buildings

MDPI

Article

From Building Deliverables to Open Scene Description: A Pipeline for Lifecycle 3D Interoperability

Guoqian Ren [1], Chengzheng Huang [2] and Tengxiang Su [2,*]

[1] College of Architecture and Urban Planning, Tongji University, Shanghai 200092, China; rengq@tongji.edu.cn
[2] School of Engineering, Cardiff University, 52 The Parade, Cardiff CF24 3AB, UK; vincenzo_huang@163.com
* Correspondence: sut2@cardiff.ac.uk

Abstract

Industrial deliverables in the AEC/FM sector are increasingly specified, validated, and governed by open standards. However, the machine-readable delivery specifications rarely propagate intact into the real-time collaborative 3D scene descriptions required by digital twins, XR, large-scale simulation, and visualization. This paper proposes a pipeline that transforms industrial deliverables into semantically faithful, queryable, and render-ready open scene descriptions. Unlike existing workflows that focus on geometric translation via connectors or intermediate formats, the proposed pipeline aligns defined delivery specifications with schema-aware USD composition so that contractual semantics remain executable in the scene. The pipeline comprises delivery specification, which records required objects, attributes, and provenance as versioned rule sets; semantically bound scene realization, which builds an open scene graph that preserves spatial hierarchy and identifiers, while linking rich properties through lightweight references; and interactive sustainment, which lets multiple engines render, analyze, and update the scene while allowing rules to be re-applied at any time. It presents a prototype and roadmap that make open scene description a streaming-ready execution layer for building deliverables, enabling consistent semantics, and reuse across diverse 3D engines.

Keywords: openBIM; OpenUSD; information delivery; 3D visualization; integration

Academic Editors: Md Morshed Alam and Antonio Caggiano

Received: 15 October 2025
Revised: 2 December 2025
Accepted: 9 December 2025
Published: 12 December 2025

Citation: Ren, G.; Huang, C.; Su, T. From Building Deliverables to Open Scene Description: A Pipeline for Lifecycle 3D Interoperability. *Buildings* **2025**, *15*, 4503. https://doi.org/10.3390/buildings15244503

1. Introduction

Information deliverables in architecture, engineering, construction (AEC) and facility management (FM) are expressed as explicit machine-readable specifications [1]. Models, attributes, and their relationships are defined so that they can be validated and traced throughout the project lifecycle. In parallel, real-time and collaborative three-dimensional scenes have become the operational substrate for digital twins and large-scale visualization [2]. However, once data are mapped into the scene graphs that support real-time environments, governed information specifications seldom persist in a verifiable form. The resulting scenes often lack the semantic structure that was present at the point of contractual delivery [3]. Prevailing practice treats the problem as one of model translation. A model is exported from an authoring tool, converted through one or more intermediates, and imported into a three-dimensional engine. This is effective for producing renderable geometry, but it does not preserve the organizing principles of the original information deliverables. When delivery requirements evolve, the pipeline is re-run or patched, resulting in duplicated effort and erosion of trust in the scene as an authoritative representation [4].

An operational scene used downstream should instead be derived under the same delivery specification that governed the upstream model [5]. This calls for a conceptual shift: from converting one file format into another to transforming an industrial deliverable into a scene description that retains identity, hierarchy, and semantics, while remaining lightweight for real-time use and open to heterogeneous engines [6].

Information-centric initiatives in both building open standards and graphics communities are converging. The Information Delivery Specification (IDS), released by buildingSMART, turns project delivery rules into machine-readable form so that the specifications that govern model hand-over can also guide validation in real-time scenes [7–10]. On the graphics side, Pixar's Universal Scene Description (USD) defines composition, layering, and referencing mechanisms that guarantee deterministic scene semantics across platforms [11,12]. Building on this foundation, Pixar, Adobe, Apple, Autodesk, and NVIDIA, together with the Joint Development Foundation, launched the Alliance for OpenUSD to drive standardization and cross-industry adoption [13]. Early results show rapid uptake: OpenUSD, often deployed through NVIDIA Omniverse, enables a federated design that keeps BIM semantics intact while offering real-time rendering and AI-assisted workflows [3,14].

