www.itcon.org - Journal of Information Technology in Construction - ISSN 1874-4753

DIGITAL TWINS USING NVIDIA OMNIVERSE AND OPENUSD: A PROOF-OF-CONCEPT IMPLEMENTATION

SUBMITTED: April 2026
PUBLISHED: August 2026
EDITOR: Robert Amor
DOI: 10.36680/j.itcon.2026.035

Chady Elias, PhD
Rinker School of Construction Management, University of Florida, Gainesville, FL, USA
chadyelias@ufl.edu

Raja R. A. Issa, Distinguished Professor
Rinker School of Construction Management, University of Florida, Gainesville, FL, USA
https://orcid.org/0000-0001-5193-3802
raymond-issa@ufl.edu

SUMMARY*: Operational digital twins of buildings require a common environment in which Building Information Modeling (BIM)-based spatial models, live telemetry from building automation systems (BAS), and institutional records of equipment and spaces are integrated to support monitoring, analysis, and decision-making at facility scale. Existing platforms and research efforts still leave open challenges in integrating heterogeneous source systems, representing building semantics within the integrated environment, and supporting analytical workflows over the composed information at runtime. This paper investigates NVIDIA Omniverse and OpenUSD as a scene-native foundation for addressing these challenges through a shared representation. The paper contributes (i) a modular five-layer reference architecture for digital twin development, (ii) a prototype implementation realized as a custom Omniverse Kit application integrating BIM-derived context, BAS telemetry, and institutional facility records in an academic building, and (iii) an analysis of how native OpenUSD constructs can encode operational building context directly within the scene. The case study demonstrates that the composed scene, continuously updated with live operational data, can support historical trend review and system-aware workflows grounded in current building context. The results show that OpenUSD can host operationally relevant building semantics alongside spatial and operational content within a single scene, positioning Omniverse as an extensible foundation for the analytical, AI-assisted, and lifecycle-spanning workflows expected of next-generation facility operation.*

KEYWORDS*: digital twins, semantic modeling, NVIDIA Omniverse OpenUSD, facility management, smart buildings.*

REFERENCE*: Elias, C., & Issa, R., R., A. (2026). Digital twins using NVIDIA Omniverse and OpenUSD: A proof-of-concept implementation. Journal of Information Technology in Construction (ITcon), 31, 813-834. https://doi.org/10.36680/j.itcon.2026.035*

1. INTRODUCTION

Digital twin (DT) technology is reshaping how buildings are monitored and managed by enabling live virtual representations of physical assets that support observation, analysis, and decision-making across the facility lifecycle (Elias et al., 2025; El Mokhtari et al., 2022). Realizing such systems at portfolio scale requires an integration environment that brings together BIM-derived spatial representations, live sensor telemetry from BAS, and institutional records from facility and asset management (FAM) systems. These sources must also be aligned to a common data model, so that information is contextualized with the equipment and spaces it describes rather than merely aggregated. In current practice, answering those questions often still requires moving across BAS dashboards, BIM viewers, and institutional systems while filling gaps through local knowledge and ad hoc reconciliation of inconsistent records (Pishdad-Bozorgi et al., 2018).

NVIDIA Omniverse is a platform for developing three-dimensional applications and simulations built on the Universal Scene Description standard, known as OpenUSD, and supports the composition of heterogeneous content into a shared scene (NVIDIA Corporation, 2025; Wandersman, 2023; Tang et al., 2025). In this environment, BIM models can be converted into USD form through dedicated connectors while preserving element identifiers and parameter values from the source environment (McKenna et al., 2024; Saldana Ochoa et al., 2025). Omniverse applications can then run simulations, connect data from multiple enterprise systems into a unified view, and host domain-specific logic on top of the same scene, yielding an interactive environment that brings spatial, operational, and descriptive context together. These capabilities have been demonstrated in manufacturing and urban DT deployments (Bhatia et al., 2025; Wajid et al., 2024), and in exploratory AECO-adjacent work adapting BIM assets for IoT-enabled twins and routing telemetry into three-dimensional dashboards (Wandersman, 2023; McKenna et al., 2024). This body of work positions Omniverse and OpenUSD as relevant candidates for DT development.

Despite this progress, limited research has examined how Omniverse and OpenUSD can be configured as an integration environment for operational building DTs. Existing work has demonstrated relevant capabilities, including BIM–IoT integration, semantic building models, interactive three-dimensional interfaces, and building analytics. However, less attention has been given to how spatial entities, current operating data, institutional records, and explicit system relationships can be composed within a shared scene representation and made directly available to modular application logic at runtime. This integration is important because analytical and diagnostic workflows depend not only on access to sensor values, but also on reliable ways to identify the equipment involved, associate data with the spaces and systems it describes, and traverse the relationships needed to interpret operational conditions.

The objective of this study is to develop and examine a scene-based architecture for operational building DTs on NVIDIA Omniverse and OpenUSD. Two research questions guide the work. RQ1: How can BIM-derived spatial models, live BAS telemetry, institutional facility records, and system relationships be integrated and maintained as a coordinated operational representation within an Omniverse application? RQ2: To what extent can native OpenUSD constructs represent equipment classifications, HVAC system relationships, and contextual building information in a machine-readable form that can be used by monitoring and analytical workflows at runtime?

The paper addresses these questions through three contributions: (i) a modular five-layer reference architecture separating source interfaces, data management, semantic context, application logic, and service delivery; (ii) a proof-of-concept implementation comprising cooperating Kit extensions and supporting services deployed for an operational academic building; and (iii) an examination of how native OpenUSD constructs can carry building-specific operational context directly within the scene.

2. BACKGROUND AND RELATED WORK

2.1 Modular digital twin architectures in AECO and FAM

Recent reviews of DT work in architecture, engineering, construction, and operations (AECO) and facility and asset management (FAM) contexts indicate that many deployments remain application-specific and still lack stable integration frameworks across the facility lifecycle (Elias & Issa, 2025; Gu & Jazizadeh, 2024; Almadhor et al., 2025). This has motivated greater attention to DT system architectures that make core functions explicit and keep them manageable over time. For instance, Nguyen et al. (2025) introduced a three-layer DT framework for the facility and asset management of modular buildings, comprising a physical layer for real-time data capture and connectivity, a digital layer for BIM and Geographic Information System (GIS) models together with data integration and analytics, and a service layer for monitoring, control, and decision support. Eneyew et al. (2022) proposed a multi-layer BIM-IoT architecture for smart-building DTs consisting of physical, data storage, data access, data integration, and application layers. Asare et al. (2024) similarly developed a three-layer reference architecture for DT-based predictive maintenance, with distinct information, application, and visualization layers.

Collectively, these studies suggest a common architectural pattern in which acquisition, data handling, integration, analytics, and user-facing application concerns are organized as distinct parts of the DT. Such decomposition is significant because DTs at building scale must coordinate complex BIM environments, telemetry services, and institutional systems that originate in different software ecosystems and serve distinct operational demands. Separating these concerns into distinct architectural layers therefore provides a practical basis for interoperability, extensibility, and long-term operation of DTs.

DT implementations can also be situated along capability progressions that extend from visual representation and current-state monitoring to contextual analytics, simulation, and autonomous operation. As illustrated in Figure 1, KPMG (2022) describes a progression from Existence and Status Twins to Operational, Simulation, and Cognitive Twins, while Autodesk (2025) presents a comparable progression from descriptive and informative twins to predictive, comprehensive, and autonomous twins (Elias & Issa, 2023). Across these frameworks, the development of advanced operational capabilities begins with reliable physical-to-digital synchronization and an information infrastructure that preserves the identity, context, and relationships associated with incoming building data. These foundations allow current and historical observations to be interpreted in relation to the relevant assets, spaces, and systems, thereby providing the structured context needed to support analytics, simulation, automation, and AI-assisted workflows. The present study therefore concentrates on the integration, synchronization, and contextual representation required to establish this operational foundation.

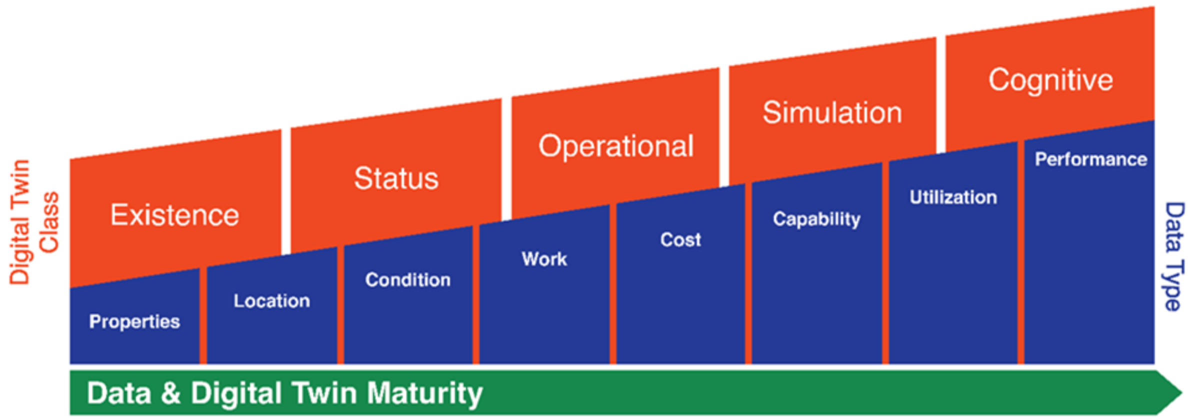

Figure 1: Digital twin maturity levels (Adapted from KPMG 2022).

2.2 Semantic Modeling for Building Information

Semantic modeling is central to DT development because DTs depend on representations that can organize heterogeneous information sources, preserve the meaning of entities and relationships across systems, and support machine-readable querying, inference, and application interoperability. Karabulut et al. (2024) situate ontologies and knowledge graphs as important mechanisms for interoperability, knowledge representation, and automated reasoning in DT research. In the built environment specifically, Ramonell et al. (2023) showed that a knowledge graph can serve as the integration backbone for DTs of built assets by linking BIM, IoT metadata, and process information in a modular and interoperable architecture. For building operations, semantic models therefore need to express both entity context, which establishes what something is and what role it plays, and relational context, which establishes how entities connect through composition, containment, distribution topology, and logical dependency. This structured context is essential because analytic, diagnostic, and decision-support applications depend on consistent ways to identify building elements, associate operational data with the equipment and spaces it describes, and navigate relationships between building systems.

