

FROM BIM TO INTERACTIVE DIGITAL TWIN: A WORKFLOW FOR INTEGRATING VIRTUAL AND PHYSICAL ENVIRONMENTS

Mujahed Mohammad Khalil Bani Mustafa⁽¹⁾, Manuel Afonso Parente⁽²⁾, Tomo Cerovsek⁽¹⁾

(1) University of Ljubljana, Faculty of Civil and Geodetic Engineering, Slovenia

(2) University of Minho, ISISE, ARISE, Department of Civil Engineering, Guimarães, Portugal

Abstract

The fragmentation among Building Information Models (BIM), IoT sensor networks, and facility management interfaces severely limits the operational value of digital construction investments. The architecture, engineering, and construction (AEC) sector requires practical workflows that extend the use of BIM beyond design and construction into facility operations. This paper presents a validated open-source methodology for transforming static IFC models into interactive augmented virtual reality (VR) digital twins using Unreal Engine 5.5. The approach has been implemented through a 4-level university building case study.

The proposed system combines IFC building geometry with real-time HOBO sensor data (temperature, humidity, and CO₂) using a three-tier navigation architecture that bridges 2D floor plans and 3D spatial exploration. Using Datasmith for model processing and Blueprint visual scripting for interaction logic, the workflow supports both desktop and VR deployment modes without requiring programming expertise. Performance validation demonstrates sub-100 ms response times for all interactions, teleportation for predefined spaces, and consistent 120 FPS desktop/90-120 FPS VR operation while processing live sensor streams.

The multi-modal interface design enables progressive complexity adoption, allowing facility managers to transition from familiar 2D interfaces to immersive 3D environments at their own pace. This democratized approach uses freely available tools to make digital twin technology accessible to educational institutions and small-to-medium enterprises previously excluded by proprietary platform costs. The validated workflow directly provides practical implementation patterns for BIM-to-digital-twin transformation. This solution serves AEC researchers and academic institutions seeking an open, extensible framework for experimental digital twin development, immersive visualization research, and real-time BIM-IoT integration studies without reliance on proprietary platforms.

1. Introduction

The digital transformation of the Architecture, Engineering, and Construction (AEC) industry has generated substantial investment in Building Information Modeling, yet a persistent fragmentation exists between static BIM models, real-time sensor data, and facility management interfaces. Deng et al. [1] propose a five-level taxonomy mapping the evolution from traditional BIM to ideal digital twins, demonstrating that most current implementations remain at early development stages (Levels 2–3), where sensor connectivity and simulation capabilities are still being explored.

This gap also has a particular relevance for Portugal's digital construction transition. Decreto-Lei 10/2024 [2] establishes mandatory BIM adoption milestones, with PEPU (Plataforma Eletrónica para os Procedimentos Urbanísticos) requirements effective from January 2026 and full BIM submission format mandated by 2030. As Portuguese organizations prepare for compliance, the availability of cost-effective digital twin solutions becomes critical, particularly for educational institutions and small-to-medium enterprises that cannot absorb the licensing costs of commercial platforms such as Autodesk Tandem or Bentley iTwin.

Current academic literature reveals a significant imbalance in game engine applications for AEC: Unity dominates existing implementations while Unreal Engine remains notably underrepresented, with only one documented study employing an early-generation engine without modern BIM integration or VR support [3]. This gap constrains evidence-based technology selection for digital twin applications.

This paper presents an open-source BIM-to-digital twin workflow implemented in Unreal Engine 5.5 and validated on the four-story faculty building of the Faculty of Civil and Geodetic Engineering. The building is equipped with a Li-Cor HOBO sensor network deployed within an Industrial Internet of Things (IoT) infrastructure to enable real-time environmental monitoring. The initial prototype was developed at the same faculty as part of an advanced digital twin demonstration project. Previous work leading to this study included the development and iterative updating of the BIM model, as well as its test implementation across several digital twin platforms, including Autodesk Tandem. Based on several technical and research criteria, including interoperability, real-time data integration, scalability, and open-source accessibility, the University of Ljubljana invested in the development of an advanced IoT network to support continuous data acquisition and seamless integration with the digital twin environment.

