BIM Information Management Throughout the Lifecycle of Multi-Story Civil Buildings

Yixin Zhang [1, *], Xinyu Wei [2], and Weiye Huang [3]

[1] International Education School, Suzhou University of Science and Technology, 215009 Suzhou, China

[2] College of Architecture and Art, Taiyuan University of Technology, 030024 Taiyuan, China

[3] School of Civil Engineering and Architecture, Wuhan Polytechnic University, 430023 Wuhan, China

* Corresponding Author Email: 23200142102@post.usts.edu.cn

Abstract. Against the backdrop of digital transformation in the construction industry, BIM is widely adopted; however, challenges such as delayed information management, insufficient research on multi-stage information integration, and unsmooth information interaction during technology implementation. To address these issues, this study takes a self-established six-story dormitory building model as the research object and proposes a full-life-cycle management approach integrating BIM with construction schedule, projection cost, and operational and maintenance. Unlike traditional plug-in-based methods that modify native model files, the proposed workflow preserves data integrity throughout the process. Relying on the Revit platform this research integrates Dynamo-based visual programming with Python scripting and adopts a non-invasive methodology. A five-step process is employed to establish a multi-dimensional parametric integration management system. This effectively achieves zero-intrusive, multi-dimensional, and realize low-threshold information management for building engineering projects, while preventing model data corruption, enhancing development efficiency, and reducing migration costs. The proposed approach provides a viable pathway for the standardized and scalable application of BIM full- life-cycle management.

Keywords: BIM, full‑life‑cycle management, visual programming, non‑invasive methodology, building engineering.

1. Introduction

In the era of digital and intelligent transformation in the construction industry, advances in next-generation information technologies have facilitated comprehensive digital transformation and upgrades across various sectors [1]. Effective monitoring is crucial for the success of construction projects, involving the continuous and systematic collection, analysis, and interpretation of data to evaluate progress, performance, and compliance with established schedules, budgets, and quality standards [2]. With its core advantages of visualization, parameterization, and collaboration, Building Information Modeling (BIM) technology has emerged as a key enabler in breaking down information silos in traditional engineering management and achieving integrated control throughout the project life cycle, already finding broad application in building engineering [3]. During the design phase, the BIM allows project stakeholders to better understand the project [4]. The standardized delivery of building data to owners upon project completion is critical for efficient and effective facility management during the operational phase. Structured data processing enhances facility management efficiency and shortens the time gap between project completion and data availability [5]. Information management, as the core of efficient operation throughout the life cycle of multi-story civil buildings, enables unified, standardized, and traceable information flow and integrated management. This helps eliminate information fragmentation among different phases and stakeholders, ensures data consistency and accuracy, reduces rework and resource waste, and improves collaboration efficiency and decision-making precision. Moreover, comprehensive information assets provide reliable support for subsequent operations, maintenance, renovation, and safety management, lowering life-cycle costs and extending project value. However, in conventional projects, monitoring often relies on

manual observation of construction activities, periodic reporting, and physical inspections to track progress and identify deviations from planned schedules. This approach frequently results in delayed problem identification and limited real-time visibility into project performance [6]. According to research by the National Institute of Building Sciences (NIBS), approximately 60% of critical operation and maintenance information is lost or underutilized during project delivery in traditional construction, leading to inefficient building operations, increased energy costs, and compromised facility performance [7].

These gaps highlight the current limitations of BIM technology in practical full-life-cycle management of construction projects. Existing research predominantly focuses on the instrumental application of BIM in isolated phases of building projects, with limited systematic investigation into multi-dimensional parametric integrated information management. In implementation, challenges persist, including disconnected dynamic control during construction, inefficient information collection and integration across processes, and poor interoperability among stakeholders [8]. Consequently, the potential of BIM for information integration and collaborative management remains underutilized, falling short of meeting the refined and standardized management demands of building projects over their full life cycle.

In light of the above research context and practical needs, this thesis proposes a full-life-cycle management method that integrates BIM technology with project schedule, cost, and operational data and conducts an in-depth study [9]. Using a six-story dormitory building as a case model and Revit as the base platform, data extraction and analysis of basic components are performed via external Dynamo visual programming files to establish a multi-dimensional parametric integration management system. By employing Python scripts and leveraging native shared parameters, a non-invasive information management approach is analyzed for feasibility without modifying the model or components, thereby obtaining comprehensive parametric information. This method achieves zero-intrusion, multi-dimensional, and low-threshold information management, addressing key challenges in BIM full-life-cycle applications such as difficult implementation, high costs, and poor reusability. It offers a feasible path toward the standardized and scalable adoption of BIM technology in construction project full-life-cycle management.

