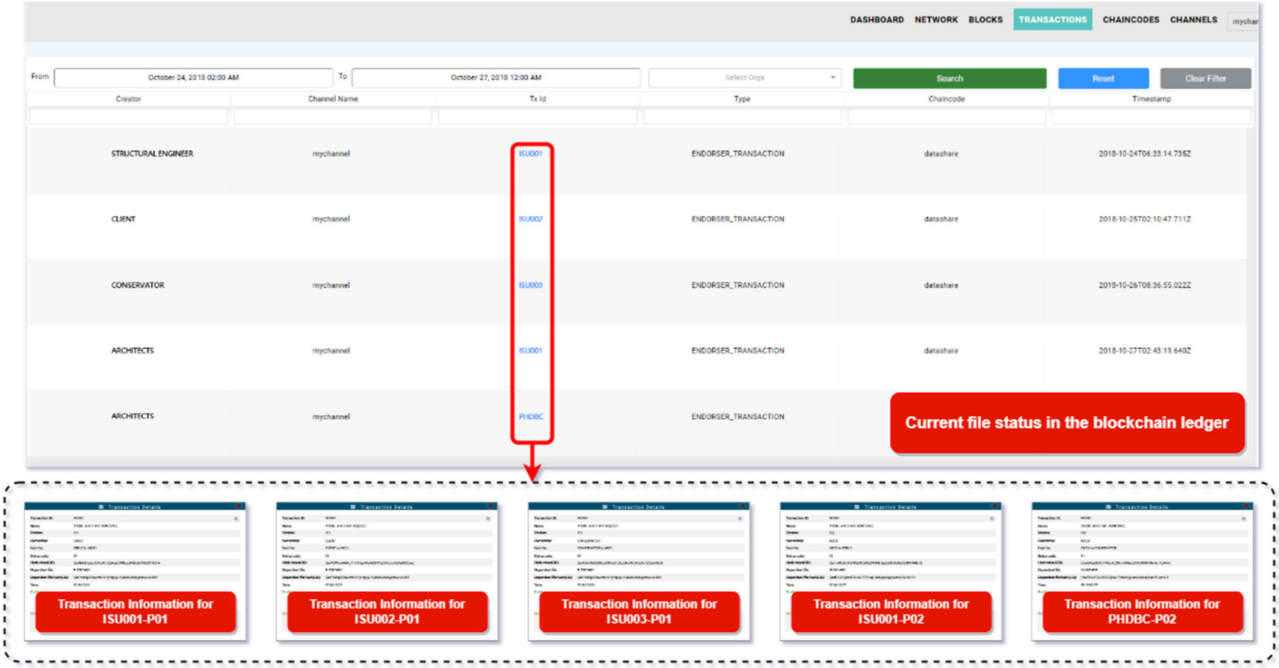
Transaction data	
Transaction ID	PHDBC
Name	PHDBC-ARCH-M3-REINFORCE
Version	P02
Ownership	ARCH
From to	ARCH to ALL
Status code	S1
Hash value(CID)	QmbWqxBEKC3P8tqsKc98xmWNzrzDtRLMiMPL8wBuTGsMnR
Dependent file	ISU002-P02
Dependent file hash(CID)	QmZFL6xE9G8DZJ7QFbU7TeMvDj2vhxGNMqB6m2FB3p4LJZ
Time	27-10-2018

Figure 11. Stage 2 flowchart: multi-team collaborative design based on HBIM.

4.2.3. Change HBIM Model Status from SHARED to PUBLISH

After the architectural heritage restoration design within the AHR-DCDE framework is completed, it needs to be officially published by migrating the design files from the