In this paper, conventional data fidelity is understood as the preservation of geometric accuracy and attribute values at the time of conversion between formats. By contrast, we use semantic continuity to denote not only geometry and properties but also delivery-side semantics: stable identifiers, spatial and system hierarchies, and the rule context expressed in information-delivery specifications, such that the same checks can be replayed on downstream representations. Without such semantic continuity, real-time scenes cannot be treated as authoritative views of contractual deliverables, and interoperability remains limited to visual consistency. Framing the pipeline in these terms allows the scene to remain an executable view of the delivery contract over time, rather than a detached snapshot of the source file. Accordingly, this study addresses the following research questions:

Q1: How can information-delivery specifications be used to drive scene construction so that semantic continuity across identifiers, hierarchies, and rule context is preserved in real-time 3D environments?

Q2: Can a specification-driven pipeline achieve practical performance while retaining more delivery semantics than existing connector- and converter-based workflows used in AEC/XR?

To answer these questions, we develop a specification-driven pipeline that treats information deliverables as a scene contract and derives a semantics-bound USD representation from them. The pipeline combines hierarchy-preserving mappings to maintain delivery semantics in a lightweight engine-agnostic scene, and we implement and evaluate this design on real project models against representative connector- and converter-based workflows. The structure of the paper is as follows: Section 2 reviews related efforts in information exchange and open scene description. Section 3 elaborates the stages of the proposed pipeline. Section 4 describes the prototype implementation and its evaluation on representative architectural, electrical, and steel-structure datasets. Section 5 discusses the limitations and future work.

2. Related Works

2.1. Data Schema and Information Delivery Specifications

IFC, as the core specification for data exchange in the engineering domain, defines the geometry, semantics, properties, and interrelationships of building components, thereby facilitating information integration and reuse across the building lifecycle. IFC supports multiple geometric representations, including BRep, Swept Solid, and CSG, and possesses strong semantic expressiveness [15]. It has also expanded to support infrastructure, geographic information systems (GIS), as well as temporal and spatial representations [16]. Major AEC software platforms have incorporated extensive support for the IFC standard. According to IFC standardization research, the schema is evolving from IFC2x3 to IFC4x3 and further to IFC5, which will enhance integration with other domains such as Life Cycle Assessment (LCA), FM, and GIS systems [17,18]. As BIM data coupling across disciplines becomes stronger, the generic attributes in source models can be further pruned and refined to improve interoperability [19,20].

IDS and bSDD (buildingSMART Data Dictionary) have been incorporated into the openBIM standards framework, working in close coordination with IFC to address its shortcomings in data validation and semantic consistency. Numerous studies have analyzed the extensibility of the IFC schema and the technical integration pathways with IDS and bSDD. IDS can precisely define which elements, properties, and validation rules a model should contain at a given project stage, thereby enabling automated compliance checking, and has been gradually applied in scenarios such as urban regeneration and prefabricated construction [21]. IDS can simplify model compliance checking, particularly demonstrating broad applicability in rule-based building code evaluations such as accessibility and fire resistance [10]. In addition, Zhu (2024) proposed an "intelligent review framework" that combines semantic ontologies with IDS-based validation methods to address semantic conflicts and information redundancy currently present in BIM deliveries [22]. There is also research that focuses on integrating circularity assessment indicators into BIM by using IDS to define property requirements, thereby enhancing delivery consistency in sustainable design scenarios [9].

As a semantically neutral dictionary, Data Dictionary provides IFC entities with unified URIs, class definitions, language mappings, and property standards, improving terminology consistency across different regions and software platforms. This data dictionary is highly coupled with the IFC Schema and IDS specification, becoming one of the key infrastructures for the semantic enrichment and extensibility. Research has shown that bSDD demonstrates value in scenarios such as heritage conservation, data integration, and sustainable building assessment [23–25]. Studies indicate that integrating bSDD into automated compliance-checking workflows can ensure data consistency, particularly in the connection of component libraries, product data, Environmental Product Declarations (EPDs), and LCA models [26].