2. Application of bim technology in the whole life cycle of multi-story civil buildings

2.1. Technical principles of extracting bim component data

Traditional methods of extracting BIM component parameters have many technical drawbacks. In Revit, designers must create custom nodes, which requires traditional programming knowledge and is time-consuming. This can also easily lead to family file corruption or model parameter redundancy. Compiled Add-in plugins require installation and registration and are strictly limited by Revit versions. Team collaboration requires full deployment, resulting in high operating costs. Furthermore, tools developed based on specific shared parameter sets may become directly invalid when facing models from different design institutes or projects due to inconsistent parameter naming, requiring a large amount of adaptation and modification, which increases the workload of the project [10].

In response to the above problems, this thesis completes the data extraction of model components through external files. And only reads the operation through Application Programming Interface (API) throughout the process, without creating, deleting, or modifying any component, family parameters, or shared parameters in the Revit model. It fundamentally eliminates the loss or destruction of model data caused by secondary development operation errors and ensures the integrity of production-level BIM models. At the same time, visual programming based on Dynamo + Python, which exists in the form of a ".dyn" file, requires no compilation or installation, just open Revit and run the script, enabling non-intrusive, lightweight data extraction. Based on the native core parameters of Revit, this thesis classifies and labels components by category, which has strong universality. For different versions of Revit software and different project standards, as long as the components are type-marked,

the script can run seamlessly, greatly reducing the cost of technology migration. In addition, this thesis adopts non-invasive extraction of model component data, which can repeatedly run and modify the script without destroying the model. Even if there is an error with Iterable expected or type mismatch, the logic only needs to be corrected without rolling back the model, greatly improving development efficiency and debugging fault tolerance.

2.2. Method for extracting bim model member data

This study uses a dormitory building (as shown in the Fig.1) as the research model. The extraction of member data from BIM employs a phased operational process consisting of five core steps: component naming and parameter setting, node setting and binding, data extraction and sorting, table export and result verification, and data backfilling and validation.

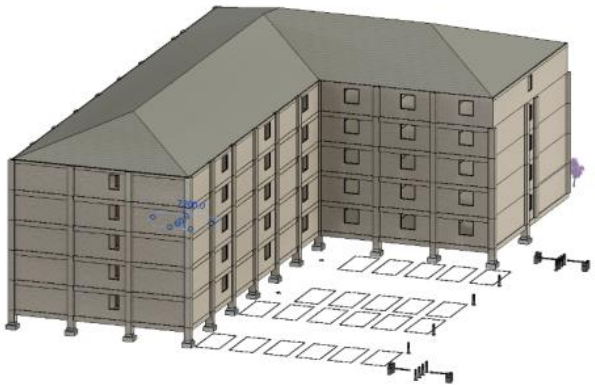


Fig. 1 Revit 3D model of a dormitory building.
(Picture credit : Original)

2.2.1. Component naming and parameter setting

Revit is first opened to standardize the naming of all components of the target model. And renaming according to the category, specialty, floor, function and other dimensions of the components to ensure the uniqueness and readability of the component names and provide a basis for subsequent component search and parameter binding. Then, open the shared parameters function in the "Management" module of the toolbar, and set the core parameters to be extracted from three dimensions: cost, schedule, and Operation and Maintenance (O&M) based on the full lifecycle management requirements. Associate and bind the newly created shared parameters to component categories such as walls, plates, columns, beams, doors, and windows to ensure that each type of component is equipped with corresponding dimension parameters. Finally, all components are reviewed one by one to confirm the integrity and accuracy of parameter binding, in order to avoid affecting subsequent data extraction due to missing parameters or errors.

2.2.2. Node setup and binding

Open the Dynamo visual programming tool to carry out the whole process of precise screening, professional classification, logical association, and structured binding of nodes based on the components that have completed the shared parameter binding in the early stage. In order to build a stable and reusable data flow framework for subsequent efficient data extraction. First, through the filtering node of Dynamo, the invalid and redundant component objects in the model are eliminated, and the effective collection of core building components such as walls, plates, columns, beams, doors and windows is retained. Then, based on the professional attributes (architecture, structure, electromechanical), type characteristics and functional partitioning of components, the standardized definition and binding of vertical headers (Category columns) are completed in the Dynamo workspace. And the components of different categories are accurately mapped into sub-headers under the corresponding category level, so as to realize the one-to-one correspondence between components and headers.