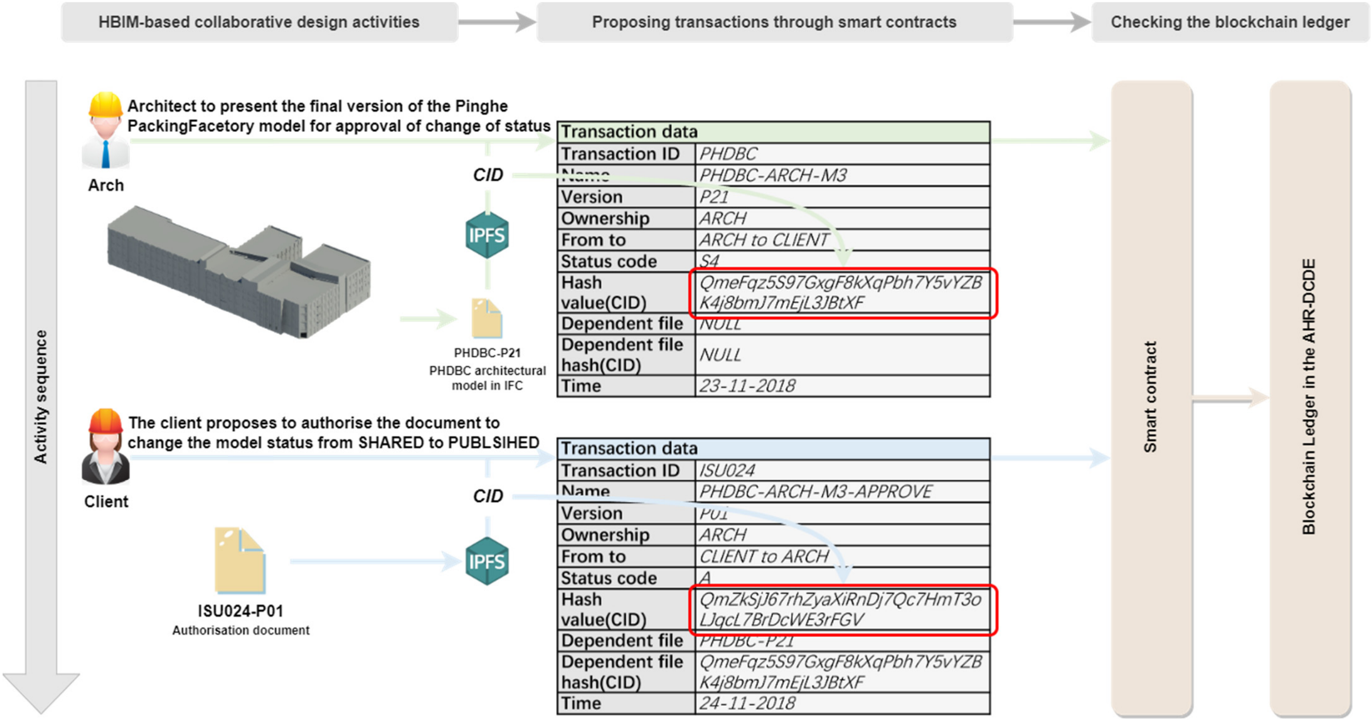
"SHARED" container to the "PUBLISH" container, and this requires authorization from the client. Figure 13 illustrates this process. After multiple rounds of design iterations, the restoration plan is ready for submission for review. At this time, the architect initiates a new transaction containing the CID of the final HBIM model, which is version P21. The status code S4 indicates that the file is ready for publication. Through smart contracts, the architect shares this transaction with the project team, notifying them that the PHDBC-P21 model is prepared to move into the next stage.



From October 24, 2018 02:00 AM To October 27, 2018 12:00 AM

Creator	Channel Name	Tx Id	Type	Chaincode	Timestamp
STRUCTURAL ENGINEER	mychannel	ISU001	ENDORSER_TRANSACTION	datashare	2018-10-24T06:33:14.735Z
CLIENT	mychannel	ISU002	ENDORSER_TRANSACTION	datashare	2018-10-25T02:10:47.711Z
CONSERVATOR	mychannel	ISU003	ENDORSER_TRANSACTION	datashare	2018-10-26T08:36:55.022Z
ARCHITECTS	mychannel	ISU001	ENDORSER_TRANSACTION	datashare	2018-10-27T02:43:19.640Z
ARCHITECTS	mychannel	PHDBC	ENDORSER_TRANSACTION		

Figure 12. Transactions in stage 2 have been successfully shared.