The key scientific and technical contributions of this study are summarized as follows:

1. Pioneering comprehensive Unreal Engine 5.5 implementation for digital twins in AEC.
2. Workflow for preparation of BIM model for use in VR Facility Management
3. Real-time IoT sensor integration with sub-100 ms interface responsiveness.
4. Cross-platform desktop and VR implementation from a unified codebase, achieving 90–120 FPS.
5. A seamless multi-modal navigation system bridging 2D floor plans with 3D immersive exploration, achieving 0.3 m teleportation accuracy.

The paper is organized as follows: Section 2 presents related work and commercial platform comparison; Section 3 describes the system architecture; Section 4 details implementation; Section 5 presents results; Section 6 discusses findings and limitations; and Section 7 concludes with future directions.

2. Related Work

2.1 Digital Twins in Facility Management

Digital twin adoption in facility management remains constrained by persistent technical and organizational barriers. As established in Section 1, most implementations remain at Levels 2–3 of the taxonomy proposed by Deng et al. [1]. Empirical evidence from operational deployments remains limited. Zhao et al. [4] report a 10% improvement in staff satisfaction and 1% reduction in energy consumption in AEC facility management contexts, representing measurable but modest outcomes. More critically, Signorini and Pomè [5] document that digital twin adoption in facility management remains limited to approximately 13% of surveyed organizations, with implementation costs, integration complexity, and workforce skill gaps cited as primary barriers. Several researchers proposed frameworks to address these challenges. Hosamo et al. [6] develop conceptual models for predictive maintenance applications, while Revolti et al. [7] establish protocols specifically for BIM-to-digital twin transition, focusing on sensor connection methodologies. However, these frameworks largely remain at theoretical or prototype stages, with limited validation through comprehensive operational deployments that include quantified performance metrics and user experience evaluation.

A critical gap persists in the availability of cost-effective, open-source implementation approaches suitable for educational institutions and small-to-medium enterprises. Most documented implementations rely on commercial platforms with significant licensing requirements, creating adoption barriers for resource-constrained organizations [5], [6].

2.2 Game Engines in AEC Applications

Game engines have emerged as viable platforms for architectural visualization, design review, and facility management applications, offering real-time rendering, VR integration, and interactive capabilities that traditional BIM viewers cannot provide. However, current literature reveals a significant imbalance in engine representation within AEC research.

Analysis of game engine applications in AEC contexts demonstrates that Unity dominates current implementations, with documented applications including real-time BIM data exchange through UnityRev [8], interactive design review in healthcare facilities, immersive VR visualization for construction monitoring, and multi-user quality assurance workflows. These implementations validate Unity's capabilities for BIM integration and VR support, while also documenting persistent challenges with performance optimization for large architectural datasets and frame rate consistency in VR environments.

In contrast, Unreal Engine applications in AEC remain notably underrepresented. The only documented implementation employs an early version of the Unreal Tournament engine for basic remote collaboration and 3D visualization, without modern BIM integration or VR capability [3]. This research gap is significant because Unreal Engine offers several technical advantages relevant to facility management applications: superior real-time rendering quality through Lumen global illumination and Nanite virtualized geometry, industry-leading VR performance optimization, the Blueprint visual scripting system that enables rapid prototyping, and the Datasmith pipeline providing native IFC import capability.

The absence of comprehensive Unreal Engine implementations in AEC literature constrains evidence-based technology selection and limits understanding of the engine's potential for digital twin applications requiring high-fidelity visualization and immersive interaction.

2.3 Commercial Platform Comparison

Current commercial digital twin platforms offer varying levels of functionality but share common limitations regarding cost, flexibility, and accessibility. Table 1 presents a comparative analysis of leading platforms against the open-source workflow proposed in this research.