2.2.3. Data extraction and organization

First, filter members by different types, read the type parameters of each member through a Python script node (as shown in the Fig.2), and retain the required parameters. Then, output the filtered member list, which serves as the sole data source for all subsequent nodes. Extract the required data from the filtered member list, perform batch reading through the Parameter.GetBy name method, and generate a two-dimensional data list as output. Subsequently, integrate all custom parameters into a large list using List. Create. Then, transpose it with List. Transpose to change "store by column" to "store by row," forming a standard table structure. Finally, the null and abnormal values are simply handled. Ensure that the exported table data is complete and unified, and provide standardized data interfaces for subsequent cost accounting, progress tracking, and operation and maintenance management.

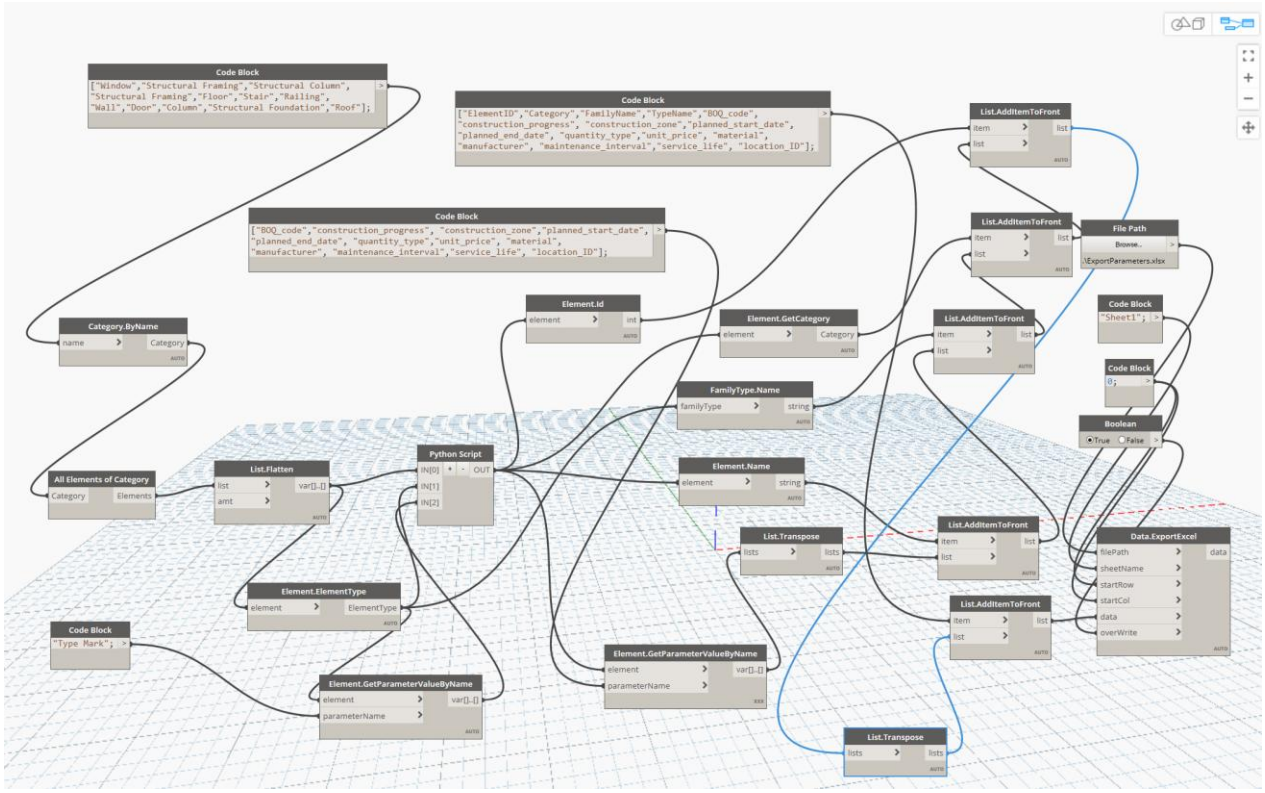


Fig. 2 Dynamo visual programming flowchart.