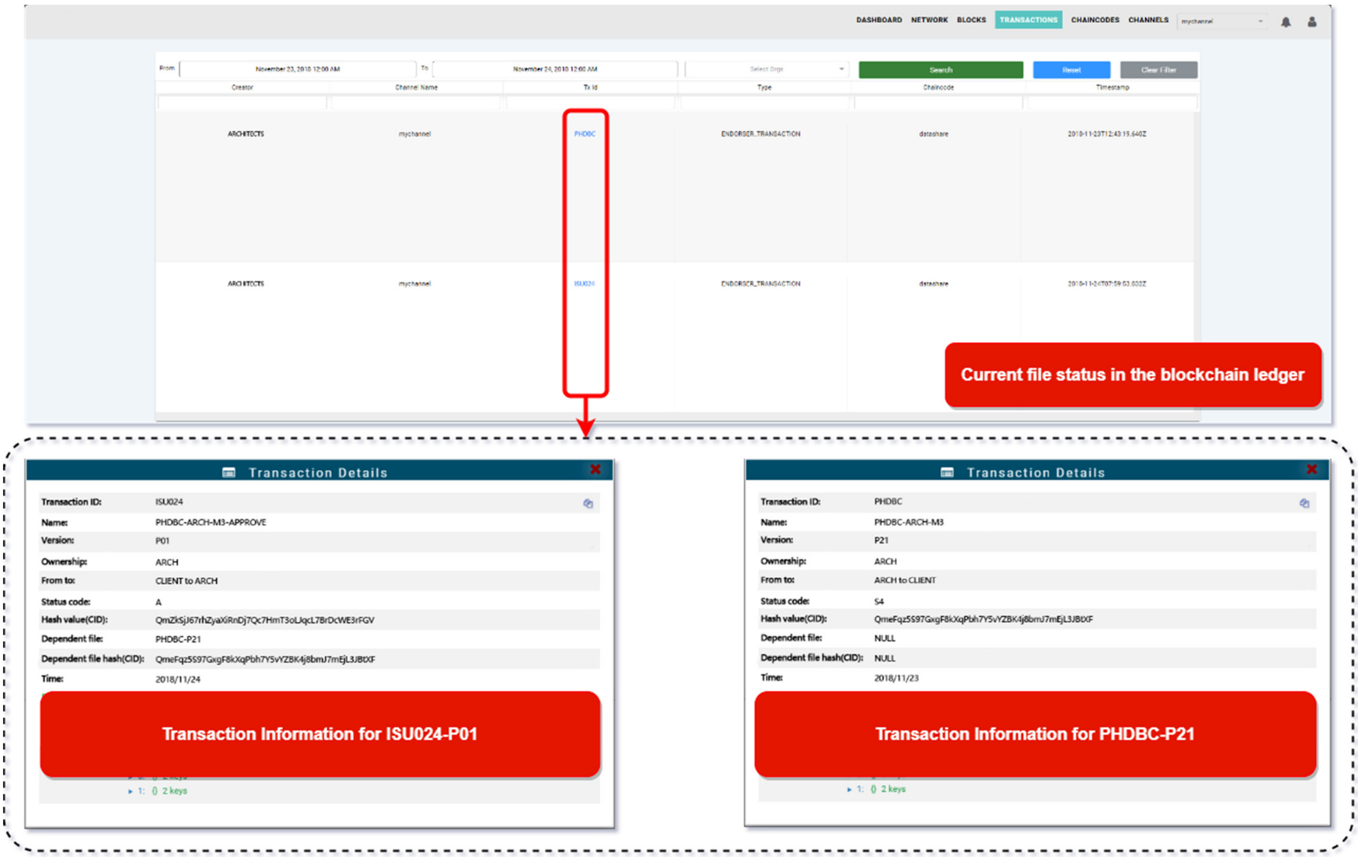


Transaction data	
Transaction ID	*PHDBC*
Name	*PHDBC-ARCH-M3*
Version	*P21*
Ownership	*ARCH*
From to	*ARCH to CLIENT*
Status code	*S4*
Hash value(CID)	*QmeFqz5S97GxgF8kXqPbh7Y5vYZB K4j8bmJ7mEjL3JBtXF*
Dependent file	*NULL*
Dependent file hash(CID)	*NULL*
Time	*23-11-2018*

Transaction data	
Transaction ID	*ISU024*
Name	*PHDBC-ARCH-M3-APPROVE*
Version	*P01*
Ownership	*ARCH*
From to	*CLIENT to ARCH*
Status code	*A*
Hash value(CID)	*QmZkSjJ67rhZyaXiRnDj7Qc7HmT3o LJqcL7BrDcWE3rFGV*
Dependent file	*PHDBC-P21*
Dependent file hash(CID)	*QmeFqz5S97GxgF8kXqPbh7Y5vYZB K4j8bmJ7mEjL3JBtXF*
Time	*24-11-2018*

Figure 13. Stage 3 flowchart: change model status from SHARED to PUBLISH.

After reviewing the PHDBC-P21 model, the client team agreed to the architect's request. The client then initiated a new transaction containing the CID of the approval document (ISU024-P01) and the related HBIM model's CID (AHR001-P21), and shared this transaction through smart contracts. With the completion of this transaction, all project members received notification of the new block, indicating that the publication file's status code was updated to A, signifying that the relevant information had been officially permitted to move to the "PUBLISH" container, ready for the subsequent construction phase. Figure 14 clearly shows the successful sharing of two key transactions within this stage, thereby verifying the application of the AHR-DCDE framework in ensuring the architectural heritage restoration project.



From November 23, 2018 12:00 AM To November 24, 2018 12:00 AM

Creator	Channel Name	Tx Id	Type	Chaincode	Timestamp
ARCHITECTS	mychannel	PHDBC	ENDORSER_TRANSACTION	datashare	2018-11-23T12:43:19.640Z
ARCHITECTS	mychannel	ISU024	ENDORSER_TRANSACTION	datashare	2018-11-24T07:59:53.032Z

Current file status in the blockchain ledger

Transaction ID:	ISU024
Name:	PHDBC-ARCH-M3-APPROVE
Version:	P01
Ownership:	ARCH
From to:	CLIENT to ARCH
Status code:	A
Hash value(CID):	QmZkSjJ67rhZyaXiRnDj7Qc7HmT3oLJqcL7BrDcWE3rFGV
Dependent file:	PHDBC-P21
Dependent file hash(CID):	QmeFqz5S97GxgF8kXqPbh7Y5vYZBK4j8bmJ7mEjL3JBtXF
Time:	2018/11/24

Transaction Information for ISU024-P01

Transaction ID:	PHDBC
Name:	PHDBC-ARCH-M3
Version:	P21
Ownership:	ARCH
From to:	ARCH to CLIENT
Status code:	S4
Hash value(CID):	QmeFqz5S97GxgF8kXqPbh7Y5vYZBK4j8bmJ7mEjL3JBtXF
Dependent file:	NULL
Dependent file hash(CID):	NULL
Time:	2018/11/23

Transaction Information for PHDBC-P21

Figure 14. Transactions in stage 3 have been successfully shared.