Research and industry practice have responded to this need through several complementary standards. Project Haystack uses descriptive tags to annotate equipment, points, and spaces with type and role information (Project Haystack, 2018). Brick provides an ontology expressed in the Resource Description Framework with a class hierarchy for equipment, points, and spaces, together with a fixed relationship vocabulary (Balaji et al., 2016). ASHRAE Standard 223P specifies a formal semantic model oriented toward building systems and control relationships (ASHRAE, 2018), while RealEstateCore offers a related ontology-driven approach for real estate portfolios. Practical experience with these standards indicates that creating and maintaining semantic models in production environments remains labor-intensive and requires specialized expertise and tooling (Pritoni et al., 2021; Mulayim et al., 2024). Collectively, the literature establishes both why semantic structure is essential and why encoding it comprehensively remains an active implementation challenge in DT development.

2.3 Data storage strategies in digital twins

A core architectural question in DT development is how different categories of building data should be managed so that the DT can support current-state inspection, historical analysis, and retrieval of operational context. BAS telemetry, BIM-derived building context, and facility records impose different storage and query requirements. Telemetry arrives as dense, time-indexed measurements, whereas building semantics such as spaces, equipment, relationships, and asset metadata are updated far less frequently and are typically queried as structured descriptive information.

One option is to maintain these data families in a single graph or ontology-oriented database so that temporal and relational context can be queried in one place. In practice, this becomes difficult when high-frequency BAS telemetry is represented directly as graph nodes or edges, because ingestion volume grows rapidly and temporal queries become costly (Khan et al., 2025; Khan & Bartolini, 2025). Khan et al. (2025) reported that one month of telemetry stored as Resource Description Framework (RDF) triples required more than seven hundred times the storage of its NoSQL equivalent. Similarly, Ammar et al. (2025) note that graph-oriented systems often store telemetry as node attributes or linked time-series references rather than representing every measurement as a first-class temporal entity.

Digital twin implementations often rely on multiple data platforms; each aligned with a different category of information and query workload. Purpose-built time-series databases are therefore commonly adopted for continuous BAS and IoT measurements, while building context, asset metadata, and facility records remain in the enterprise information systems that manage them. Geometry and BIM-derived properties are typically carried through model-based building representations and exchange standards such as Industry Foundation Classes (IFC) (ISO, 2024), with additional records retrieved from the systems that maintain them. This separation aligns with the broader distinction between systems centered on design and asset information management and those centered on building automation (Biagini et al., 2024).

2.4 NVIDIA Omniverse and OpenUSD in digital twin development

NVIDIA Omniverse has become a prominent platform for integrated digital twins across various sectors. As a modular three-dimensional integration and simulation environment built on OpenUSD, it supports the composition of heterogeneous models and data sources into a shared interactive scene (Wandersman, 2023; Vatanen, 2024). In manufacturing, Omniverse has been used for industrial DT workflows, and Bhatia et al. (2025) integrated design, simulation, and operational visualization for fusion power plant DTs. On the urban scale, the TEC District work combined BIM and GIS geometry, IoT telemetry, and knowledge graph outputs within Omniverse for visualization and monitoring (Wajid et al., 2024).

In AECO contexts, BIM models can be synchronized with Omniverse through Revit connectors and IFC-based workflows (NVIDIA Corporation, 2025), transforming static design models into interactive environments that can support facility and asset management, predictive maintenance, and energy analysis (Archi Communications Team, 2025; Simio, n.d.). OpenUSD has also evolved beyond its origins in visual effects through custom schemas and AECO-oriented extensions, including efforts to encode IFC content in USD and to formalize related interoperability practices (Corke, 2022; Alliance for OpenUSD, n.d.). Omniverse's extensible application architecture further allows project teams to connect enterprise systems, add domain-specific logic, and use the shared scene as a programmable integration surface for visualization, simulation, and operational workflows (Colombo et al., 2023).

These capabilities position Omniverse as a programmable environment for DT development. Prior AECO-adjacent work has explored converting BIM content for IoT-enabled applications, routing telemetry into three-dimensional dashboards, and combining urban-scale geometry with operational information in Omniverse (Wandersman, 2023; McKenna et al., 2024; Wajid et al., 2024). This work demonstrates the capacity of OpenUSD-based environments to bring model-based and operational information into interactive scene representations. The present study extends this line of work by examining how BIM-derived entities, live telemetry, institutional records, system relationships, and application logic can be composed and coordinated directly within the scene to support facility-scale operational workflows.

2.5 Related work and study context

Related building-DT research has demonstrated complementary approaches to integrating building information and supporting operational applications. Lu et al. (2020a) developed a DT demonstrator for the West Cambridge Campus that connected IFC-based building models with building management, asset management, and space management systems, real-time sensors, and QR-based maintenance feedback. Associations between BIM object GUIDs and corresponding asset or space identifiers were maintained through matching tables in DynamoDB, enabling external records to be retrieved for selected model objects. The resulting environment supported pump anomaly detection, ambient-condition monitoring, maintenance optimization, and repair prioritization. Arsiwala et al. (2023) developed an Azure-based residential DT for indoor-air-quality monitoring and eCO_2 prediction. RealEstateCore-aligned DTDL models represented spaces, sensors, capabilities, and their relationships in Azure Digital Twins; device identifiers connected incoming IoT observations to the corresponding twin instances, which were subsequently linked to a three-dimensional scene for live monitoring and alert visualization.

Semantic integration has also been used to structure operational analytical processes. Merino et al. (2023) federated IFC, Brick, and a BOT-informed platform data model through modular pipelines that transformed reference information into an indexed in-memory representation and exposed the integrated context through APIs and real-time subscriptions. In the Alan Reece Building case, an HVAC fault-detection and diagnostic workflow used this representation to identify relevant sensors, determine their associated rooms and equipment, traverse HVAC functional relationships, retrieve additional telemetry, and progressively investigate potential fault sources. Detected faults were reported by email and visualized against affected assets and spaces in the building model. Xie et al. (2023) similarly demonstrated how machine-readable fault tags could be assigned to sensor entities and appended to real-time streams, enabling an FDD process to automatically identify and retrieve the data dimensions relevant to a particular fault.

Ma et al. (2025) extended this line of research by implementing a DT in Unity for Microsoft HoloLens and demonstrating it on a typical floor of a high-rise office building. The implementation combined BIM, BAS, IoT, and cloud-based analytics to support equipment visualization, indoor navigation, environmental monitoring, live AHU and VAV inspection, VAV setpoint adjustment, and AI-supported thermal control. Integration within Unity relied on component names retained during BIM-to-FBX conversion to associate sensor points with model components. Time-series data, static BIM attributes, and analytical outputs remained managed in separate cloud databases and were accessed using application scripts.

Collectively, these studies demonstrate substantial progress in cross-system building-data integration, semantic modeling, operational analytics, and interactive three-dimensional applications. However, these capabilities are commonly demonstrated through separate or application-specific implementations. Some studies achieve broad source-system integration through maintained identifier mappings, while others apply richer semantic representations to narrowly scoped analytical use cases or connect operational data to BIM interfaces through project-specific component associations. Consequently, limited evidence is available on how heterogeneous operational data, institutional records, equipment classifications, and explicit system relationships can be integrated as a coherent and extensible information foundation that supports a broad range of day-to-day facility-management activities, from monitoring and investigation to decision support and operational coordination. The present study addresses this integration gap through a modular Omniverse and OpenUSD implementation in which these forms of operational context are coordinated around common building entities and made available to multiple application services.

3. SYSTEM ARCHITECTURE

This section presents a five-layer reference architecture for building DTs on NVIDIA Omniverse and OpenUSD. The architecture organizes the source systems, data-management functions, semantic structures, application logic, and user-facing services needed to coordinate spatial, operational, and institutional information within a common environment. It emphasizes interoperability across heterogeneous sources, separation between telemetry handling and institutional-data access, and explicit representation of building context within the OpenUSD scene. Figure 2 provides an overview of the architecture and the relationships among its five layers. The layers are then described sequentially in Sections 3.1 through 3.5: source systems and data interfaces in Section 3.1, data management in Section 3.2, semantic and system context in Section 3.3, application logic and workflow services in Section 3.4, and service delivery and interaction in Section 3.5.

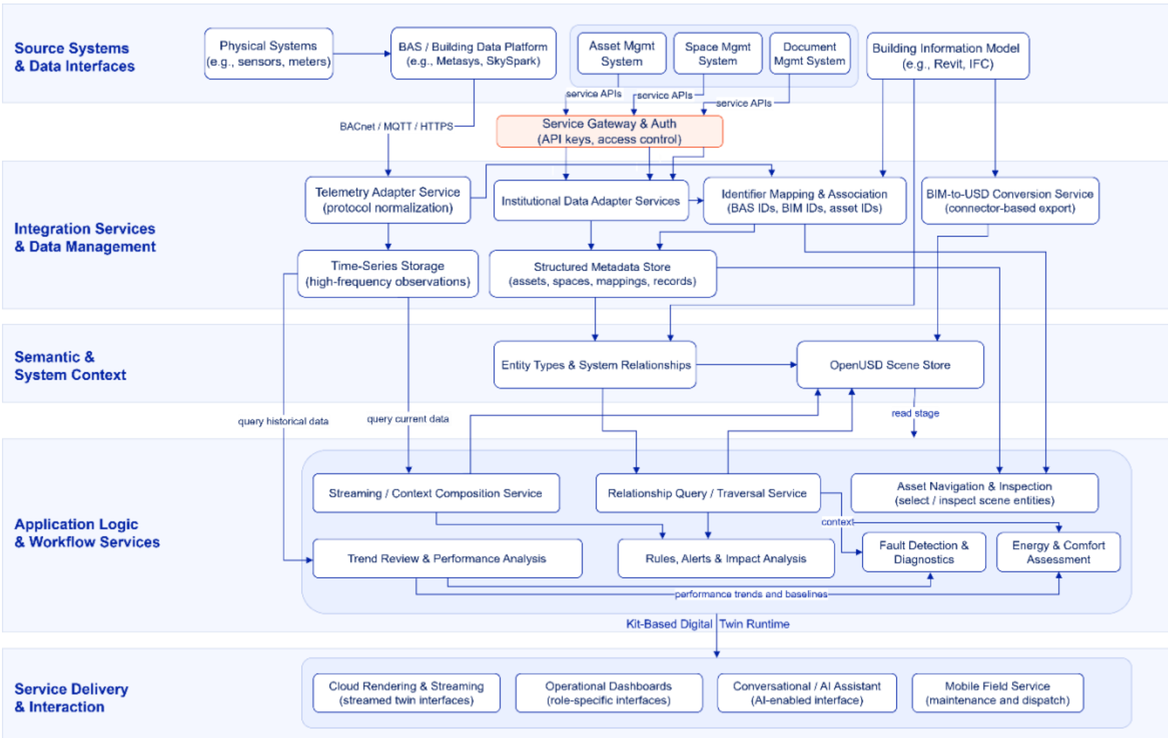

Figure 2: Reference architecture for building digital twins on NVIDIA Omniverse.