(Picture credit : Original)

2.2.4. Table export and result verification

The component data that has been transposed to form a standard table structure is associated with the Excel. Write To node in Dynamo. At the same time, complete the precise configuration of export parameters: specify the exclusive file storage path, set the Excel file name to ExportParameters.xlsx, and the worksheet name to Sheet1. The export start row and column are set to the first row and the first column and overwrite: True is enabled to ensure the timeliness of data updates. After completing the configuration, run the node to export the parameter data of the model components to the Excel table. After the export work is completed, carry out two-way verification and verification immediately. Through random sampling, the core content such as component parameter values, component numbers, table header fields, and data formats is checked, focusing on verifying the accuracy of parameter extraction and data integrity. Ensure that the exported table data can be directly connected to subsequent cost accounting, progress tracking, and operation and maintenance management, without data missing or numerical deviations.

2.2.5. Data backfilling and verification

Based on the exported and verified excel parameter tables, the Dynamo visual programming platform builds an exclusive backfill node process and writes adaptive Python scripts. A precise parameter mapping channel between the excel table and the Revit model is constructed to realize the automatic and batch backfilling of external table parameters to the Revit model component property panel (as shown in the Fig.3). The processed parameter data is written to the attribute panel of the corresponding component to realize the batch binding and update of parameters. The interaction is completed only through API writing operations, and no additional redundant parameters are added to maintain the cleanliness of the model data.

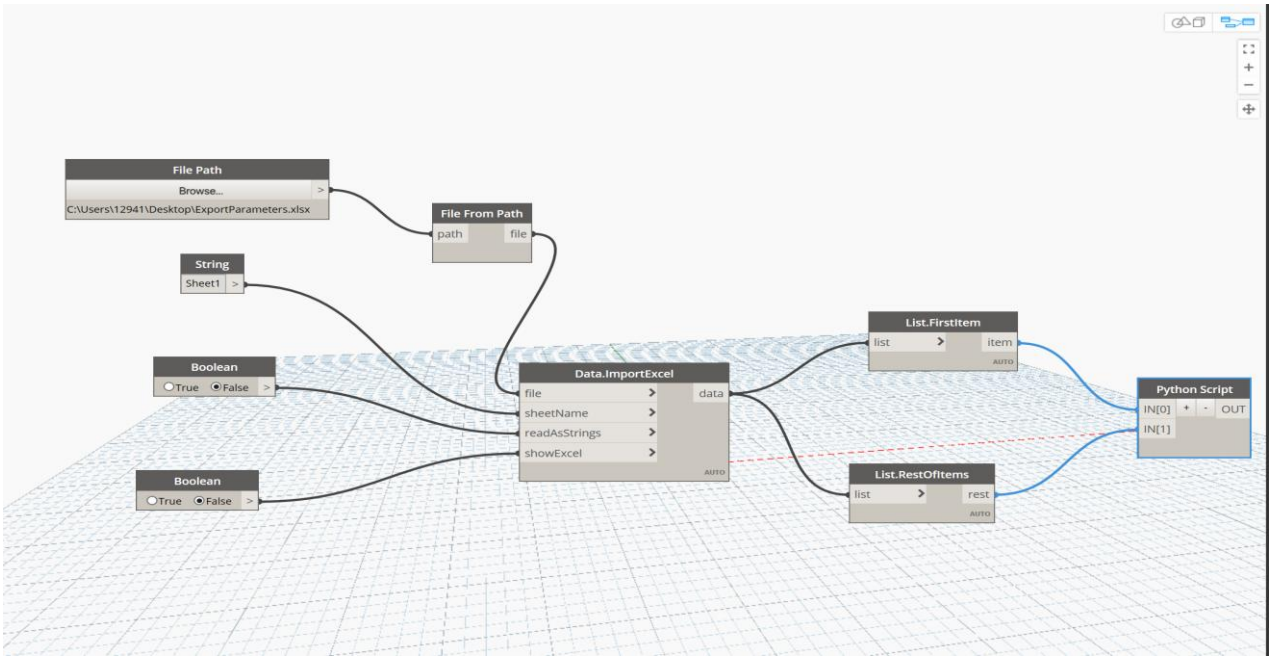


Fig. 3 Data backfilling flow chart.