2.2. Open Scene Description

Amid rising demand for cross-disciplinary digital twins and real-time visualization, long-standing 3D interchange formats show clear limits: (i) inconsistent carriage of rich heterogeneous scene semantics across tools, leading to metadata loss in downstream pipelines [3]; (ii) inadequate support for voxels, point clouds, field data, ultra-large multi-source assets, and long-chain provenance [27]; and (iii) absence of native mechanisms for parallel collaboration and change management [28]. In workflows that integrate multiple standards across engineering, manufacturing, and media, these gaps cause interoperability conflicts and semantic loss.

To address these challenges, recent research and industrial practice have converged on the OpenUSD approach. This class of frameworks emphasizes complete declarative serialization of digital scene elements, using an extensible type of system to unify the geometry, materials, animation, physics, and temporal data [29]. Non-destructive composition serves as the core mechanism: through combinations of referencing, variants, overrides, on-demand loading, and namespaces, multi-source assets can be assembled, trimmed, and reused within a single scene graph. Unlike linear "write-and-overwrite" file workflows, this model enables multiple participants to edit slices of a scene in parallel on separate branches, while preserving the provenance and minimal-diff representations at composition time, thereby reducing redundant duplication [30,31]. At the engineering level, OpenUSD implements sparse overrides and layered composition to enable fine-grained, non-destructive edits at the hierarchy level; employs variants and instancing to balance parametric design exploration with memory and storage for large structures; enforces consistent units, precise time sampling, comprehensive metadata, and asset resolution for cross-system consistency; extends via schemas and bindings to integrate domain-specific data for mechanics, simulation, rendering, and sensing; and provides loosely coupled paths to real-time rendering and physics backends to meet low-latency needs from preview to delivery [27,30,32]. These capabilities support a shift from file exchange to scene collaboration. From a software-ecosystem perspective, sustained governance by cross-industry alliances is moving open scene description from organization-specific practice to a shared substrate for cross-domain collaboration [33,34]. Standardized interfaces and compliance constraints improve alignment with engineering data standards and quality-assurance workflows [35]. Coupled with real-time collaboration and physics-based simulation, OpenUSD is enabling high-fidelity digital twins and automated pipelines across sectors, supporting multi-disciplinary semantic alignment and spatiotemporal coordination in AECO, closed-loop validation in manufacturing and robotics, and asset-management continuity with parallel authoring in content production [36].

2.3. Existing BIM-to-Real-Time Scene Workflow

Although the IFC open exchange standard has been implemented, a systematic semantic mapping and composition mechanism targeting USD remains lacking (Figure 1). There are three principal pathways for converting AEC data schema to USD (Figure 2): (i) connector-based conversion via BIM authoring tools, (ii) platform-mediated conversion within a USD-centric collaboration environment, and (iii) transcodings that proceed through intermediate exchange formats.