3.1 Source systems and data interfaces

The first layer of the architecture defines the upstream systems that supply spatial, operational, and institutional information to the DT environment. Typical sources include building automation systems, Internet of Things (IoT) telemetry infrastructure, facility and asset management platforms, energy management systems (EMS), and BIM environments. Each source contributes a different form of information. BAS and IoT environments provide time-varying measurements and control values. FAM platforms provide asset, maintenance, and space records. BIM environments provide spatial and equipment representations that anchor the three-dimensional scene.

These source systems expose information through interfaces with different structures, update patterns, and operational roles. Telemetry is delivered through controls and IoT integration pathways specific to the building automation environment. Institutional records from FAM platforms are commonly queried via structured interfaces for asset, maintenance, and space information and are refreshed less frequently than BAS telemetry. BIM information is brought into Omniverse by converting source BIM models into USD through connector-based workflows that preserve geometry and selected metadata in a form the runtime can use. The design accommodates this variation in how sources expose information, rather than assuming a single upstream interface or exchange pattern.

Converting BIM models into USD establishes the spatial basis of the scene. During this process, element identifiers, family and type information, instance metadata, and selected source-model properties can be preserved on the corresponding scene entities. These retained fields provide stable anchors for associating telemetry, institutional records, and semantic relationships at later stages.

3.2 Data management

Intermediary services translate, coordinate, store, and link source data before it is used within the DT. Whereas the source-interface layer identifies the systems that provide telemetry, records, and spatial data, this stage addresses how those inputs are prepared for consistent use by later components of the architecture. Its role is to reconcile heterogeneous upstream systems with the shared environment in which semantic, analytical, and user-facing functions operate.

Telemetry from BAS and IoT sources is handled through ingestion services that translate protocol- and schema-specific messages into a consistent internal representation and persist them in storage suited to time-series workloads. This supports frequent updates, high ingestion volumes, and efficient retrieval of current and historical observations, while keeping telemetry acquisition separate from the user-facing tools that later consume these data. Asset, maintenance, and space records are retrieved through controlled service interfaces that enforce authentication and access rules, transform source-specific fields into structured metadata, and make those records available alongside telemetry and scene-based context.

Different classes of building information are therefore managed according to their distinct operational characteristics. It also supports the identifier mappings needed to associate telemetry points, BIM elements, and institutional asset records that refer to the same underlying equipment, so that later layers can work from a coordinated building context rather than from disconnected source records.

3.3 Semantic and system context

A DT must represent more than the geometry of a building and the data attached to individual assets. To support interpretation, it must also capture the semantic and system context that defines what each entity is, how it relates to other entities, and what role it plays within the broader built environment. The semantic layer provides this contextual grounding by organizing assets, spaces, systems, and their relationships into a structured representation that supports interpretation and analysis beyond isolated objects or sensor values.

In Omniverse, this context can be maintained directly on the same OpenUSD scene that carries the spatial model and the composed operational information. OpenUSD organizes a scene as a composed hierarchy of prims with typed attributes (OpenUSD, 2026a) and explicit relationships between scene entities (OpenUSD, 2026b). This allows spatial representation, equipment context, and system relationships to be expressed within a single digital environment in which they can be inspected, queried, and used by application logic. Representing this structure directly on the scene makes relevant building relationships explicit in a machine-readable form. Equipment can therefore be modeled in terms of categories, connected systems, service relationships, and distribution paths, with that context supporting later diagnostics, comparison, and decision support workflows. Semantic modeling standards such as Project Haystack (Project Haystack, 2018), Brick Schema (Balaji et al., 2016), and ASHRAE Standard 223P (ASHRAE, 2018) remain relevant as domain references for equipment classes, relationship semantics, and validation expectations. Where projects already rely on such standards, they can inform how categories, system links, and rule-based checks are represented within the DT. Semantic context supports the interpretation of system behavior while accommodating project-specific semantic conventions.

3.4 Application logic and workflow services

Once spatial, operational, institutional, and semantic information has been assembled within the DT, the next architectural concern is how that combined information is used to perform analysis and support operational workflows. The application layer hosts the logic and workflow services that operate within Omniverse. Its purpose is to translate integrated building information into analytical and decision-support functions, including energy performance assessment, fault detection and diagnostics, impact tracing, alert evaluation, and other forms of scenario-based interpretation grounded in equipment condition, system dependencies, and current operating state.

Within Omniverse, application logic is hosted in a Kit-based environment that supports the modular composition of functionality (NVIDIA, 2026a). Workflow-specific capabilities can be implemented as extensions aligned with particular analytical and operational functions (NVIDIA, 2026b), while drawing on the same shared scene and underlying data infrastructure. This structure allows individual services to be developed, modified, or extended with limited changes to other application components, provided that the shared data structures and interfaces through which they access the DT remain consistent. It also enables project-specific concerns to be separated from reusable workflow logic through configuration of target equipment classes, scope definitions, thresholds, and data bindings. The resulting architecture is intended to reduce the amount of application logic that must be redesigned when workflows are introduced, revised, or adapted to additional facilities. However, the extent to which this modularity reduces adaptation effort across heterogeneous buildings remains to be evaluated.

At runtime, when a specific asset is selected, the environment can assemble its spatial location, current and historical operating values, institutional records, equipment classification, and relevant upstream and downstream associations into a unified analytical context. This supports inspection, diagnosis, comparison, and other workflow-specific evaluations while remaining extensible to additional services as operational needs evolve. Operational questions can therefore be examined in relation to the condition of the asset, the behavior of the surrounding system, and the broader state of the building, rather than through data sources interpreted in isolation from one another.

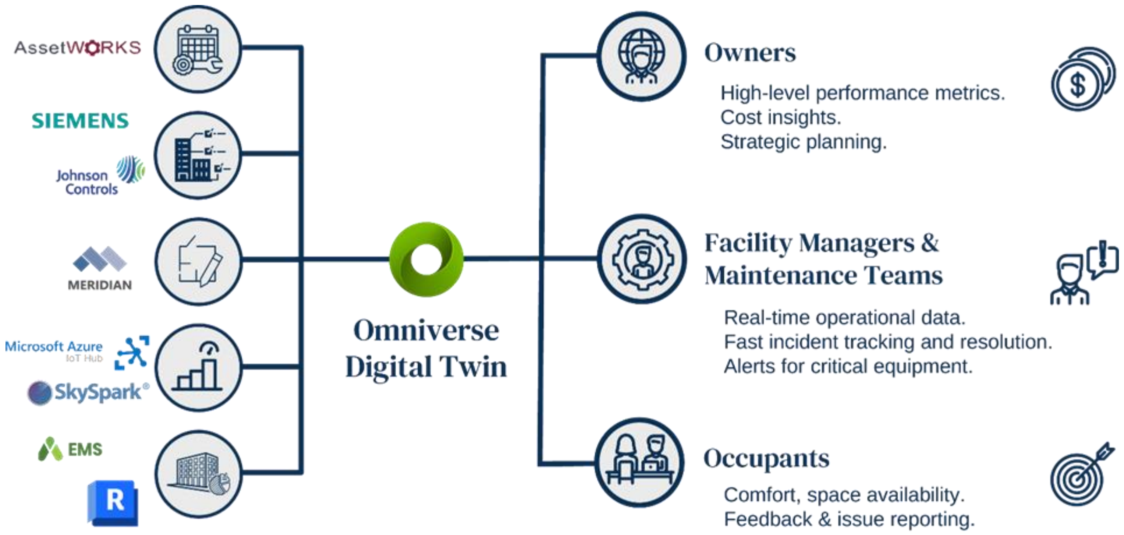

Figure 3: Omniverse digital twin as an integration hub and role-based service environment.

3.5 Service delivery and interaction

The value of the DT depends on how effectively integrated building information can be delivered within the operational settings where decisions are made. In this role, the Omniverse environment provides a shared point of access through which operators, facility managers, field technicians, and other decision-makers can interact with building information at levels of detail appropriate to their tasks, as illustrated in Figure 3.

Within the environment, access to information can be organized across multiple levels of operational scope, from campus- or district-scale oversight to building-, system-, and asset-level investigation. A cloud-rendered scene can support broad situational awareness and navigation across facilities, while coordinated extensions can provide more targeted interfaces for dashboards, alert review, mobile maintenance activities, and tool-augmented language-model interaction. Because these interfaces draw on the same scene and underlying information base, movement from high-level overview to detailed asset interpretation can occur within a shared operational context rather than across disconnected applications and separately maintained views.

Outputs from the integrated environment can also be routed into the enterprise systems used to manage building operations. Diagnostic summaries, alert conditions, and maintenance-relevant findings can be sent to work-order, maintenance-management, and notification systems, either as information for human review or as inputs to automated processes. In this role, the DT serves as a point of translation between heterogeneous building data and the decision-ready outputs needed for operational coordination, maintenance response, and related downstream action.

4. PROOF-OF-CONCEPT IMPLEMENTATION

This section describes the proof-of-concept Kit application developed for the academic facility case study. The implementation assembled BIM-derived building context, BAS telemetry, institutional asset records, room-level space information, and system relationships within Omniverse to support contextualized equipment review, comparative time-series analysis, system-topology navigation, and operational workflow development. The implementation is presented to examine how the proposed architecture can be realized in a real facility setting and how integrated scene composition and native OpenUSD relationships can support operational workflows grounded in this composed context.