(Picture credit : Original)

Due to the inherent limitations of the Revit software view display mechanism, all attribute parameters of model components cannot be displayed directly in batches in the view. And the parameter configuration status can be viewed by manually clicking on a single component to retrieve the attribute panel. In order to verify the feasibility, accuracy and stability of the data backfilling method in this study, a large number of door components with typical parameter types in housing construction projects were selected as the verification objects. Manually click to retrieve the attribute panel of the door component to check the completeness, numerical accuracy and field matching of the backfilled parameters (as shown in the Fig. 4 and Fig. 5). It is fully proved that the parameter backfilling method proposed in this study can realize the accurate and efficient update of model component parameters.

ElementID	Category	FamilyName	TypeName	BOQ code	Construction Progress	Construction Zone	Planned Start Date	Planned End Date	Quantity Type	Unit Price	Material	Manufacturer	Maintenance Interval	Service Life	Location ID
286792	Window	Fixed	900 x 1200mm	BOQ-Wn-F-001	Finished	Zone2	05-12-2024	05-12-2024	Set	2100	Broken Bridge Aluminum + Insulated Glass	B	12	25	L-Wn-F-001
369554	Structural Column	Concrete-Rectangular Column	750 x 750mm	BOQ-F-001	Finished	Zone1	15-03-2024	15-03-2024	m^3	680	C-30 Concrete	A	12	50	1001

Fig. 4 Export of the result of the gate component excel.

(Picture credit : Original)

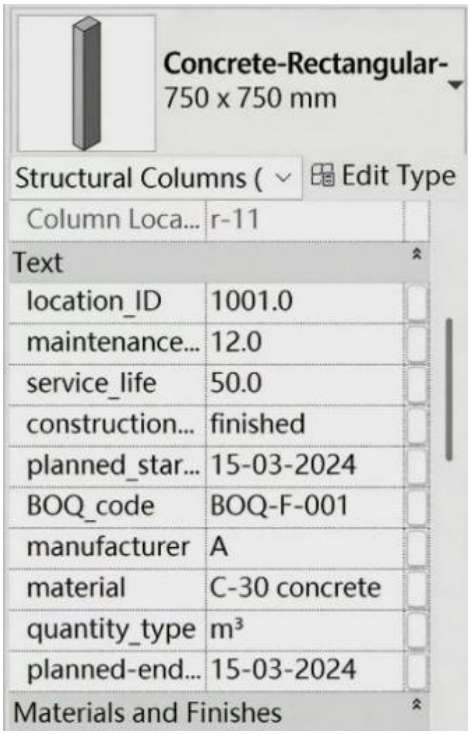


Fig. 5 Display of the backfilling results of door components.

(Picture credit : Original)

2.3. Research results

Construction projects require complex structures, involve numerous participants, and have long construction cycles. Extracting and utilizing information from all stages of the entire cycle of construction projects is one of the current research directions in the construction industry. This study focuses on the combination of BIM and full-life-cycle management. After reading a large number of literature and conducting various analyses and studies, it summarizes the existing problems in the process of extracting model component information. This study takes a six-floor dormitory building as the research model and uses Revit as the base platform to achieve data extraction and analysis of the basic components of the research model through external files via the Dynamo visual programming tool, and has established a multidimensional parametric integrated management system. The main conclusions and innovations of this study are as follows:

Construction a lightweight, automated, and non-intrusive BIM full-life-cycle parameter management technology system. Based on Dynamo visual programming and the Revit API, it achieves full-process automation from model parameter extraction to Excel editing, and finally batch backfilling. Overcoming the interference of intrusive development on the original structure modeling, through a non-intrusive interaction of 'read-only, no write,' efficient management of the entire lifecycle of engineering projects was achieved without modifying any model attributes or contaminating modeling data, forming an innovative technical solution of 'low-code-needing, high-security, full-process'. It has eliminated the technical threshold of needing to write a large number of redundant scripts, enabling automation control of progress, cost, and operation and maintenance through lightweight code logic. It has broken through the limitation of Revit where building property lists require manual input, achieving precise positioning of building parameters based on element Identifier (ID) and batch entry. This proves the engineering practicality of non-intrusive beam-driven technology in the structural design phase.

Optimized the technical barriers of cross-platform data incompatibility. For Excel floating-point Element ID (such as 286792.0), proposed a string-splitting and type-compatibility algorithm, resolving errors such as non-numeric formats and constructor mismatches, achieving a 100% data recognition success rate. Through parameter name mapping and exception handling mechanisms,

undefined parameters in Revit are automatically skipped, ensuring the stability of processes and the safety of the model. In the dormitory building research model, batch parameter backfilling was completed for 2,220 building components, achieving 100% data matching accuracy. The solution increased the efficiency of traditional manual operations by approximately 1,480 times while reducing data entry error rates to zero, forming an efficient BIM management solution characterized by 'low code, high safety, full process,' and providing a feasible technical path for refined management of engineering data.