4.3. Subsection Framework Performance Evaluation

To quantitatively evaluate the performance of the AHR-DCDE framework, the study established specific testing conditions to measure three main performance indicators: network latency, throughput, and storage costs [45]. The specific conditions are as follows:

1. Development Environment: The AHR-DCDE framework was developed based on Hyperledger Fabric v1.4 and deployed on the Ubuntu 16.04 (Linux) operating system.
2. Design Team Simulation: Considering the project involves four design teams, four virtual machines (or Docker containers) were used to simulate these teams. Each virtual machine was configured with an Intel(R) Core(TM) i5-12600H CPU and 8 GB RAM.
3. Performance Evaluation: Hyperledger Tape (HT), a streamlined benchmark tool, was employed to assess the latency and throughput of the Hyperledger Fabric blockchain network. The assessment unfolded over ten rounds, during each of which five blocks were produced, with every block encapsulating ten transactions.
4. Storage Cost Evaluation: To estimate storage costs, it was assumed that a project could generate 500 design change transactions in a day. The AHR-DCDE architecture was configured to process ten transactions per second.

4.3.1. Latency and Throughput

In this study, we measured the latency from sending a request to the blockchain to receiving confirmation. The operations for uploading and querying transactions were executed by automatically invoking the smart contracts developed in Section 3.3 using the HT tool to test latency times. Based on the results of ten rounds of testing (as shown in Figure 15), the average latency time for uploading transactions was 156 milliseconds, and the average latency time for querying transactions was 137 milliseconds. These latency times are very short, at the millisecond level, and therefore the impact on performance is negligible.

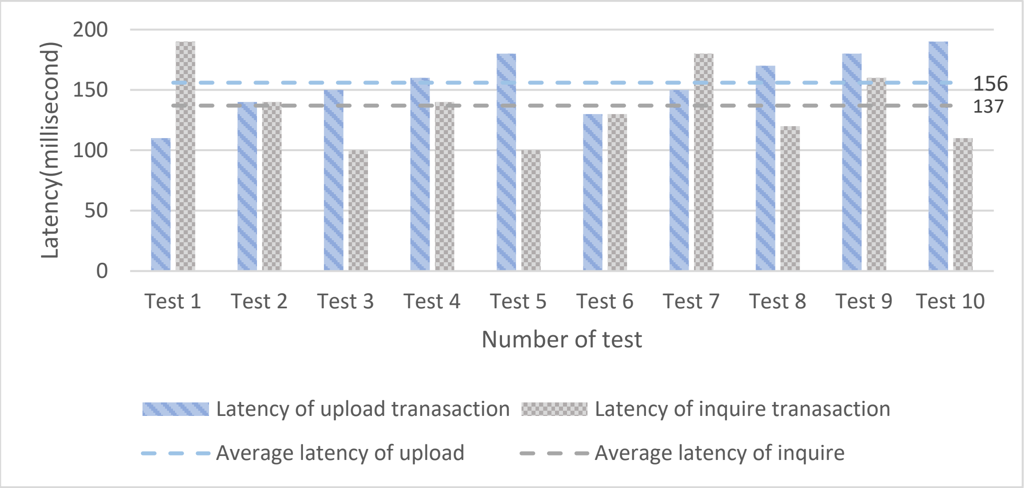


Figure 15. Latency of uploading and inquiring design transactions in the AHR-DCDE.

In this study, we measured the transactions per second (TPS) and queries per second (QPS) to evaluate the AHR-DCDE architecture's capability to process upload and query transactions per second. Based on the test results shown in Figure 16, the architecture can upload an average of 65 transactions per second and query an average of 64 transactions per second. This indicates that the AHR-DCDE architecture can meet the needs of the vast majority of collaborative design work for architectural heritage restoration. It is important to note that performance (in terms of latency and throughput) may vary with different network configurations.

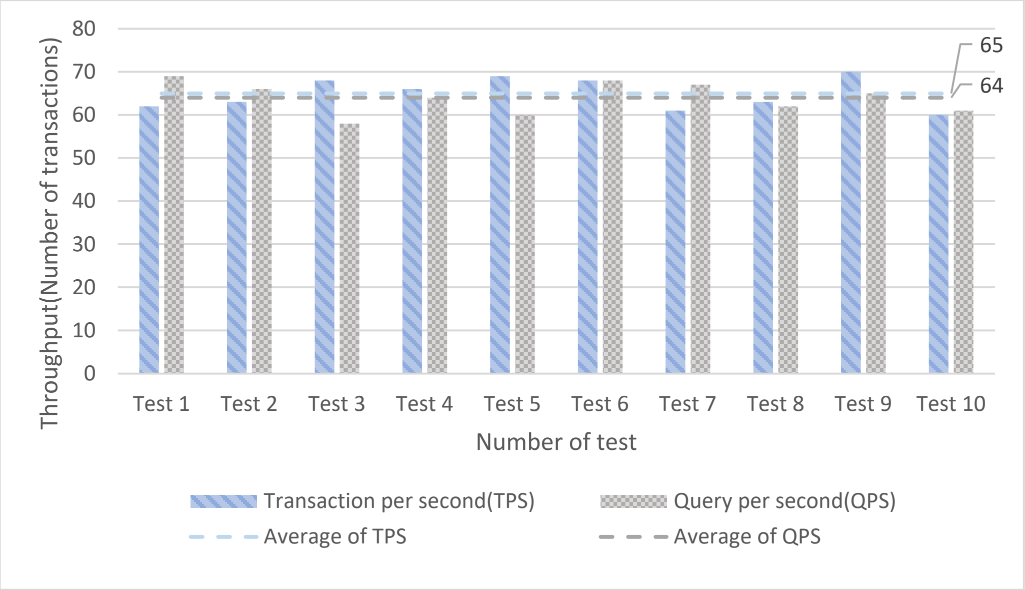


Figure 16. Throughput of AHR-DCDE.