Tabela 1: Comparison of Digital Twin Platforms for Facility Management

Feature	Autodesk Tandem [9]	Bentley iTwin [10]	Unity Reflect [11]	This Work
BIM Import	Revit (native)	Multiple (native)	Revit (plugin)	IFC via Datasmith
IoT Integration	API-based	API-based	Custom development	HOBOT Li-Cor API
VR Support	Limited	Limited	Yes (Unity XR)	Full (Meta Quest)
Real-time Rendering	Cloud-based	Cloud-based	Unity engine	UE5.5
2D-3D Navigation	Separate views	Separate views	Custom required	Seamless workflow
Licensing Cost	Commercial	Commercial	Open-source (€0)	Open-source (€0)
Customization	Limited API	SDK available	Full (Unity)	Full (Blueprint/C++)
Target Users	Enterprise FM	Infrastructure	Developers	FM / Education

Commercial platforms require significant financial investment and constrain customization, while cloud-based solutions (Tandem, iTwin) introduce network dependency. The proposed workflow eliminates licensing costs, provides full Blueprint/C++ customization, supports offline operation, and delivers superior rendering quality, particularly relevant for educational institutions and research environments with constrained budgets.

3. System Architecture

The system proposed in this work implements a four-layered modular architecture (Figure 1), where each layer builds upon the one below.

The Physical Environment Layer encompasses the building's 3D model geometry, deployed Li-Cor HOBOT sensor hardware, and the environmental conditions being monitored. The Data Integration Layer manages three data streams: IFC BIM data imported through the Datasmith pipeline, preserving geometry, materials, and metadata; IoT sensor data through authenticated

HTTP communication with the Li-Cor API using a custom C++ component (IoTHttpManager); and external weather data from the OpenWeather API. The Navigation Layer provides floor management through level streaming, room selection via collision-based detection, and teleportation coordination using a two-point calibration coordinate transformation. The User Interface Layer delivers three parallel modalities: mini-map, full-screen interactive map, and VR environment, all sharing the same data structures and logic.



Figure 1: System Architecture Diagram

Three core data structures support the architecture. **S_FloorData** associates each building level with its floor plan image and vertical index. **S_RoomInfo** links room identification with sensor availability flags, calibrated teleportation coordinates, and descriptive metadata. **FSensorData**, implemented in C++ for performance-critical JSON parsing, encapsulates temperature, humidity, and CO2 readings with availability flags and timestamps. This hybrid C++/Blueprint separation enables non-programmers to modify room configurations through visual scripting while maintaining efficient data processing.

4. Implementation

This section details the implementation of the proposed system using a four-story building as a case study. The building comprises four levels with active environmental monitoring through Li-Cor HOB0 sensors deployed across classrooms and offices.

4.1 BIM Model Processing

The building model was originally created in ArchiCAD 24 and exported as IFC 2.3. The import process encountered several practical challenges representative of real-world BIM-to-game engine integration workflows.

The IFC 2.3 schema caused metadata loss during initial processing attempts. Direct re-export through ArchiCAD 28 failed due to missing library dependencies between software versions, resulting in geometric element loss. The solution employed Blender Bonsai, an open-source IFC authoring platform, to stabilize and upgrade the model prior to Unreal Engine import. While this intermediate step deviated from the ideal direct pipeline, it produced a fully functional IFC model compatible with the Datsmith importer.

The Datsmith pipeline processed the four-level building in approximately one minute, preserving architectural geometry, material assignments, and IFC metadata, including element classification references, ownership history, and schema traceability. Post-import optimization involved selective removal of high-polygon vegetation geometry that contributed disproportionately to rendering overhead without value for facility management. This targeted optimization achieved a stable 120 FPS performance while preserving all architecturally relevant elements.

A critical limitation of the Datsmith importer was the absence of IFC spatial entity support; IfcSpace and IfcRoom objects were not imported, requiring an alternative approach to room identification. The solution employed manually positioned collision box volumes within each room's geometric boundaries, with each volume containing metadata for room identification, sensor logger mapping, and teleportation coordinates. This collision-based system provides automatic room detection through OnOverlapBegin/End events when users navigate through the building.

The building was organized using Unreal Engine's level streaming system across four levels: Klet_2 (basement), Pritličje_2 (ground floor), Nadstropje_2 (second floor with primary monitored spaces), and Nadstropje_3 (third floor). This organization enables selective level loading based on user position, optimizing memory utilization during navigation.