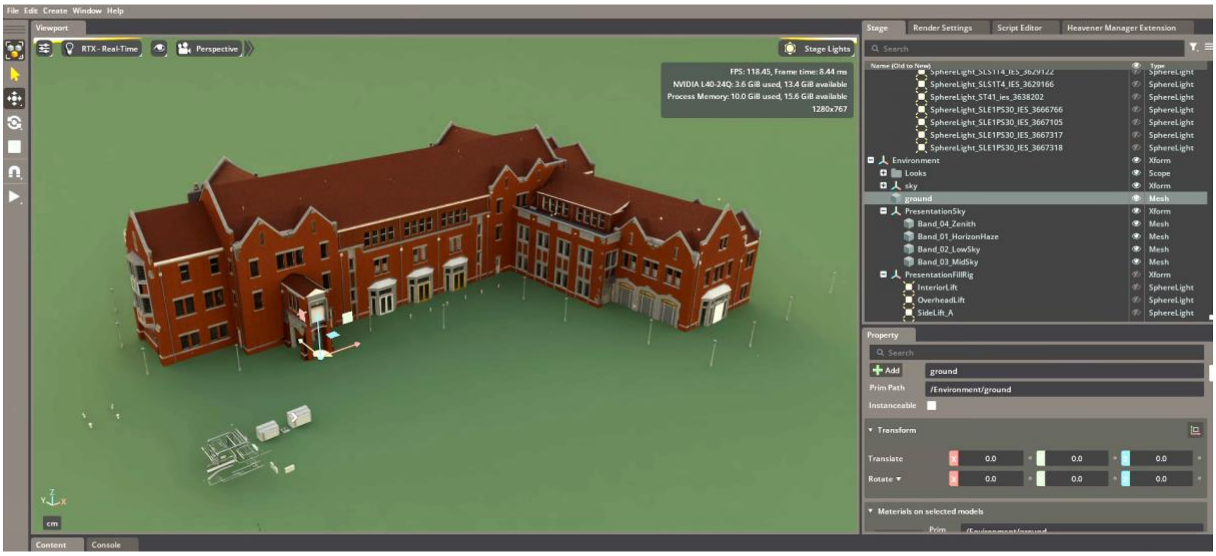

Figure 4: Case-study building digital twin scene.

4.1 Case study context

The case-study building is a three-story academic facility of approximately 57,370 square feet consisting of classrooms, instructional spaces, and offices. It is served by a conventional heating, ventilation, and air-conditioning (HVAC) system comprising multiple air-handling units (AHUs) and 55 BAS-connected terminal units, including variable air volume (VAV) terminals and fan-coil units (FCUs). This building scale and HVAC configuration reflect a common instructional-facility context and provide a practically relevant setting for developing and examining the implementation under routine operating conditions. The full building scene loaded in the Kit runtime is shown in Figure 4.

Four source systems supply the spatial, operational, and institutional information used in the implementation. The spatial basis is the building's Revit model, converted to OpenUSD through NVIDIA's Revit Connector (NVIDIA Corporation, 2025), which preserves geometry, equipment context, source-model identifiers, and related metadata on the scene. Current and historical operating data are provided by SkySpark, the institution's building data analytics platform, which aggregates telemetry from BACnet-connected field controllers and exposes structured point payloads aligned with Project Haystack conventions (Project Haystack, 2018). These records include point descriptions, values, timestamps, and the identifiers needed to associate telemetry with reference equipment. Institutional asset context is retrieved from AssetWorks, the university's facility and asset management platform, which supplies asset tags, names, manufacturers, serial numbers, model numbers, and links to external asset-profile and work-history views. Additional room-level context is retrieved from the university's internal space-management system, which supplies room identifiers, use categories, organizational affiliations, and selected space metrics.

4.2 Kit application and data integration

Omniverse and OpenUSD provided the user-interface, scene-composition, and development environment for the implementation, including the viewport and property interfaces, the Kit extension framework, runtime mechanisms for observing scene-attribute changes, and the OpenUSD prim, attribute, and relationship model. The Revit Connector supported the conversion of BIM content to USD for use within this environment. Building on these capabilities, the implementation developed in this study included the supporting data services, methods for reconciling records across source systems, scene composition logic, equipment classification, system relationships generation and modeling, operational monitoring applications, and the coordinated extensions required to support the implemented building workflows.

The proof of concept was structured as a custom Kit application composed of cooperating extensions with distinct responsibilities. A data-integration extension maintains the managed equipment set and composes telemetry and institutional attributes onto the scene. A semantic-context extension represents equipment categories and directional system relationships directly in OpenUSD. An early-stage monitoring extension provides change detection and threshold-based alert logic, while an application-specific window and orchestration extension coordinates connection management, equipment browsing, relationship visualization, comparative plotting, and monitoring interactions. A shared utilities extension supports application programming interface (API) access, plotting, and graph export. Figure 5 shows the extension layout used in the implementation.

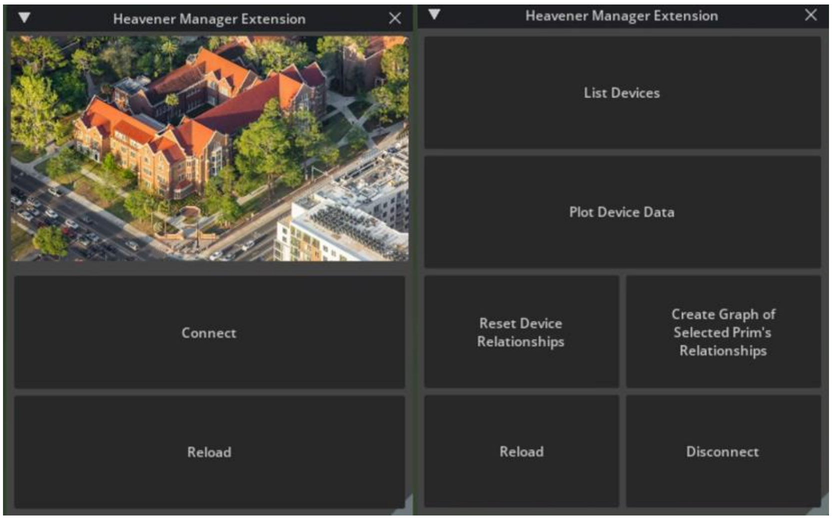

Figure 5: Omniverse kit application extension layout.

The supporting time-series service is implemented in FastAPI and backed by PostgreSQL. At the database level, the service centers on three main tables: building, device, and device_data. The building table groups telemetry by institutional building identifier. The device table stores the BAS-facing device_id together with a server-generated device_uid, preserving the source-system identifier while maintaining a stable internal reference for the service. The device_data table stores timestamped telemetry bundles as JSONB, allowing different equipment types to carry distinct point sets without requiring schema changes to the database. This service supports telemetry ingestion, most-recent retrieval, historical retrieval, and device discovery through dedicated API routes. SkySpark delivers telemetry to a configured HTTPS ingestion endpoint, whereas the Kit application queries separate routes for latest readings, historical time windows, and the building-level device index that populates the managed equipment list. As new telemetry is ingested, the Kit application retrieves the latest equipment payloads and writes the updated values and timestamps to attributes on the corresponding scene entities. This process maintains the physical-to-digital connection used by the prototype and ensures that the monitoring functions operate on the most recently available BAS state. As a result, the same service layer supports scene updates, isolating and inspecting equipment, and on-demand visualization of historical trends in the time-series data. Read and write access are separated through distinct API-key roles for ingestion and query operations.

The integration workflow also aligns telemetry, BIM-derived scene entities, and institutional records around the same equipment instances. USD prim paths are associated with BAS devices through a metadata reconciliation process that uses identifiers and descriptive fields preserved during BIM conversion, together with metadata carried in the telemetry and institutional source records. The resulting equipment mappings are then used to attach asset and room context to the corresponding scene entities. This allows a selected object in Omniverse to expose current telemetry, asset metadata, room information, and later-added semantic relationships through a common scene-based representation.

4.3 Semantic modeling and context composition in OpenUSD

With the integrated data brought into the scene, the next step is to express what each entity represents within the HVAC system and how those entities relate to one another. In the proof of concept, this was implemented through native UsdRelationships to explore whether OpenUSD itself can carry building-specific categories and system relationships, rather than relying on an external graph database.

A dedicated ‘RelationshipManager’ prim holds classifier prims for the equipment categories used in the case study, and equipment prims are assigned to these classifiers so that roles such as AHUs and VAV terminals are explicit on the stage and classifier membership can be retrieved directly through native USD APIs. Additional relationships are then defined to represent system topology, for example, connecting AHUs to the VAV terminals they supply. In the proof of concept, the logic that generates these categories and relationships relies on project-specific naming and mapping conventions, whereas OpenUSD provides the native constructs for representing, storing, and querying them. These links are grouped under ‘Relationship Graph’, which separates them from other USD attributes and relationships in the property interface and allows the semantic relationships to be enumerated and queried as a distinct set. Figure 6 shows a classified AHU and its connected VAV terminals as they appear in the Omniverse property interface, and Figure 7 shows the corresponding graph, obtained through USD traversal of the relationship layer and rendered as a directed NetworkX graph with matplotlib.

Figure 6: Device inspection and trends interface in Omniverse DT.

These relationships add equipment classification and system connectivity to the same scene entities that already carry BIM-derived parameters, live BAS values exposed as native USD attributes under ‘Device Data’, and asset metadata composed through the integration step. Within the Omniverse property panel, BIM parameters, ‘Device Data’, ‘Asset Data’, and ‘Relationship Graph’ are presented under separate headings, and Figure 8 shows this organization for a selected asset together with the equipment listing used in the application. Maintaining this contextualized building information directly on the scene is important for facility-scale interpretation, which depends on spatial, operational, semantic, and institutional information being accessible from a shared reference rather than dispersed across disconnected viewers and platforms. When these sources are composed onto the same stage, native OpenUSD queries and Kit application logic can relate a selected asset to its category, connected equipment, and current operating state within one scene, providing the contextual basis for the analytical and monitoring functions that follow.