4.3.2. Storage Cost

In this study, we calculated the average size of transaction data to be 187 bytes (as shown in Table 3). Since each block contains ten transactions, the size of a single transaction data expands to 1870 bytes. According to existing research, some elements in the blockchain structure, such as the block header and Merkle tree, have a fixed size. Figure 17 shows the structure of the Merkle tree in the AHR-DCDE system, where each node contains a fixed-length hash string (32 bytes). Therefore, the total size of the Merkle tree in the AHR-DCDE is 32 multiplied by 19, which equals 608 bytes. Based on this, the total size of a single block is approximately 2.6 kilobytes (KB), as shown in Table 4. If 500 transactions are generated each day, these transactions would be stored in 50 blocks. If 50 blocks are produced each day, then the daily storage cost per member would be approximately 13 KB, which is within an acceptable range.

Table 3. Size of transaction data.

Item	Size (Byte)
ID	6 B
Name	22 B
Version	3 B
Ownership	6 B
From to	15 B
Status code	2 B
Hash value (CID)	64 B
Dependent file	9 B
Dependent file hash	50 B
Time	10 B
Total	187 B

Source of the form: Reference [39].

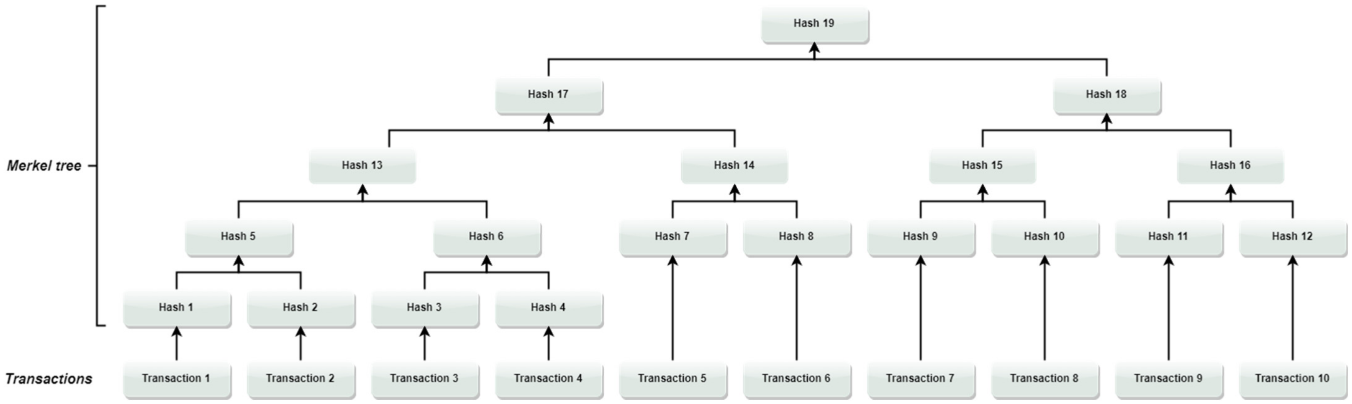


Figure 17. Merkle tree structure in AHR-DCDE.

Table 4. Size of blocks in AHR-DCDE.

Item	Size (Byte)
Hash of previous block	32 B
Hash of current block	32 B
Merkle root	32 B
Timestamp	15 B
Merkle tree Hash number	608 B
Transactions of design record	1870 B
Total	2589 B

Source of the form: Reference [39].

5. Discussion

The innovations of this study are primarily manifested in:

1. The proposition of the AHR-DCDE framework that integrates blockchain and IPFS technologies. The traditional HBIM model adopts a centralized data management method, which faces risks such as data access denial or data loss. The proposed AHR-DCDE framework effectively solves the problem of handling large files in collaborative design for architectural heritage restoration and ensures the security of the HBIM-based design process. Utilizing cryptographic hash technology, a one-way encryption method that irreversibly transforms data from its original form (plaintext) into an encrypted form (ciphertext), the AHR-DCDE framework ensures the immutability of data. The blockchain is responsible for providing immutable storage for design records, while the IPFS ensures the integrity of design files. By adopting a distributed data storage approach, the AHR-DCDE achieves equivalency between project members and access to multiple data sources. In addition, each member maintains a full blockchain ledger, ensuring multiple storage and backup of design records.
2. This study developed key technical components required for the AHR-DCDE framework to support its feasibility. Firstly, a transaction data model was designed in this study aimed at facilitating information exchange within the blockchain network. This data model provides design teams with comprehensive collaboration information, enabling them to quickly and accurately identify and resolve model issues. The design of the transaction data model strictly adheres to the ISO 19650 [9,10] standard and blockchain data format requirements to ensure compatibility across different applications. Secondly, corresponding smart contracts were developed to manage design information in the blockchain ledger, allowing project members to efficiently upload new transactions or query existing information in the ledger.
3. This study expands the application domain of blockchain technology in the architectural heritage restoration industry. Existing research shows that the application of blockchains in the construction industry has a wide range of research prospects. Many scholars have applied blockchain technology to the construction industry and achieved rich research results. Starting from two unique perspectives, this study first proposes a conceptual method for applying blockchain technology to the field of architectural heritage restoration. Firstly, by integrating blockchain technology into the CDE (Common Data Environment) it has facilitated collaborative design based on HBIM (Historical Building Information Modeling), marking one of the preliminary explorations of this technology's application in this field. Secondly, this study further extends the theoretical blockchain model into a practical and feasible solution for the architectural heritage restoration industry, thus broadening the boundaries of existing research and providing a new methodological framework.