4.2 IoT Sensor Integration

The IoT integration connects the digital twin with real-time environmental data from the Li-Cor HOBOLink sensor network deployed within the building. The implementation employs a custom C++ component (IoTHttpManager) that handles the complete data pipeline from authenticated API requests to structured data delivery.

The IoTHttpManager class extends UActorComponent and provides two key interfaces: a BlueprintCallable function for initiating HTTP requests with bearer token authentication, and a BlueprintAssignable delegate that broadcasts parsed sensor data to the Blueprint logic layer. This hybrid C++/Blueprint approach was adopted after Blueprint-native JSON parsing proved insufficient for the nested data structures returned by the Li-Cor API.

The sensor data pipeline operates as follows. When a user enters a monitored room, the collision detection system identifies the room and retrieves the associated sensor logger ID. The system generates an authenticated HTTPS request to the Li-Cor API with UTC timezone handling for accurate temporal data alignment. The C++ layer parses the JSON response and populates an FSensorData structure containing temperature (°C), humidity (%), and CO2 (ppm) values alongside availability flags for each sensor type. This structure is then broadcast to the Blueprint system, which updates the user interface within sub-100ms response times.

To support building-wide overview, the system pre-loads sensor data for all monitored rooms at initialization, enabling instant retrieval with visual differentiation between live and cached readings.

4.3 Navigation and Interaction Implementation

The multi-modal navigation system was implemented through three interconnected UMG widget systems sharing the same underlying data structures.

The mini-map (WBP_MiniMap) provides continuous spatial orientation at the bottom-center of the viewport. Real-time player tracking transforms 3D world positions to 2D floor plan coordinates using linear interpolation between two calibrated building-corner reference points. Floor-level detection uses Z-coordinate thresholds to automatically switch floor plan displays across all four building levels.

The full-screen interactive map (WBP_FullScreenMap) provides the central facility management interface. A notable technical challenge involved unreliable mouse interaction with UMG Image widgets in UE5.5, resolved by overlaying invisible Button components over room areas with hover and pressed visual feedback. The floor navigation dropdown dynamically manages room button visibility based on FloorLevel properties, displaying only relevant elements for the selected building level.

The teleportation system enables direct 3D navigation from the map interface. Upon room selection, the TeleportToRoom function retrieves calibrated coordinates from the S_RoomInfo structure and executes instant player positioning. Teleportation coordinates were manually calibrated for each room to ensure optimal positioning with clear sightlines, achieving 0.3m average accuracy against a 1m target requirement.

4.4 VR Adaptation

The VR implementation extends the desktop system to Meta Quest hardware through Widget Component conversion and rendering pipeline optimization, maintaining identical functionality across both platforms.

Desktop UMG widgets were converted to 3D Widget Components attached to the VR Pawn, configured with World widget space, 800×600 pixel draw size at 0.3x scale, positioned 2 meters from the user's viewpoint. Controller-based interaction replaces mouse input through line trace methodology on the Visibility channel, with trigger press events converted to mouse click simulations on detected Widget Components. Widgets implement billboard behavior through Event Tick look-at rotation toward the VR camera position.

Rendering optimization was essential for achieving VR comfort standards. The system transitioned from deferred to Forward Shading, disabled hardware ray tracing and Lumen global illumination, reduced anti-aliasing from 4x to 2x MSAA, and enabled fixed foveated rendering. These modifications improved frame rates from 65–73 FPS to consistent 90–120 FPS on an Intel i9-13980HX / RTX 4070 mobile GPU configuration, meeting the 90 FPS minimum for comfortable VR experiences.