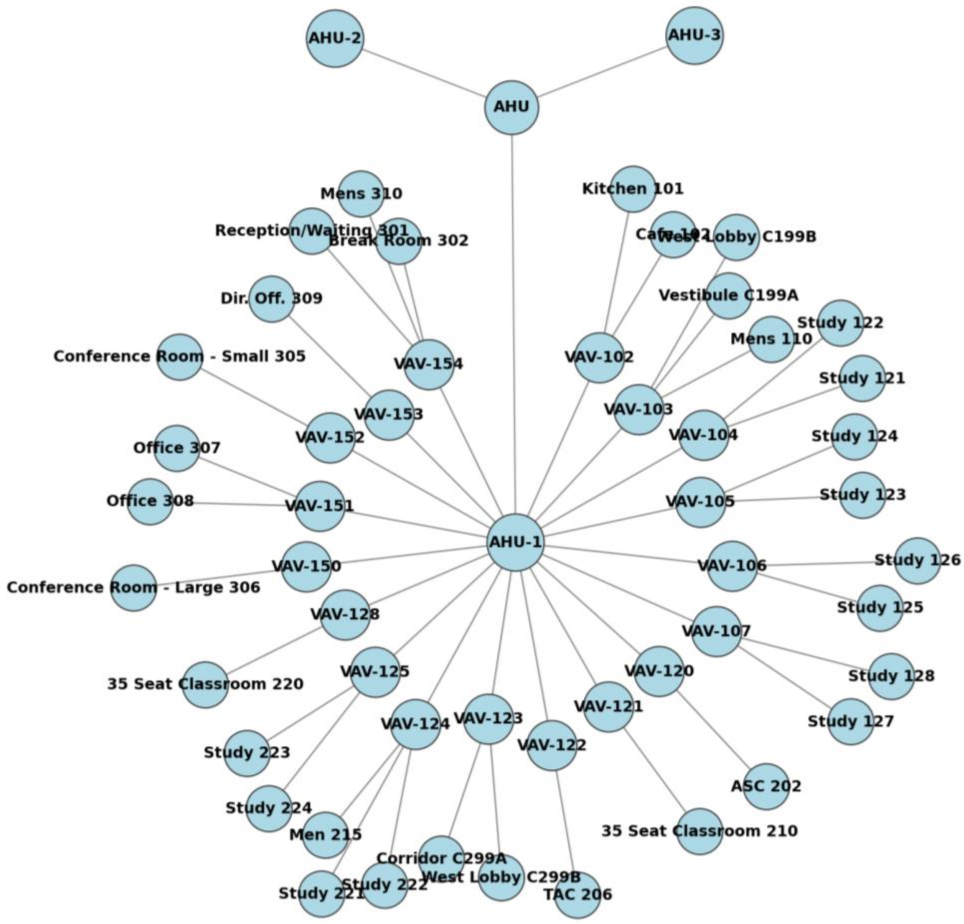

Figure 7: Relationship graph of the AHU 1 subsystem.

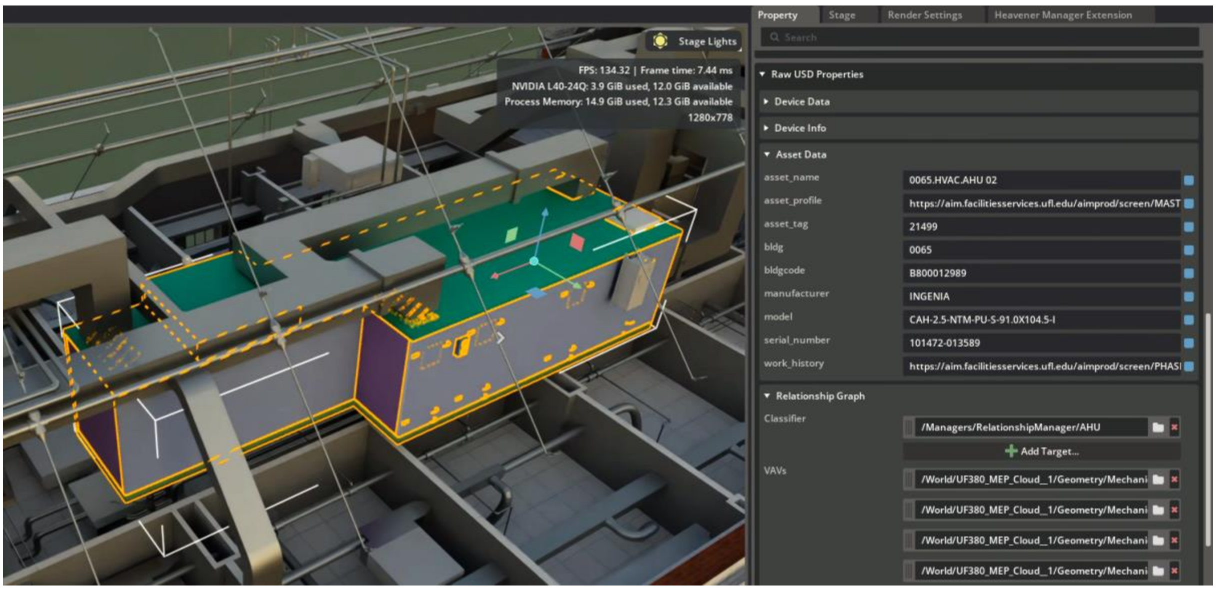

Figure 8: Property panel organization with representative asset and relationship data.

4.4 Analytical and monitoring functions

Assembling spatial, operational, semantic, and institutional context on a common USD stage creates the conditions for more context-aware operational interpretation, in which equipment behavior can be understood in relation to system configuration and live context. The case study implementation illustrates this potential through a set of core capabilities grounded in the USD scene, including comparative review of historical telemetry, stage-native monitoring of USD attributes, and context-aware notification logic informed by system relationships and institutional records.

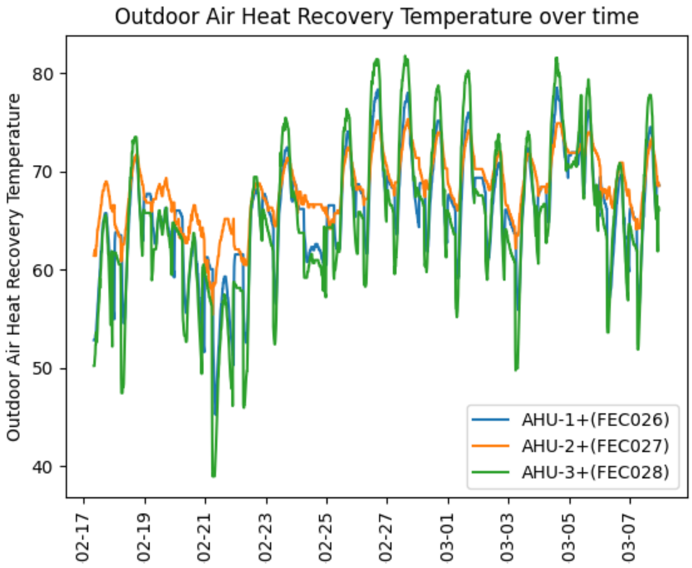

Figure 9: Multi-AHU Comparative Trend View.

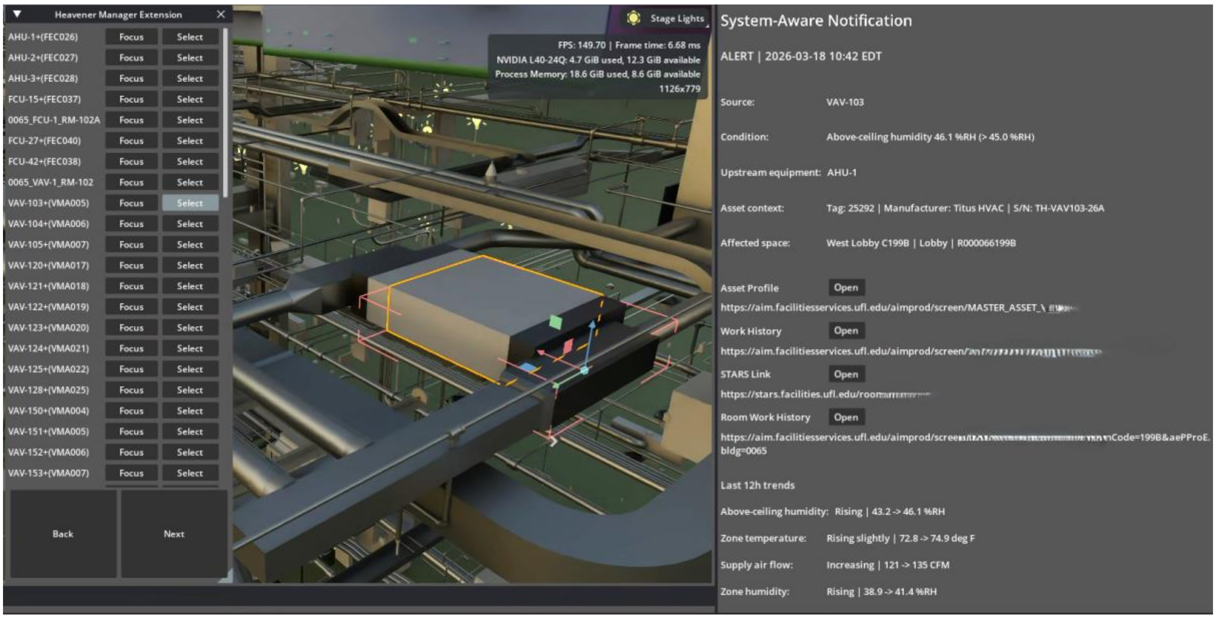

Figure 10: System-Aware Notification with Integrated Operational Context.

Comparative time-series review - draws on the historical retrieval capability of the time-series service described in Section 4.2. After the user selects a time range and a set of equipment, the application retrieves the corresponding historical payloads and identifies the telemetry points shared across the selection. Restricting the comparison to these common points keeps the resulting multi-equipment plot (Figure 9) aligned to the same measurement across equipment of comparable type, enabling comparisons such as shared AHU points across multiple air-handling units or zone and discharge conditions across multiple VAV terminals without predefining a separate analytic template for each case.

Attribute-level monitoring - includes an early-stage monitoring framework for reacting to changes in selected USD attributes after updated BAS values have been written onto the scene. Using Kit's native `UsdAttributeModel` to observe attribute updates, the framework supports alerts based on value changes, comparisons between two attributes, and threshold checks against configured values. Because these checks operate directly on stage attributes, changing equipment conditions can be identified through the same representation used for contextual review and comparative analysis.

Context-aware notification logic - monitoring becomes more actionable when interpreted together with the system relationships and asset context already represented on the stage. The case-study implementation illustrates this through notification logic in which a threshold exceedance on one asset can be interpreted together with the connected equipment, the downstream areas potentially affected along the distribution path, and the institutional asset information relevant to follow-up action. Rather than surfacing a breached value in isolation, the same event can therefore be associated with the equipment involved, the downstream areas potentially impacted, and the maintenance context available for response (Figure 10). This illustrates how live telemetry, semantic relationships, and institutional records can be combined into system-aware notifications.

5. DISCUSSION

Building on the case study DT implementation, this section examines the practical realization of the proposed architecture in a real building setting, with particular attention to integrated scene composition, semantic representation through native OpenUSD constructs, and the conditions that shape operational deployment.