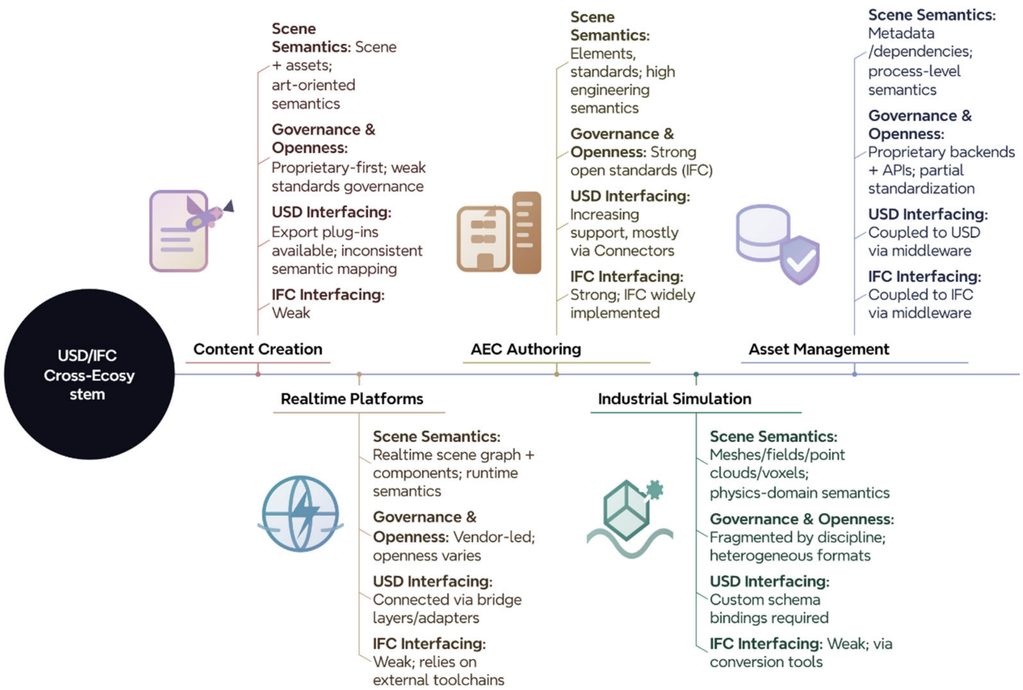


Figure 1. USD/IFC cross-ecosystem in different application domains.

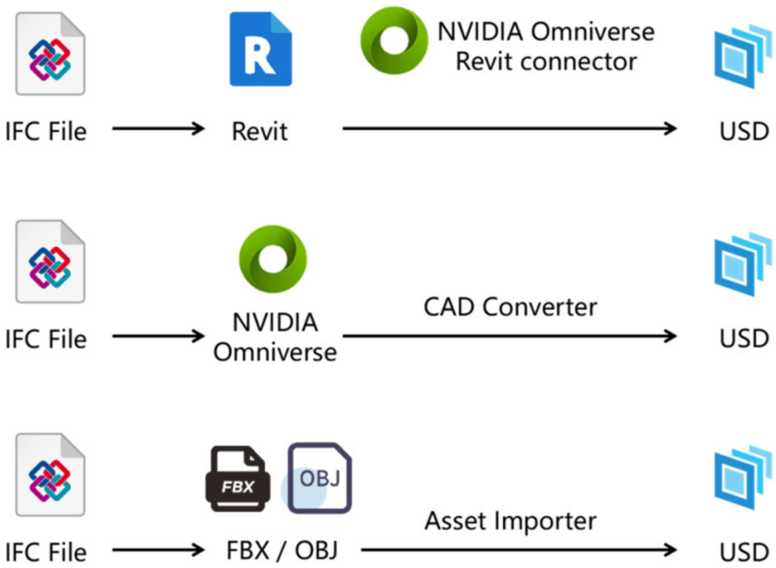


Figure 2. Three pathways for converting AEC data schema (IFC) to USD.

A small body of research addresses the real-time collaborative transformation from BIM to open scene description [37–41]. A common workflow first uses a modeling environment's IFC import to map building elements to the host category and attribute system and adjust the geometric and semantic information according to delivery requirements [42]. It employs an export interface (e.g., the Omniverse Revit Connector) to translate spatial hierarchy, geometry/materials, and attribute semantics into USD primitives and layered files, enabling cross-tool collaboration [43]. The main challenges are semantic loss and incomplete mappings, where custom attributes and parametric relations often degrade; structural bloat and interaction latency in large models; and visual inconsistencies introduced by repeated format conversions [44]. Omniverse supports the import of IFC files and converts them into the USD. However, the conversion maps geometric semantics, resulting in the loss of appearance and material information.

Converting IFC via FBX or OBJ into USD preserves meshes but loses BIM semantics. CSG or B-rep geometry is tessellated; so, curved and constructive forms may be simplified. Since FBX and OBJ focus on visual data and carry little standardized domain metadata, IFC properties, classifications, and spatial hierarchies are often dropped or flattened before reaching USD.

Overall, the fusion from an engineering data-exchange paradigm to a general scene-description paradigm remains under exploration. Current practice relies on multi-step intermediaries, making it difficult to maintain semantic continuity and collaborative consistency across the end-to-end pipeline. Accordingly, this paper investigates the semantic mapping between engineering data schemas and open scene description, with studies spanning specification representation and model quality verification, data extraction, semantic mapping and transformation, geometric conversion, and appearance material mapping.