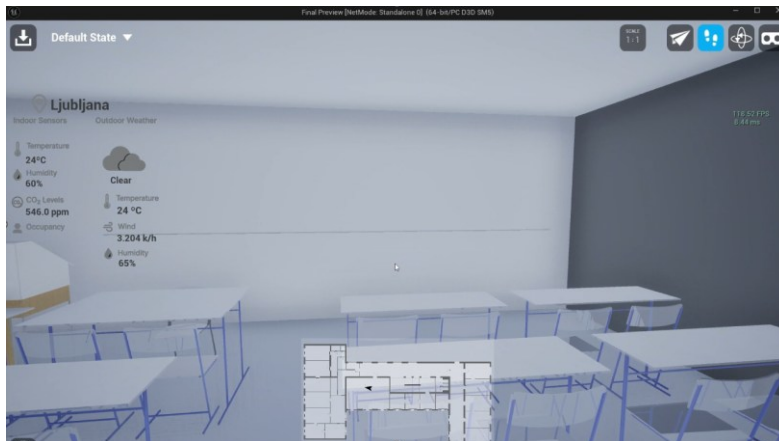


Figure 2: Implementation

5. Results and Validation

The implemented system was validated through comprehensive performance testing across desktop and VR platforms using the four-level university building as the operational test environment. Validation addressed four dimensions: technical performance, navigation efficiency, IoT data integration, and cross-platform reliability.

All performance targets were met or exceeded. Desktop rendering achieved 120 FPS, double the 60 FPS target, providing substantial headroom for future feature additions. VR performance consistently maintained the 90 FPS minimum required for comfortable immersive experiences, with frame times between 8–11 ms after systematic rendering pipeline optimization reduced GPU utilization from 95% to 75%. The navigation system demonstrated significant efficiency gains over traditional building exploration methods. The average room-to-room navigation time of 15 seconds through the map-and-teleport workflow represents a 75% reduction compared to the 60-second target based on manual building traversal estimates.

Interface responsiveness consistently achieved sub-100ms response times for all user interactions, including floor switching, room selection, sensor data display, and teleportation execution. This performance level ensures that the system feels immediately responsive, a critical factor for user adoption in facility management contexts where operators expect real-time feedback. The sensor data integration was validated across both instrumented rooms with varying sensor configurations:

Lecture Hall 201 (full sensor suite): Temperature, humidity, and CO2 readings were successfully retrieved, parsed, and displayed in real-time upon room entry. The C++ JSON parsing pipeline processed Li-Cor API responses within sub-100ms, with sensor availability flags correctly reflecting the complete sensor deployment. CO2 readings provided additional occupancy indication value beyond direct environmental monitoring.

Office 205 (partial sensor suite): The system correctly identified the absence of CO2 and occupancy sensors through availability flags, displaying only temperature and humidity data while clearly indicating unavailable sensor types through visual indicators. This graceful degradation demonstrates the system's adaptability to heterogeneous sensor deployments, a practical requirement for real building environments where sensor coverage varies between spaces.

6. Discussion

The results demonstrate that professional-grade digital twin capabilities can be delivered at zero licensing cost, challenging the assumption that commercial platforms are required. The hybrid C++/Blueprint architecture proved essential: complex JSON parsing runs in C++ for performance, while interface logic and room configuration remain in Blueprint visual scripting, enabling non-programmers to maintain and extend the system.

This work directly addresses the Unreal Engine research gap in AEC. Within Deng et al.'s [1] taxonomy, the system operates at Level 3 (BIM + IoT), with the modular architecture providing a foundation for Level 4 advancement through future ML integration. The workflow is relevant for Portugal's BIM transition under Decreto-Lei 10/2024 [2], offering institutions a practical entry point for digital twin adoption.

Several limitations should be acknowledged. The current implementation covers two instrumented rooms; scalability to full building coverage requires validation. The absence of Datasmith support for IFC spatial entities necessitated manual collision box placement — a labor-intensive step scaling linearly with room count. User experience evaluation involved a limited participant group; broader studies with diverse facility management professionals would strengthen validation. The IFC 2.3 source model introduced interoperability challenges; validation with native IFC 4.3 models would confirm compatibility with current standards [7].

7. Conclusions and Future Work

This paper presents an open-source workflow for developing interactive digital twins in Unreal Engine 5.5, integrating BIM data, real-time IoT sensor monitoring, and immersive VR navigation. The system was validated on a four-level faculty building at the University of Ljubljana, demonstrating that game engine technology can deliver professional-grade facility management capabilities without licensing costs.