5.1 Integrated building context

Facility management and building operations depend on multiple enterprise systems, each adopted at a different time for a distinct function and each carrying its own schema and conventions. Many operational questions are inherently contextual rather than reducible to isolated data retrieval. A warm room, for example, becomes meaningful only when interpreted against schedule, occupancy, upstream and downstream system connections, and the type of space and equipment involved. In practice, experienced operators can often answer such questions by navigating across fragmented systems and reconciling the results using tacit or domain knowledge. Conventional programmatic workflows, however, are far less effective in this setting because the information needed to interpret operational conditions is distributed across systems, represented inconsistently, and often left implicit rather than encoded in a machine-interpretable form.

The value of a DT lies not only in bringing this information into a common environment but also in aligning it to a shared data model so that observations can be interpreted within the building systems, spaces, and assets to which they refer. The DT developed in the case study addressed this by composing geometry, telemetry, asset metadata, room context, and equipment relationships onto the same scene entities, allowing contextualized review, cross-equipment performance comparison, threshold-based monitoring, and system-aware notifications to operate on a common representation.

This research also explored Omniverse and OpenUSD as an environment for semantic modeling and for representing building relationships directly on the scene. Equipment categories and distribution topology were expressed through native USD constructs and resolved alongside geometry and live operating data. In this way, the semantic layer became part of the same working representation used for analysis, operator-facing applications, and future AI-assisted workflows. Within this capability progression (KPMG, 2022; Autodesk, 2025), the implementation establishes the synchronization and information infrastructure needed to support increasingly advanced operational services. Continuous BAS telemetry ingestion maintains the physical-to-digital connection, while BIM-derived entities, institutional records, and HVAC relationships are composed around common equipment references. This integrated context supports current-state inspection, historical comparison, attribute-level monitoring, and system-aware notification within a shared runtime representation. The separation of data acquisition, contextual representation, and application logic also allows new analytical services to be developed or extended without requiring substantial changes to the mechanisms that collect, organize, and expose building information, provided that the shared data structures and interfaces remain consistent. This creates a structured basis for adding fault-detection, predictive, simulation, and AI-assisted workflows as operational requirements evolve.

5.2 OpenUSD for building semantic representation

Semantic modeling forms the foundation of any operationally useful DT, as analytical and control applications require machine-readable representations of equipment identity, system structure, and the relational organization of building systems. Standards such as Project Haystack and Brick address these needs through distinct modeling philosophies, each supported by its own conventions for entity typing, relationship representation, schema validation, and structured retrieval. Table 1 presents the OpenUSD constructs that provide comparable capabilities, alongside the corresponding mechanisms used in Project Haystack and Brick.

In OpenUSD, these capabilities map to native scene mechanisms. Entity identity and role can be represented through prim types, classifier structures, attributes, and attached metadata, while system-level connections can be encoded through USD relationships. Applied API schemas can define the properties and relationships expected of particular prim types, and application-level checks can evaluate scene content against those expectations. OpenUSD does not itself provide a predefined building-domain vocabulary equivalent to Project Haystack, Brick, or the Building Topology Ontology (BOT); rather, it provides general representational mechanisms through which such domain conventions can be implemented. One potential pathway is to translate selected building-domain classes, properties, and relationship predicates into applied USD schemas and scene relationships, allowing standards-informed context to be carried directly on scene prims rather than maintained solely in an external graph. Such an approach could allow the scene representation to reuse selected vocabulary and validation principles from established building-domain standards while retaining the compositional and runtime properties of OpenUSD. The prototype did not implement a complete translation of an established ontology; instead, it used classifier prims and USD relationships to encode the project-specific equipment categories and HVAC topology required by the selected workflows, thereby demonstrating the underlying scene-native representation pattern.

Table 1: Semantic modeling capabilities across Project Haystack, Brick, and OpenUSD.

Dimension	Project Haystack	Brick Schema	OpenUS (within Omniverse)
Modeling Approach	- Flexible, tag-based convention	- Formal class-based ontology (RDF/OWL)	- Composable scene description - Classifier prims and relationships
Entity Typing	- Marker and value tags - Xeto types for formalization	- Membership in a curated class hierarchy	- Prim types - Applied API schemas
Relationships	- Reference tags (e.g., equipRef, siteRef)	- Relationship predicates (e.g., feeds, hasPoint, isPartOf)	- USD relationships
Validation	- Tag conventions with Xeto-based schema validation	- SHACL shapes over the RDF graph	- Domain-specific schemas and conformance rules
Retrieval	- Haystack filter language over tag sets	- SPARQL over the RDF graph	- Stage traversal via USD APIs

Validation in this setting spans a range of checks that progress from general structural consistency to workflow-specific requirements. Structural checks verify that entities of a given type carry the properties and relationships expected for that type, for instance, that every VAV terminal is connected to an upstream AHU, or that equipment of a given class carries its required properties and points. Established building-semantic frameworks provide precedents for this form of validation, including Shapes Constraint Language (SHACL) for the formal validation of RDF graphs against structural shapes (Knublauch & Kontokostas, 2017) and Project Haystack's Xeto, which enables schema-based validation within Haystack environments (Project Haystack, 2025). Building on this foundation, application-level checks identify the subset of entities, relationships, and properties required by a particular analytical workflow, such as confirming that the points required by a fault-detection method are present and consistently represented for each air-handling unit before the method is executed. Xeto type libraries provide a practical precedent for this kind of validation through libraries based on reference sequences such as ASHRAE Guideline 36, which specify the equipment, points, and relationships required for standard-compliant control logic. Within Omniverse, analogous capabilities can be supported through applied USD schemas combined with rule-checking pipelines that encode equipment-type profiles and application-level manifests for particular operational workflows and compliance contexts.

In the prototype DT, equipment roles and HVAC topology were encoded directly on the scene through classifier prims and USD relationships, while BAS points were carried as attributes on their parent equipment entities. This configuration reflected both the structure of the available source information and the level of semantic detail required by the workflows examined in the study. The same semantic principles extend from equipment topology to point-level context. Compositional relationships indicate which points belong to which equipment, while operational relationships capture how measurements, setpoints, commands, and feedback signals are related within system behavior and control logic. A damper position sensor, for example, belongs to a VAV terminal, while the associated actuator command and airflow measurement are involved in regulating airflow through that terminal. Capturing such relations is therefore part of semantic representation, not only at the equipment level but also at the level of individual points and parameters.

The representation of this point-level context in OpenUSD depends on the semantic granularity required by the application and on the level of detail available in the source BIM or controls model. In the attribute-based approach adopted in the DT, points remain as attributes of an equipment prim, and USD relationships established at the prim level can target specific attribute paths on that prim when point-level references are needed. This makes it possible to represent dependencies among individual measurements, setpoints, commands, and feedback values without introducing a separate entity for each BAS point. Alternatively, sensors and control points can be modeled as distinct elements, each with its own class, properties, and relationships. This representation is appropriate when the point itself must carry information not naturally attributable to its parent equipment, such as instrument identity, calibration history, provenance, or confidence, or when it participates independently in multiple relationships across the scene. Such a representation depends on sensors and control points being available as discrete elements in the source BIM or controls model, a condition that many design-stage models do not satisfy and that may therefore require additional modeling before this approach can be adopted.

5.3 Practical deployment considerations

Extending this approach beyond the prototype DT depends substantially on the condition and completeness of the underlying source information. Traditional BIM delivery practices remain oriented primarily toward design and construction workflows and often place limited emphasis on the structured handover of operator-ready data, leaving downstream information incomplete, fragmented, or inconsistently aligned across systems. As a result, existing facilities commonly represent the same asset differently across the BAS, the BIM-derived model, and institutional asset records, such that asset identities, naming conventions, and related metadata must often be reconciled and verified before they can support reliable operational workflows. Addressing this issue ultimately requires stronger taxonomies and metadata conventions throughout delivery and commissioning, while Natural Language Processing (NLP) and language-model-assisted entity matching, coupled with human verification, could provide a practical approach for improving cross-system alignment in existing facilities (Xie et al., 2019; Huang, 2024). More broadly, this challenge underscores the persistent difficulty of maintaining consistent building information across operational systems over the facility lifecycle.

The reliability of the composed scene also depends on how accurately the source BIM reflects the current state of the facility. In existing buildings, models may not fully capture renovation history, undocumented field changes, or equipment replacement, with the result that the geometry and parameter content represented in the DT may diverge from present site conditions. Where visual or parametric fidelity affects operational interpretation, verification against field conditions remains necessary. Sustaining the long-term usefulness of the DT therefore also depends on maintaining the underlying BIM and associated information as the facility evolves.

Another practical consideration concerns model conversion. Converting a Revit model to USD with the NVIDIA Omniverse Revit Connector preserves useful geometry and can export Revit element parameters as USD properties, but material and texture translation is not always automatic and may require additional remapping or post-export adjustment (NVIDIA Corporation, 2025). Similar practical issues apply to semantic modeling. In the prototype DT, equipment classifications and system relationships were derived from project-specific naming and mapping conventions, with the resulting assignments manually verified for accuracy. Broader deployment will require more standardized conventions and more authoritative building-information sources for deriving classifications and system relationships, together with the more systematic extraction approaches discussed in Section 6.1. Data management choices similarly reflect the scope of the case study. The telemetry service uses PostgreSQL with JSONB payloads, selected for operational simplicity and for schema flexibility in accommodating the heterogeneous point sets carried by different equipment types. This arrangement supported ingestion, retrieval, and historical query for the scale considered in this work, though a purpose-built time-series database may be more suitable as deployment extends to larger portfolios or higher-frequency workloads.

The evidence provided by the case study is primarily architectural and technical. The comparative, monitoring, and notification functions demonstrate how operational workflows can be composed over the integrated scene context, but their effects on diagnostic accuracy, maintenance efficiency, decision quality, energy performance, and user outcomes were not evaluated. Further work should therefore examine these operational outcomes, validate the semantic representation more systematically, and assess the effort required to adapt the architecture across additional buildings and source-system configurations, as discussed in Section 6.

6. FUTURE RESEARCH DIRECTIONS

This study presented a system architecture and developed a proof-of-concept implementation for DT development on NVIDIA Omniverse and OpenUSD, demonstrating an integrated ecosystem of Kit-based extensions oriented toward facility operations and maintenance. The work addressed foundational aspects of DT development, including source-system integration, semantic modeling using native OpenUSD constructs, and analytical workflow composition, as well as an examination of the practical conditions that shape deployment in existing facilities. Several directions for future work follow from this foundation.