Existing IFC-to-USD workflows can be broadly categorized as connector-based, platform-mediated, or intermediate-format pipelines. Connector-based routes rely on the host authoring tool's category system and often re-embed semantics into proprietary parameter fields (Figure 3). Platform-mediated routes provide convenient IFC and CAD ingestion but primarily map geometric semantics, with limited control over which contractually required attributes are preserved. Intermediate pipelines that tessellate IFC into FBX/OBJ before importing into USD-centric tools maximize visual compatibility but discard IFC property sets and spatial hierarchies. In contrast, the present work positions filtering and model quality checks at the head of the pipeline and then performs a direct schema-aware mapping from IFC to USD while separating the model geometry from the GlobalId-indexed JSON attributes. This design aims to maintain delivery semantics and verification context across the full scene lifecycle, rather than treating USD as a geometric export format.

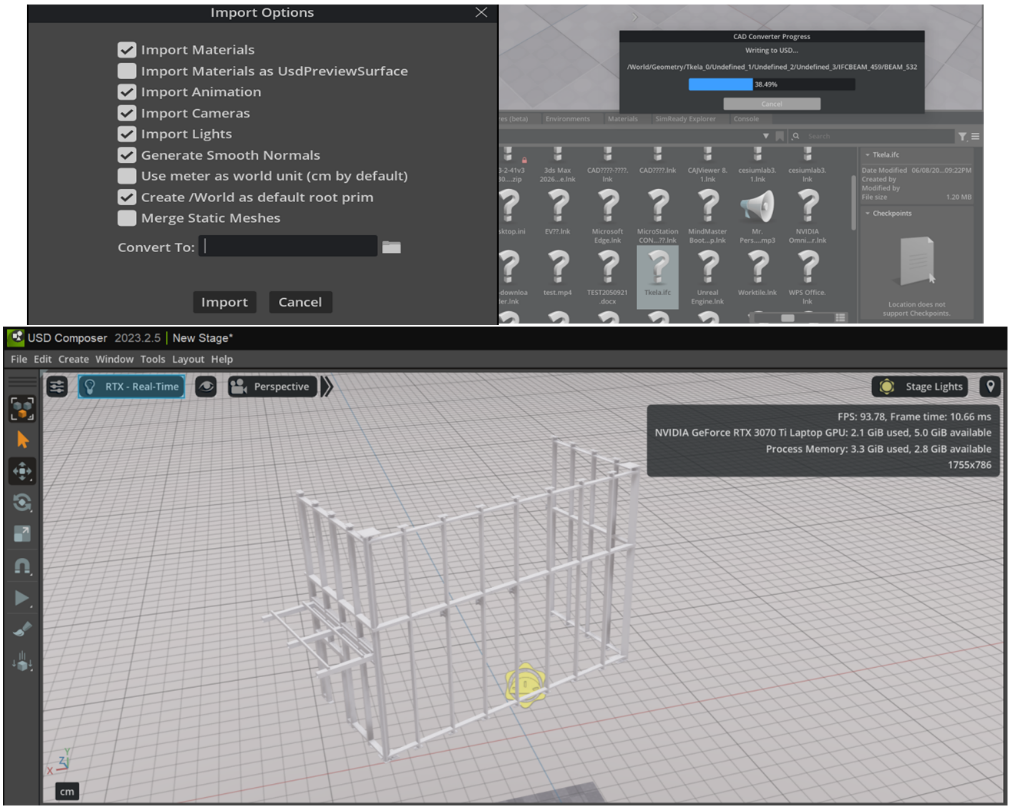


Figure 3. Example using existing converting tools.

3. Methodology

This paper focuses on establishing a bidirectional method for semantic alignment and scene composition (Figure 4). The core elements are mapping entities and relations to an extensible schema, harmonizing attributes and measurements, preserving assembly and instancing structures, and enabling traceable representations. Building on these, the pipeline (Figure 5) advances AEC practice from file-level exchange to scene-level interoperability and collaboration. Instead of treating USD as a simple export target, the USD stage is derived from and remains explicitly linked to the governing information-delivery specification.