Potential practical implications of the AHR-DCDE framework introduced in this study include:

1. Maximizing Trust to Facilitate Collaboration: In existing HBIM collaborative projects, the lack of trust between project members is a major challenge. Project members are concerned that ownership of HBIM data may be lost or that data may be illegally altered. This leads to a tendency to withhold their data, thereby resisting collaboration. The AHR-DCDE framework, by providing an immutable data storage solution, establishes a trustworthy collaborative environment. In such an environment, project members are more willing to exchange information, thereby enhancing the overall efficiency of the project. Moreover, blockchain technology ensures the authenticity and traceability of project data without the need for third-party intermediaries, significantly reducing the time cost required to resolve disputes or provide proof of payment.
2. Driving the Digital Transformation of the Construction Industry: By integrating blockchain technology within a Common Data Environment (CDE), this study show-

cases the potential of blockchain as a foundational data storage technology. Blockchain can eliminate barriers encountered in adopting other information technologies such as the HBIM, the Internet of Things (IoT), and the Geographic Information System (GIS), accelerating the digital transformation of the construction industry. Furthermore, the integration of the blockchain with other technologies offers new avenues for developing decentralized software and systems for the industry. This method of digital information sharing allows project members and their technical expertise to be highly customized within the blockchain network, free from third-party constraints. This lowers the barrier to digital collaboration, paving the way for further digital transformation in the construction industry.

6. Conclusions

This study, through the introduction of the AHR-DCDE framework, showcases a novel and efficient method for collaborative design in architectural heritage restoration. The development and implementation of the AHR-DCDE framework not only integrate the latest blockchain and IPFS technologies but also offer innovative solutions to key challenges long faced in the field of architectural heritage restoration, such as data security, collaboration efficiency, and file management. By conducting an in-depth analysis of the specific case of the Pinghe Packing Factory heritage restoration design project in Wuhan, Hubei Province, this study not only validates the feasibility and benefits of the AHR-DCDE framework in practical application scenarios but also demonstrates its tremendous potential in promoting the conservation and restoration of architectural heritage. The main conclusions of this study are summarized as follows:

1. Technical Innovation: The implementation of the AHR-DCDE framework highlights the innovative application of blockchain and IPFS technologies in the field of architectural heritage conservation. This framework ensures the immutability and traceability of data, significantly enhancing the security and transparency of project data, and provides a solid data support platform for architectural heritage restoration projects.
2. Collaboration Efficiency Enhancement: By facilitating seamless collaboration among interdisciplinary teams, the AHR-DCDE framework enables project participants to access the latest design documents and materials in real time, significantly improving the efficiency of decision-making and the quality of design solutions. This efficient collaboration model brings unprecedented workflow optimization to complex architectural heritage restoration projects.
3. Performance Optimization: By utilizing IPFS technology, the AHR-DCDE framework successfully overcomes the performance bottlenecks encountered by traditional CDEs in processing and sharing large HBIM model files. It provides an effective solution strategy for the storage and fast retrieval of large volumes of data, thereby significantly enhancing the operational efficiency of the project.
4. Cost-Effectiveness Analysis: Preliminary evaluations indicate that the AHR-DCDE framework performs excellently in aspects such as network latency, data throughput, and storage costs, meeting the demands for efficient data processing and sharing in architectural heritage restoration projects. This proves its economic benefits in practical applications.

This study is not without limitations; the technical components of the AHR-DCDE framework, such as transaction data models and smart contracts, do not have the principle of universality. When applied to different heritage restoration projects, the corresponding technical components may need to be adapted. Therefore, it is necessary for the operating members to have certain software development capabilities to adjust the functions of their technical components under different project requirements. In addition, due to the characteristics of distributed technology, there may be delays in actual projects, which are related to the differences in hardware configurations among project members. Therefore, network optimization solutions need to be developed when the AHR-DCDE framework is fully implemented in architectural heritage restoration projects in the future. With the continuous advancement of blockchain and IPFS technologies, the performance and user experience of the AHR-DCDE framework can also be optimized in the future to better meet the evolving needs of the architectural heritage conservation sector. In summary, the introduction of the AHR-DCDE framework not only brings a fresh technological perspective and solutions to the field of architectural heritage restoration and collaborative design but also has profound significance and value for promoting the effective protection and restoration of architectural heritage globally.