6.1 Systematic semantic generation and validation

Semantic classifications and relationships in the prototype were derived from project-specific naming conventions and manually verified, and scaling the representation beyond HVAC to additional building systems and to the portfolio level will require systematic methods for generating equipment classifications, system topology, and point-level context from authoritative building information sources. Relevant foundations include Revit-API-based and IFC-based MEP connectivity extraction, geometry-based semantic enrichment of imperfect BIMs, and ontology-based representations of served-space relationships (Wang et al., 2023; Li & Zhang, 2022; Mavrokapnidis et al., 2023; Lilis et al., 2025), together with validation pipelines that confirm structural completeness and processes that maintain the representation as facilities evolve through renovation and equipment replacement. Within Omniverse, these mechanisms can inform how equipment-type profiles and application-level conformance checks are encoded through applied USD schemas and conformance-checking pipelines, particularly where such schemas are aligned with established building-domain ontologies.

6.2 Advanced analytics and AI-enabled workflows

The analytical and monitoring functions demonstrated in the prototype are deliberately narrow in scope, and the composed scene supports broader workflows, including rule-based fault detection aligned with operational standards such as ASHRAE Guideline 36 and APAR (ASHRAE, 2021; Schein et al., 2006), machine-learning-based anomaly detection, and energy analysis informed by operational and spatial context. Event-driven workflows can be supported by USD's built-in scene-change notifications, allowing analytical routines to be triggered when relevant attributes or relationships update.

AI-enabled workflows that reason over the composed context, invoke external tools, and interact with connected platforms (Yao et al., 2023; Schick et al., 2023) represent a further direction, with potential applications including fault interpretation in relation to system configuration and operating history, automated drafting of maintenance documentation and work orders, and cross-building review of equipment behavior through a consistent semantic and spatial framework.

6.3 Operational delivery and broader application contexts

The prototype was realized as a single Kit-based application in one academic building, and extending the approach will require both broader delivery modalities and portfolio-scale coordination. Web-based, mobile, and equipment-linked entry points such as QR codes could align access with the inspection, review, and coordination activities carried out by different user groups, while integration with maintenance-management and work-order systems would allow findings surfaced in the scene to flow directly into the enterprise platforms where documentation, scheduling, and follow-up action are managed.

At larger scale, Omniverse's support for geospatial composition and large-scale scene coordination, including integration pathways such as Cesium for Omniverse (CesiumGS, n.d.) creates a basis for situating multiple building digital twins within a shared campus or district context, and its simulation capabilities open additional avenues including control-strategy evaluation in sandboxed settings, robotic inspection and maintenance workflows through integration with NVIDIA Isaac Sim, and immersive training environments for operators and technicians.

A further priority is structured evaluation with facility managers, operators, and maintenance personnel. Task-based studies can examine whether the integrated environment improves the efficiency, completeness, or usability of routine equipment review, diagnostic investigation, and maintenance-information retrieval relative to existing multi-system practices. Such evaluation is also needed to identify the interface, training, trust, and workflow requirements that cannot be established through technical implementation alone.

7. CONCLUSION

A DT for the built environment must function as more than a three-dimensional viewer; its operational value depends on an environment that integrates heterogeneous source systems, represents building semantics, and supports analytical workflows over the combined context. This paper examined NVIDIA Omniverse and OpenUSD as such an environment through a five-layer reference architecture and a prototype DT developed in an academic building setting. Through the case study implementation, the work assessed both the architectural feasibility of this approach in a real facility setting and the capacity of native OpenUSD constructs to carry operationally relevant semantic context directly on the scene.

The proof of concept demonstrated the technical feasibility of composing BIM-derived spatial context, live and historical BAS telemetry, institutional records, and equipment relationships directly within Omniverse to support contextualized equipment review, comparative analysis, monitoring, and system-aware notification. Current BAS values were continuously refreshed onto scene attributes, and these capabilities operated directly on the composed scene, so that operational and semantic context could be queried through a single environment rather than reconstructed across separate viewers and databases.

The study further showed that OpenUSD can represent equipment classifications and system relationships through native scene constructs. Classifier prims and directional UsdRelationships encoded equipment-level semantics and HVAC topology directly on the scene, and applied schemas and rule-based checks provide a pathway toward richer point-level semantics and standards-informed validation. OpenUSD supplies these general-purpose representational mechanisms rather than a building-domain ontology; its contribution in this architecture is a programmable scene representation in which operational context is carried alongside spatial content and made available to modular application logic.

Several directions warrant further investigation, including the systematic population of the semantic infrastructure from authoritative building-information sources, domain-specific validation frameworks, broader analytical and AI-assisted workflows, interfaces suited to routine operational use, and extension to portfolio scale. More broadly, the work points toward a role for OpenUSD-based environments as operational infrastructure for the built environment: a shared and extensible representation on which the AI-driven, automated, and lifecycle-spanning applications expected to define the next generation of facility operation can be developed, integrated, and sustained.

ACKNOWLEDGMENT

The authors would like to acknowledge Rhenn Handel for technical support in developing the software implementation and the Omniverse scene used in this study. The authors also acknowledge the UF CIO for providing research funding and UF Research Computing and UF Facilities for providing resources and support that facilitated this research.

DATA AVAILABILITY STATEMENT

The operational datasets and full application source code are not publicly available because they include institution-specific system information and are subject to cybersecurity, access-control, and data-governance restrictions. Representative configuration schemas, mapping examples, and methodological documentation may be made available by the corresponding author upon reasonable request, subject to institutional approval.

REFERENCES

Alliance for OpenUSD. (n.d.). Alliance for OpenUSD. https://aousd.org/

Almadhor, A., Ghazouani, N., Bouallegue, B., Kryvinska, N., Alsubai, S., Krichen, M., Al Hejaili, A., & Sampedro, G. A. (2025). Digital twin based deep learning framework for personalized thermal comfort prediction and energy efficient operation in smart buildings. Scientific Reports, 15, 24654. https://doi.org/10.1038/s41598-025-10086-y

Ammar, M., Rost, C., Tommasini, R., Agarwal, S., Bonifati, A., Selmer, P., Kharlamov, E., & Rahm, E. (2025). Towards hybrid graphs: Unifying property graphs and time series. In Proceedings of the 28th International Conference on Extending Database Technology (EDBT 2025) (pp. 970–977). OpenProceedings. https://doi.org/10.48786/edbt.2025.79

Archi Communications Team. (2025, March 20). NVIDIA Omniverse is revolutionizing building development and management. Archi Blog. https://goto.archi/blog/post/nvidia-omniverse-is-revolutionizing-building-development-and-management

Arsiwala, A., Elghaish, F., & Zoher, M. (2023). Digital twin with machine learning for predictive monitoring of CO2 equivalent from existing buildings. Energy and Buildings, 284, 112851. https://doi.org/10.1016/j.enbuild.2023.112851

Asare, K. A. B., Liu, R., Anumba, C. J., & Issa, R. R. A. (2024). Real-world prototyping and evaluation of digital twins for predictive facility maintenance. Journal of Building Engineering, 97, 110890. https://doi.org/10.1016/j.jobe.2024.110890

ASHRAE. (2021). Guideline 36-2021: High-performance sequences of operation for HVAC systems. ASHRAE.

ASHRAE. (n.d.). Titles, purposes, and scopes: SPC 223P, Semantic data model for analytics and automation applications in buildings. Retrieved April 22, 2026, from https://www.ashrae.org/technical-resources/standards-and-guidelines/titles-purposes-and-scopes

Autodesk. (2025). What is a digital twin and why is it so valuable? Autodesk Tandem. Retrieved from https://intandem.autodesk.com/about-digital-twins/

Balaji, B., Bhattacharya, A., Fierro, G., Gao, J., Gluck, J., Hong, D., Johansen, A., Koh, J., Ploennigs, J., Agarwal, Y., Berges, M., Culler, D., Gupta, R., Kjærgaard, M. B., Srivastava, M., & Whitehouse, K. (2016). Brick: Towards a unified metadata schema for buildings. In Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '16) (pp. 41–50). Association for Computing Machinery. https://doi.org/10.1145/2993422.2993577

Bhatia, N., Costa, R., Pamela, S. J. P., Davis, A., Gonzalez Beltran, A. N., Gopakumar, V., Zanisi, L., de Witt, S., & Akers, R. (2025). Visualizing digital twins of fusion power plants using NVIDIA Omniverse. AIP Advances, 15(4), 045018. https://doi.org/10.1063/5.0261883

Biagini, C., Bongini, A., & Marzi, L. (2024). From BIM to digital twin: IoT data integration in asset management platform. Journal of Information Technology in Construction (ITcon), 29, 1103–1127. https://doi.org/10.36680/j.itcon.2024.049

CesiumGS. (n.d.). Cesium for Omniverse. Retrieved April 18, 2026, from https://cesium.com/platform/cesium-for-omniverse

Colombo, M., Gelfusa, G., Greco, G., & Picchioni, F. M. B. (2023, December 7). Utilizing NVIDIA Omniverse™ for digital twin projects at the NTT DATA Innovation Center. NTT DATA Focus. https://www.nttdata.com/global/en/insights/focus/2023/utilizing-nvidia-omniverse-for-digital-twin-projects-at-the-ntt-data-innovation-center

Corke, G. (2022, August 9). Nvidia looking to encode the IFC schema in USD. AEC Magazine. https://aecmag.com/collaboration/nvidia-looking-to-encode-the-ifc-schema-in-usd/

El Mokhtari, K., Panushev, I., & McArthur, J. J. (2022). Development of a cognitive digital twin for building management and operations. Frontiers in Built Environment, 8, Article 856873. https://doi.org/10.3389/fbuil.2022.856873

Elias, C., & Issa, R. R. A. (2023). Digital twins for smart decision making in asset management. In P. Capone, V. Getuli, F. Pour Rahimian, N. Dawood, A. Bruttini, & T. Sorbi (Eds.), CONVR 2023—Proceedings of the 23rd International Conference on Construction Applications of Virtual Reality: Managing the digital transformation of construction industry (pp. 1255–1260). Firenze University Press. https://doi.org/10.36253/979-12-215-0289-3.123

Elias, C., & Issa, R. R. A. (2025). AI-driven digital twins for predictive maintenance. Proceedings of the 2025 European Conference on Computing in Construction & 42nd International CIB W78 Conference on Information Technology in Construction (pp. 654–661). European Council on Computing in Construction. https://doi.org/10.35490/EC3.2025.369

Elias, C., Southern, I., & Issa, R. R. A. (2025). Digital twins for smart decision making in facilities and asset management (FAM). In A. Francis, E. Miresco, & S. Melhado (Eds.), Advances in information technology in civil and building engineering: ICCCBE 2024 (Lecture Notes in Civil Engineering, Vol. 628). Springer. https://doi.org/10.1007/978-3-031-84208-5_22

Eneyew, D. D., Capretz, M. A. M., & Bitsuamlak, G. T. (2022). Toward smart-building digital twins: BIM and IoT data integration. IEEE Access, 10, 130487–130506. https://doi.org/10.1109/ACCESS.2022.3229370

Gu, Y., & Jazizadeh, F. (2024). Digital twin-assisted anomaly detection for smarter built environment management: A literature review. In J. S. Shane, K. M. Madson, Y. Mo, C. Poleacovschi, & R. E. Sturgill (Eds.), Advanced technologies, automation, and computer applications in construction (Construction Research Congress 2024, Vol. 1, pp. 1347–1357). American Society of Civil Engineers. https://doi.org/10.1061/9780784485262.137

Huang, Z. (2024). Disambiguate entity matching using large language models through relation discovery. In Proceedings of the Conference on Governance, Understanding and Integration of Data for Effective and Responsible AI (GUIDE-AI '24) (pp. 36–39). Association for Computing Machinery. https://doi.org/10.1145/3665601.3669844

International Organization for Standardization. (2024). Industry foundation classes (IFC) for data sharing in the construction and facility management industries—Part 1: Data schema (ISO 16739-1:2024). ISO.