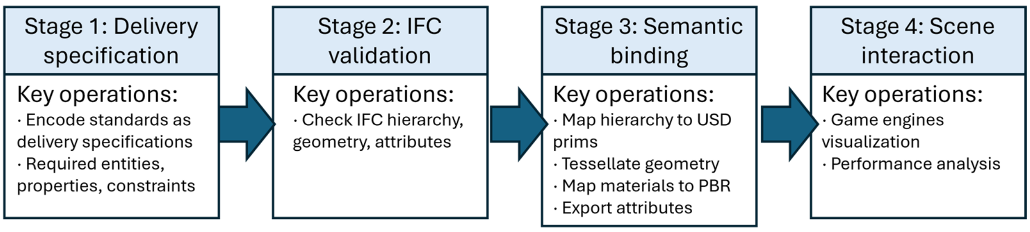


Figure 4. Specification-driven pipeline in this study.

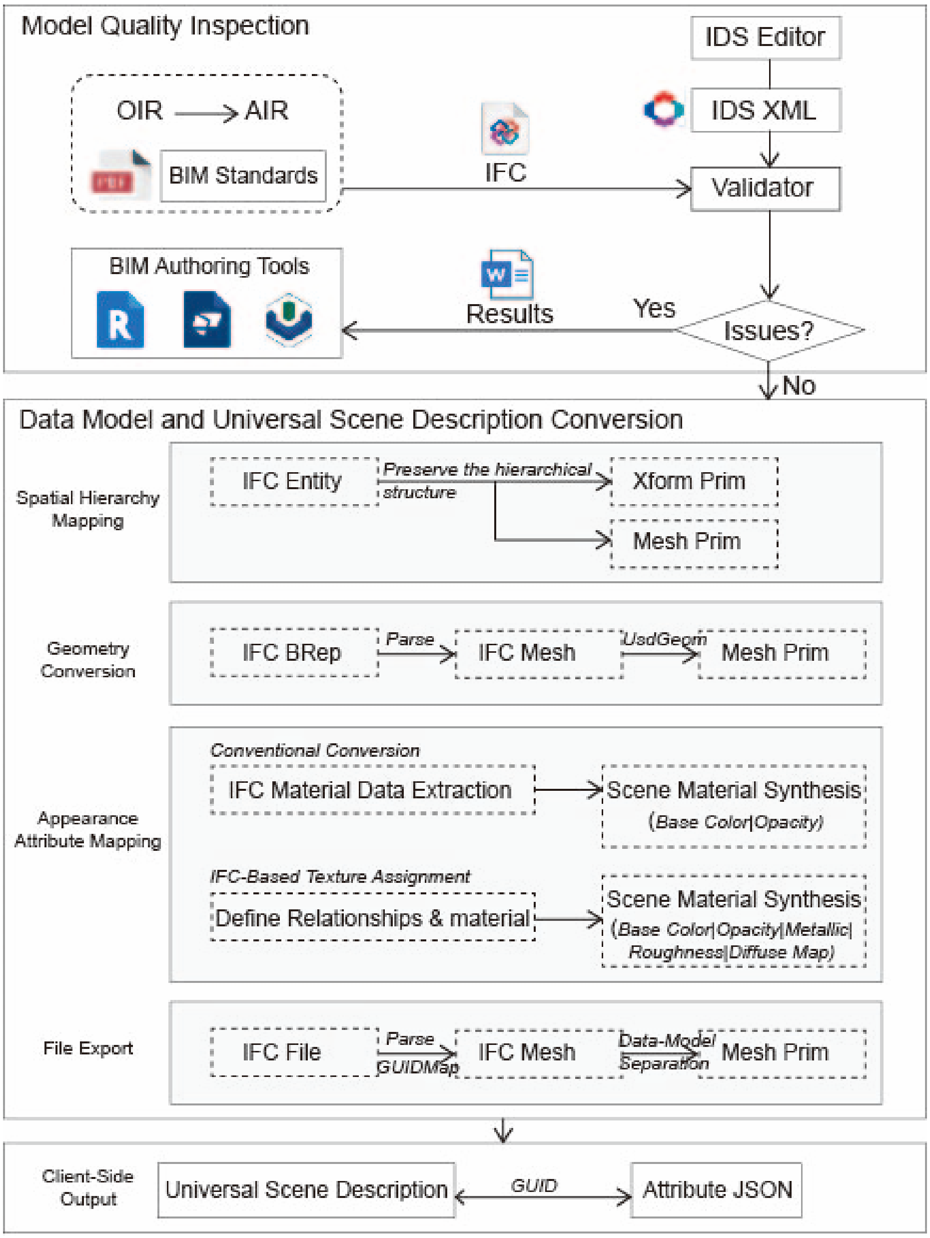


Figure 5. IFC2USD methods used in this study.