Author Contributions: Conceptualization, C.Z. and X.D.; methodology, C.Z. and X.D.; software, C.Z. and Y.Z.; validation, C.Z., X.D., and Y.Z.; resources, H.Y.; data curation, J.Z. and Z.R.; writing—review and editing, X.D.; supervision, H.Y.; project administration, Y.Z.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Major Program (JD) of Hubei Province (2023BAA007).

Data Availability Statement: Data are available from the corresponding author and can be shared upon reasonable request. The data are not publicly available due to confidentiality.

Acknowledgments: We would like to thank Hubei University of Technology for providing technical guidance for this research. Secondly, we would like to thank Central South Architectural Design Institute Co. for providing technical support. Finally, we would like to thank the leadership of CITIC General Institute of Architectural Design and Research Co. for their support.

Conflicts of Interest: Author Hao Yang was employed by the company Central South Architectural Design Institute Co., Ltd. This paper is a major research project of Hubei Province. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

References

1. Jagielska, B.A.; Pszczyński, M.; Stec, P. Cultural Heritage Education in UNESCO Cultural Conventions. *Sustainability* **2021**, *13*, 3548. [CrossRef]
2. Mohamed, Y.M.A.B.; Mahmood, K. Post-Conflict Safeguarding of Built Heritage: Content Analysis of the ICOMOS Heritage at Risk Journal, 2000–2019. *Sustainability* **2023**, *15*, 12364.
3. Alison, H.; Stavroula, G. Conservation science: Reflections and future perspectives. *Stud. Conserv.* **2015**, *60*, 2–6.
4. ICOMOS. Available online: https://www.icomos.org/en/participer/179-articles-en-francais/ressources/charters-and-standards/157-the-venice-charter (accessed on 6 November 2023).
5. Wen, L.; Yahaya, A.; Hazrina, H.B.M. The development of the concept of architectural heritage conservation and its inspiration. *Built Herit.* **2023**, *7*, 21.
6. Bastem, S.S.; Cekmis, A. Development of historic building information modelling: A systematic literature review. *Build. Res. Inf.* **2022**, *50*, 527–558. [CrossRef]
7. Diara, F. HBIM Open Source: A Review. *ISPRS* **2022**, *11*, 472. [CrossRef]
8. Isabel, J.P.; Patricia, T.; Jorge, G.V.; Eugenio, P. Protocol to Manage Heritage-Building Interventions Using Heritage Building Information Modelling(HBIM). *Sustainability* **2018**, *10*, 908.
9. ISO 19650-1:2018. Available online: https://www.iso.org/standard/68078.html (accessed on 18 December 2023).
10. ISO 19650-5:2020. Available online: https://www.iso.org/standard/74206.html (accessed on 18 December 2023).
11. Issa, J.R.; John, I.M.; Ehsan, M. IFC-Based BIM-to-BEM Model Transformation. *J. Comput. Civ. Eng.* **2020**, *34*, 2–4.
12. Minho, O.; Jaewook, L.; Seung, W.H.; Yongwook, J. Integrated system for BIM-based collaborative design. *Autom. Constr.* **2015**, *58*, 196–206.
13. Construction Management Software | Autodesk BIM 360. Available online: https://www.autodesk.com/bim-360/ (accessed on 6 November 2023).
14. Omer, H.A.E.G.; Gokce, G.R.L. *Information Sciences and Systems 2015: 30th International Symposium on Computer and Information Sciences(ISCIS 2015)*, 1st ed.; Springer International Publishing: Cham, Switzerland, 2016; pp. 20–21.
15. Abdul, M.M.; Lamine, M.; Colin, B. Challenges to BIM-cloud integration: Implication of security issues on secure collaboration. In Proceedings of the 2013 IEEE International Conference on Cloud Computing Technology and Science, Bristol, UK, 5 December 2013.
16. Alkhateeb, A.; Catal, C.; Kar, G.; Mishra, A. Hybrid Blockchain Platforms for the Internet of Things (IoT): A Systematic Literature Review. *Sensors* **2022**, *22*, 1304. [CrossRef]
17. Xingyu, T.; Peter, K.Y.W.; Yuqing, X.; Yuhan, L.; Xingbo, B.; Chengliang, Z.; Moumita, D.; Jace, C.P.C. Smart contract swarm and multi-branch structure for secure and efficient BIM versioning in blockchain-aided common data environment. *Comput. Ind.* **2023**, *149*, 103922.