Karabulut, E., Pileggi, S. F., Groth, P., & Degeler, V. (2024). Ontologies in digital twins: A systematic literature review. Future Generation Computer Systems, 153, 442–456. https://doi.org/10.1016/j.future.2023.12.013

Khan, J. A., & Bartolini, A. (2025). A unified ontology for scalable knowledge graph-driven operational data analytics in high-performance computing systems. arXiv. https://arxiv.org/abs/2507.06107

Khan, J. A., Pizzini Cavagna, H., Proia, A., & Bartolini, A. (2025). From data center IoT telemetry to data analytics chatbots—Virtual knowledge graph is all you need. arXiv. https://arxiv.org/abs/2506.22267

Knublauch, H., & Kontokostas, D. (2017). Shapes constraint language (SHACL). W3C Recommendation. https://www.w3.org/TR/shacl/

KPMG. (2022). Insight Report: Innovation and R&D in Construction (pp. 1–39). Canadian Construction Association.

Li, H., & Zhang, J. (2022). IFC-based information extraction and analysis of HVAC objects to support building energy modeling. In Proceedings of the 39th International Symposium on Automation and Robotics in Construction (ISARC 2022) (pp. 159–166). International Association for Automation and Robotics in Construction. https://doi.org/10.22260/ISARC2022/0024

Lilis, G. N., Wang, M., Katsigarakis, K., Mavrokapnidis, D., Korolija, I., & Rovas, D. (2025). BIM-based semantic enrichment and knowledge graph generation via geometric relation checking. Automation in Construction, 173, 106081. https://doi.org/10.1016/j.autcon.2025.106081

Lu, Q., Parlikad, A. K., Woodall, P., Don Ranasinghe, G., Xie, X., Liang, Z., Konstantinou, E., Heaton, J., & Schooling, J. (2020a). Developing a digital twin at building and city levels: Case study of West Cambridge Campus. Journal of Management in Engineering, 36(3), 05020004. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000763

Ma, T., Xiao, F., Zhang, C., Zhang, J., Zhang, H., Xu, K., & Luo, X. (2025). Digital twin for 3D interactive building operations: Integrating BIM, IoT-enabled building automation systems, AI, and mixed reality. Automation in Construction, 176, 106277. https://doi.org/10.1016/j.autcon.2025.106277

Mavrokapnidis, D., Lilis, G. N., Katsigarakis, K., Korolija, I., & Rovas, D. (2023). Semi-automated extraction of HVAC system topology from imperfect Building Information Models. In Proceedings of Building Simulation 2023: 18th Conference of IBPSA (pp. 964–971). IBPSA. doi: 10.26868/25222708.2023.1392

Merino, J., Xie, X., Moretti, N., Chang, J. Y., & Parlikad, A. (2023). Data integration for digital twins in the built environment based on federated data models. Proceedings of the Institution of Civil Engineers – Smart Infrastructure and Construction, 176(4), 194–211. https://doi.org/10.1680/jsmic.23.00002

McKenna, J., Sivaraman, G., & Bhushan, S. (2024, November 19). Connect real-time IoT data to digital twins for 3D remote monitoring. NVIDIA Technical Blog. https://developer.nvidia.com/blog/connect-real-time-iot-data-to-digital-twins-for-3d-remote-monitoring/

Mulayim, O. B., Paul, L., Pritoni, M., Prakash, A. K., Sudarshan, M., & Fierro, G. (2024). Large language models for the creation and use of semantic ontologies in buildings: Requirements and challenges. In Proceedings of the 11th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '24) (pp. 312–317). Association for Computing Machinery. https://doi.org/10.1145/3671127.3698792

Nguyen, T. D. H. N., Ly, D. H., Jang, H., Dinh, H. N. N., & Ahn, Y. (2025). Digital twin framework to enhance facility management for relocatable modular buildings. Automation in Construction, 176, 106249. https://doi.org/10.1016/j.autcon.2025.106249

NVIDIA Corporation. (2025). Revit — Omniverse Connect. Omniverse Connect documentation. NVIDIA.

NVIDIA. (2026a). Architecture. Omniverse Kit manual. https://docs.omniverse.nvidia.com/kit/docs/kit-manual/latest/guide/kit_architecture.html

NVIDIA. (2026b). Extensions overview. Omniverse Extensions documentation. https://docs.omniverse.nvidia.com/extensions/latest/index.html

OpenUSD. (2026a). Introduction to USD. Universal Scene Description 26.03 documentation. https://openusd.org/release/intro.html

OpenUSD. (2026b). UsdRelationship class reference. Universal Scene Description 26.03 documentation. https://openusd.org/release/api/class_usd_relationship.html

Pishdad-Bozorgi, P., Gao, X., Eastman, C., & Self, A. P. (2018). Planning and developing facility management-enabled building information model (FM-enabled BIM). Automation in Construction, 87, 22–38. https://doi.org/10.1016/j.autcon.2017.12.004

Pritoni, M., Paine, D., Fierro, G., Mosiman, C., Poplawski, M., Saha, A., Bender, J., & Granderson, J. (2021). Metadata schemas and ontologies for building energy applications: A critical review and use case analysis. Energies, 14(7), 2024. https://doi.org/10.3390/en14072024

Project Haystack. (2018, March 2). ASHRAE's BACnet Committee, Project Haystack and Brick Schema collaborating to provide unified data semantic modeling solution [Press release]. https://marketing.project-haystack.org/project-haystack-media/press-releases/ashraes-bacnet-committee-project-haystack-and-brick-schema-collaborating-to-provide-unified-data-semantic-modeling-solution

Project Haystack. (2025, May 13). Project Haystack unveils Haystack 5 with Xeto schema language and RDF integration at Haystack Connect 2025 [Press release]. https://marketing.project-haystack.org/project-haystack-media/press-releases/project-haystack-unveils-haystack-5-with-xeto-schema-language-and-rdf-integration-at-haystack-connect-2025

Ramonell, C., Chacón, R., & Posada, H. (2023). Knowledge graph-based data integration system for digital twins of built assets. Automation in Construction, 156, 105109. https://doi.org/10.1016/j.autcon.2023.105109

Saldana Ochoa, K., Delavar, Y., Balakrishnan, D., Elias, C., Srinivasan, R. S., & Anumba, C. J. (2025). A theoretical framework for integrating digital twins in building lifecycle management. In W. Lu & C. J. Anumba (Eds.), Routledge handbook of smart built environment (pp. 174–189). Routledge. https://doi.org/10.1201/9781003383840

Schein, J., Bushby, S. T., Castro, N. S., & House, J. M. (2006). A rule-based fault detection method for air handling units. Energy and Buildings, 38(12), 1485–1492. https://doi.org/10.1016/j.enbuild.2006.04.014

Schick, T., Dwivedi-Yu, J., Dessi, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. Advances in Neural Information Processing Systems, 36.

Simio. (n.d.). Harness the power of NVIDIA Omniverse digital twins: Visualize, validate, and optimize with unparalleled fidelity. Simio. https://www.simio.com/nvidia-omniverse-digital-twin-integration/

Tang, M., Nikolaenko, M., Alrefai, A., & Kumar, A. (2025). Metaverse and digital twins in the age of AI and extended reality. Architecture, 5(2), 36. https://doi.org/10.3390/architecture5020036

Vatanen, J. (2024). Exploring NVIDIA Omniverse ecosystem [Bachelor's thesis, Oulu University of Applied Sciences]. Theseus. https://urn.fi/URN:NBN:fi:amk-202405079838

Wajid, M. S., Terashima-Marín, H., Najafirad, P., Conant-Pablos, S. E., & Wajid, M. A. (2024). DTwin-TEC: An AI-based TEC district digital twin and emulating security events by leveraging knowledge graph. Journal of Open Innovation: Technology, Market, and Complexity, 10(2), 100297. https://doi.org/10.1016/j.joitmc.2024.100297

Wandersman, H. (2023, October 2). Using NVIDIA Omniverse to prepare 3D assets from building information models for AWS IoT TwinMaker. The Internet of Things on AWS – Official Blog. https://aws.amazon.com/blogs/iot/using-nvidia-omniverse-to-prepare-3d-assets-from-building-information-models-for-aws-iot-twinmaker/

Wang, Y., Lin, P., Zhang, L., Yu, H., & Robert, T. L. K. (2023). Resilience-oriented design for complex MEP systems in BIM. Advanced Engineering Informatics, 55, 101902. https://doi.org/10.1016/j.aei.2023.101902

Xie, Q., Zhou, X., Wang, J., Gao, X., Chen, X., & Liu, C. (2019). Matching real-world facilities to building information modeling data using natural language processing. IEEE Access, 7, 119465–119475. https://doi.org/10.1109/ACCESS.2019.2937219

Xie, X., Merino, J., Moretti, N., Pauwels, P., Chang, J. Y., & Parlikad, A. (2023). Digital twin enabled fault detection and diagnosis process for building HVAC systems. Automation in Construction, 146, 104695. https://doi.org/10.1016/j.autcon.2022.104695

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). ReAct: Synergizing reasoning and acting in language models. In Proceedings of the 11th International Conference on Learning Representations (ICLR 2023).