3. Conclusion

This thesis takes a six-floor dormitory building as a research model and proposes a non-intrusive information management method based on Revit、Dynamo and Python. Through five core steps—component naming and parameter setting, node configuration and binding, among others—a multidimensional parametric integrated management system was established. The study verified the engineering feasibility of the non-intrusive information management model. Data interaction was fully realized through API read and write operations, fundamentally avoiding issues such as model data corruption and parameter redundancy. Meanwhile, relying on lightweight visual programming, it overcomes the version limitations and deployment costs of traditional plugin development. This method enables automated extraction, organization, export, and backfilling of three-dimensional data on progress, cost, and operation and maintenance throughout the entire lifecycle of multi-story residential buildings. In the dormitory building model, 2,220 components achieved 100% data matching, increasing manual operation efficiency by approximately 1,480 times and reducing data entry error rates to zero. It effectively addresses industry pain points such as the difficulty of BIM implementation, high costs, and poor reuse, providing a zero-intrusion, multidimensional, and low-barrier feasible solution for BIM information management throughout the lifecycle of multi-story residential buildings. In the future, it is possible to promote a non-intrusive, lightweight management model to complex engineering fields such as super high-rise buildings and bridges, breaking the barriers of traditional heavy software development and reducing the cost of technology application for small and medium-sized projects. By integrating existing technological logic, we aim to build a standardized Dynamo Python template library, providing reusable interfaces for future researchers, lowering the threshold for secondary development, and accelerating the promotion and application of technology. Additionally, by introducing artificial intelligence algorithms, we plan to upgrade from automated data processing to proactive intelligent early warning and intelligent decision-making, improving full-process intelligent management of progress risks, cost deviations, and maintenance status. We will promote the standardization of 16 core parameter systems, establish a unified industry standard for non-intrusive data management, break down data interoperability barriers across multiple projects and platforms, and support the implementation of digital transformation in the construction industry.

Author Contributions

These authors contributed equally to this work. Yixin Zhang: Conceptualization, Methodology, and Experimentation. Xinyu Wei: Conceptualization, Methodology, Writing – Original Draft, and Writing – Review & Editing. Weiye Huang: Conceptualization, Methodology, Resources, and Writing – Review & Editing.

References

[1] B. Qi, C. Li, Research on the Whole Lifecycle of Prefabricated Construction Management Based on BIM Technology. Construction Technology. 43, 25-29 (2014)

[2] M. Li, J. Tang, From BIM to Digital Twin: Research on the Information Delivery and Update Mechanism for the Building Operation and Maintenance Phase Based on BIM. Architectural Design Management. 43, 84-89 (2026)

[3] Y. Zhang, Explore the Application of BIM Technology in the Intelligent Construction Stage of Prefabricated Buildings. Create Living. 125-127 (2025)

[4] A. Sharafat, M. S. Khan, K. Latif, W. A. Tanoli, W. Park, J. Seo, BIM-GIS-Based Integrated Framework for Underground Utility Management System for Earthwork Operations. Appl. Sci. 11, 5721. (2021) https://doi.org/10.3390/app11125721

[5] F. K. T. Cheung, J. Rihan, J. Tah, Early-stage multi-level cost estimation for schematic BIM models. Autom. Constr. 27, 67-77 (2012)

[6] W. Wang, Analysis of the Impact of BIM Technology on Cost Control in Project Cost Management. Sci. Technol. Innov. 143-145 (2026)

[7] Y. Zhu, R. R. A. Issa, Computing in Civil Engineering. International Workshop on Computing in Civil Engineering (2011)

[8] X. Liu, D. Wan, Z. Su, Research on the Application of Digital Construction Technology in the Field of Full Life-Cycle Project Management, in: ICCREM, pp. 1010-1019 (2023)

[9] H. Alavi, M. Wang, G. Wang, S. Kookalani, I. Brilakis, Digital twins in construction: leveraging point cloud data and BIM for monitoring project schedule, in: Computing in Civil Engineering, (2024)

[10] J. Lucas, W. Thabet, Using a case-study approach to explore methods for transferring BIM-based asset data to facility management systems, in: Construction Research Congress, pp. 439-448 (2018)