18. Denis, J.S.; Tim, B.; Ling, M. Exploratory literature review of blockchain in the construction industry. *Autom. Constr.* **2021**, *132*, 103914.
19. Yang, L. The Blockchain: State-of-the-Art and Research Challenges. *J. Ind. Inf. Integr.* **2019**, *15*, 80–90.
20. Yanling, N.; Baolin, S.; Yucai, W. Blockchain-based BIM Digital Project Management Mechanism Research. *IEEE Access* **2021**, *9*, 161342–161351.
21. Sommez, R.; Ahmadisheykhsarmast, S.; Gungor, A.A. BIM integrated smart contract for construction project progress payment administration. *Autom. Constr.* **2022**, *139*, 104294. [CrossRef]
22. Nawari, O.N.; Shriraam, R. Blockchain technology and BIM process: Review and potential applications. *J. Inf. Technol. Constr.* **2019**, *24*, 209.
23. Discover What's Out There with IPFS. Available online: https://ipfs.tech/ (accessed on 18 December 2023).
24. Livio, D.L.; Chawee, B.; Chiara, S.; Philippe, V.; Michel, F. A semantic-based platform for the digital analysis of architectural heritage. *Comput. Graph.* **2011**, *35*, 227–241.
25. Murphy, M.; McGovern, E.; Pavia, S. Historic Building Information Modelling (HBIM). *Struct. Surv.* **2009**, *27*, 311–327. [CrossRef]
26. Coşgun, N.T.; Çügen, H.F.; Arslan Selçuk, S. A Bibliometric Analysis on Heritage Building Information Modeling (HBIM) Tools. *ATA Planlama ve Tasarım Dergisi* **2021**, *5*, 61–80. [CrossRef]
27. Antonopoulou, S.; Bryan, P. *BIM for Heritage: Developing a Historic Building Information Model*; Historic England: London, UK, 2017.
28. Nieto-Julián, J.E.; Farratell, J.; Bouzas Cavada, M.; Moyano, J. Collaborative Workflow in an HBIM Project for the Restoration and Conservation of Cultural Heritage. *Int. J. Archit. Herit.* **2022**, *11*, 1813–1832. [CrossRef]
29. Radl, J.; Kaiser, J. Benefits of Implementation of Common Data Environment (CDE) into Construction Projects. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *471*, 022021. [CrossRef]
30. Preidel, C.; Borrmann, A.; Oberender, C.-H.; Tretheway, M. *Seamless Integration of Common Data Environment Access into BIM Authoring Applications: The BIM Integration Framework*; CRC Press: Boca Raton, FL, USA, 2016.
31. Implementation of a Common Data Environment. Available online: https://www.scottishfuturestrust.org.uk/publications/documents/cde-implementation-research (accessed on 18 December 2023).
32. BIMcloud. Available online: https://graphisoft.com/solutions/bimcloud (accessed on 18 December 2023).
33. Parn, E.A.; Edwards, D. Cyber Threats Confronting the Digital Built Environment: Common Data Environment Vulnerabilities and Block Chain Deterrence. *ECAM* **2019**, *26*, 245–266. [CrossRef]
34. Qian, X.; Papadonikolaki, E. Shifting Trust in Construction Supply Chains through Blockchain Technology. *ECAM* **2020**, *28*, 584–602. [CrossRef]
35. Hsu, C.-L.; Wang, J.-T.; Hou, H.-Y. A Blockchain-Based Parametric Model Library for Knowledge Sharing in Building Information Modeling Collaboration. *J. Constr. Eng. Manag.* **2023**, *149*, 04023107. [CrossRef]
36. Tao, X.; Liu, Y.; Wong, P.K.-Y.; Chen, K.; Das, M.; Cheng, J.C.P. Confidentiality-Minded Framework for Blockchain-Based BIM Design Collaboration. *Autom. Constr.* **2022**, *136*, 104172. [CrossRef]
37. Sun, J.; Yao, X.; Wang, S.; Wu, Y. Blockchain-Based Secure Storage and Access Scheme For Electronic Medical Records in IPFS. *IEEE Access* **2020**, *8*, 59389–59401. [CrossRef]
38. Yu, R.; Li, Z.; Chen, Z.; Ding, G. Digital Rights Management System of Media Convergence Center Based on Ethereum and IPFS. *IEICE Trans. Inf. & Syst.* **2023**, *E106*, 1275–1282.
39. Tao, X.; Das, M.; Liu, Y.; Cheng, J.C.P. Distributed Common Data Environment Using Blockchain and Interplanetary File System for Secure BIM-Based Collaborative Design. *Autom. Constr.* **2021**, *130*, 103851. [CrossRef]
40. Nawari, N.O.; Ravindran, S. Blockchain and the Built Environment: Potentials and Limitations. *J. Build. Eng.* **2019**, *25*, 100832. [CrossRef]
41. Hyperledger Fabric. Available online: https://www.hyperledger.org/projects/fabric (accessed on 18 December 2023).
42. Gaba, P.; Raw, R.S.; Mohammed, M.A.; Nedoma, J.; Martinek, R. Impact of Block Data Components on the Performance of Blockchain-Based VANET Implemented on Hyperledger Fabric. *IEEE Access* **2022**, *10*, 71003–71018. [CrossRef]
43. Sheng, D.; Ding, L.; Zhong, B.; Love, P.E.D.; Luo, H.; Chen, J. Construction Quality Information Management with Blockchains. *Autom. Constr.* **2020**, *120*, 103373. [CrossRef]
44. Cai, L.; Gong, Q.; Jiang, F.; Yuan, M.; Xiao, Z.; Zhang, S.; Zheng, C.; Wu, Y. Retracted: The Use of the Intelligent Bayesian Network Method Combined with Blockchain Technology in the Optimisation of Tunnel Construction Quality Control. *IET Software* **2023**, *17*, 776–786. [CrossRef]
45. Li, X.; Wu, L.; Zhao, R.; Lu, W.; Xue, F. Two-Layer Adaptive Blockchain-Based Supervision Model for off-Site Modular Housing Production. *Comput. Ind.* **2021**, *128*, 103437. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.