The research yields four principal contributions. First, a seamless multi-modal navigation workflow bridging 2D floor plan interfaces with 3D immersive exploration through coordinate transformation and teleportation, achieving 0.3-meter accuracy — with no documented equivalent in current AEC literature. Second, a real-time IoT integration pipeline using a hybrid C++/Blueprint architecture that connects Li-Cor HOBO sensors with spatial building visualization, achieving sub-100ms interface responsiveness. Third, a cross-platform implementation supporting desktop (120 FPS) and VR (90–120 FPS) operation from a unified codebase. Fourth, the first comprehensive Unreal Engine 5.5 implementation for digital twins in AEC, establishing performance baselines for future comparative research.

Future work should address the identified limitations through several development directions. Expanding room coverage across all building levels with automated room boundary extraction from IFC geometry would reduce the manual configuration overhead currently required for collision-based room detection. Validation with native IFC 4.3 models would confirm compatibility with current buildingSMART standards and strengthen the workflow's applicability to new construction projects. Integration of BCF (BIM Collaboration Format) would enable issue tracking and collaborative facility management workflows, connecting the digital twin with established BIM coordination platforms such as BIMCollab. The modular

architecture also supports future advancement toward predictive facility management through integration of machine learning analytics for sensor data pattern recognition and anomaly detection.

Acknowledgement

This study was partly supported by the University of Ljubljana for Sustainable Society ULTRA Pilot 4 Sustainable Environment. The project was co-financed by the Republic of Slovenia, the Ministry of Higher Education, Science and Innovation, and the EU - NextGenerationEU. The investment is part of the plan's measures, which are financed by the Recovery and Resilience Facility. Special thanks to Dr. Andrej Vidmar for his IoT expertise and professional guidance.

References

- [1] M. Deng and C. Menassa, "From BIM to digital twins: A systematic review of the evolution of intelligent building representations in the AEC-FM industry," *J. Inf. Technol. Constr.*, vol. 26, pp. 58–83, Feb. 2021, doi: 10.36680/j.itcon.2021.005.
- [2] Presidency of the Council of Ministers, "Decree-Law 10/2024." Presidency of the Council of Ministers. [Online]. Available: <https://diariodarepublica.pt/dr/detalhe/decreto-lei/10-2024-836222484>
- [3] W. Thabet, M. F. Shiratuddin, and D. Bowman, "Virtual Reality in Construction," 2002, pp. 25–52.
- [4] J. Zhao, H. Feng, Q. Chen, and B. Garcia De Soto, "Developing a conceptual framework for the application of digital twin technologies to revamp building operation and maintenance processes," *J. Build. Eng.*, vol. 49, p. 104028, May 2022, doi: 10.1016/j.job.2022.104028.
- [5] M. Signorini and A. Pomè, "Shaping the future of facility management. Market and literature insights on digital twin adoption," *Facilities*, Apr. 2025, doi: 10.1108/F-02-2025-0029.
- [6] H. Hosamo, A. Imran, J. Cardenas-Cartagena, P. Svennevig, K. Svidt, and H. Nielsen, "A Review of the Digital Twin Technology in the AEC-FM Industry," *Adv. Civ. Eng.*, vol. 2022, pp. 1–17, Mar. 2022, doi: 10.1155/2022/2185170.
- [7] A. Revolti, L. Gualtieri, P. Pauwels, and P. Dallasega, "From building information modeling to construction digital twin: a conceptual framework," *Prod. Manuf. Res.*, vol. 12, Aug. 2024, doi: 10.1080/21693277.2024.2387679.
- [8] K. May, J. Walsh, R. Smith, N. Gu, and B. Thomas, *UnityRev - Bridging the gap between BIM Authoring platforms and Game Engines by creating a Real-Time Bi-directional Exchange of BIM data*. 2022, p. 536. doi: 10.52842/conf.caadria.2022.2.527.
- [9] Autodesk, "Autodesk Tandem," 2024. [Online]. Available: <https://www.autodesk.com/products/tandem>
- [10] Bentley Systems, "iTwin Platform," 2024. [Online]. Available: <https://www.bentley.com/software/itwin-platform>
- [11] Unity Technologies, "Unity Reflect," 2024. [Online]. Available: <https://unity.com/products/unity-reflect>