A compliant data model is established by formalizing information-delivery requirements into machine-checkable specifications and running checks on engineering models. Under this compliance premise, engineering semantics are projected into scene semantics: spatial hierarchy and element relationships are preserved as a layered structure; geometry is normalized to mesh-based scene primitives; appearance properties are mapped to material and shading descriptions; and a model–data separation strategy maintains attribute data as external structured files linked back to the scene.

3.1. Data Models and Specification Expressions for 3D Scene Delivery

Information Delivery Specification (IDS), developed by buildingSMART International (bSI), is an open-standard approach that defines information requirements for building and infrastructure projects in a machine-readable form, ensuring that delivered IFC data meet specified quality and completeness criteria [7]. Traditional verification relied on manual comparison against BIM data standards in Excel or PDF, which is inefficient and error-prone. By offering a structured computer-interpretable specification, IDS explicitly defines the required entity types, properties, classifications, and constraints that must appear in the deliverables, enabling automated validation. The core components of IDS include metadata, description, applicability, and requirements.

This study validates IFC as building deliverables using IDS. The validation workflow comprises drafting BIM standards, defining the delivery specification, creating and parsing IFC models, executing automated checks, and generating results. During the IDS definition phase, new information requirements are specified, and corresponding validation rules are established. In accordance with project contracts or industry standards, the IFC entity types, property sets, and property constraints to be checked are defined. As a concrete example from the school electrical IFC model used in this study, the IDS specification includes several effective rules: (i) all physical equipment elements must have a non-empty classification code; (ii) all cable trays (IfcCableCarrierSegment) must have a non-empty Name field; and (iii) the property on cable trays must take a value from a fixed enumeration (e.g., {Main, Branch}) and match the declared value type. Only IFC files that satisfy these and related rules are accepted for subsequent scene composition, so that the USD stage builds on data that have already passed defined information-delivery checks.

BIM standards are promulgated by public authorities or asset owners, and project teams execute modeling and downstream applications in reference to these documents. Rule sets are derived from the standards, either authored manually or configured in dedicated tools, and exported for machine use. As IDS presumes syntactically valid schema-conformant IFC as input and does not perform baseline model validation, IFC deliverables should be generated by certified software prior to IDS-based quality checks. This ensures that the deliverables meet the requirements for semantic completeness, spatial hierarchy, geometric validity, and compliance with applicable specifications.

3.2. Semantic Binding

3.2.1. Spatial Hierarchy Mapping

Mapping BIM semantics to USD is essential for transforming BIM data into visualization and interactive scenarios. The core challenge is reconciling the engineering rigor of building information models with the flexibility required for real-time rendering, bridging heterogeneous data models to achieve interoperability.

In the proposed pipeline, IFC's project breakdown structure (IfcProject, IfcSite, IfcBuilding, IfcBuildingStorey, IfcElement) is preserved and mapped one-to-one to a USD hierarchy. IFC entities without geometry are mapped to Xform prims, while geometry-bearing entities are mapped to a combination of Xform prim and Mesh prim. The Xform prim organizes and controls the scene hierarchy; the Mesh prim stores the geometry and material bindings.

Dependency relationships between elements are also reflected. For example, doors and windows in IFC are hosted by walls via IfcOpeningElement. In the USD hierarchy, after mapping an IfcWall to its Xform and Mesh prims, the corresponding IfcOpeningElement is mapped to its own Xform prim at the same level as the wall's geometric prim, and the IfcDoor is placed as a child Xform + Mesh under that opening node.

As illustrated in Figure 6, the semantic mapping and scene composition result in a USD prim structure where the *IfcWallStandardCase* "Basic Wall" Xform Prim has, as its immediate children, one Body Mesh Prim and four *IfcOpeningElement* Xform Prims. Expanding any *IfcOpeningElement* Xform Prim reveals its child level containing the IfcDoor's Xform Prim and Mesh Prim.

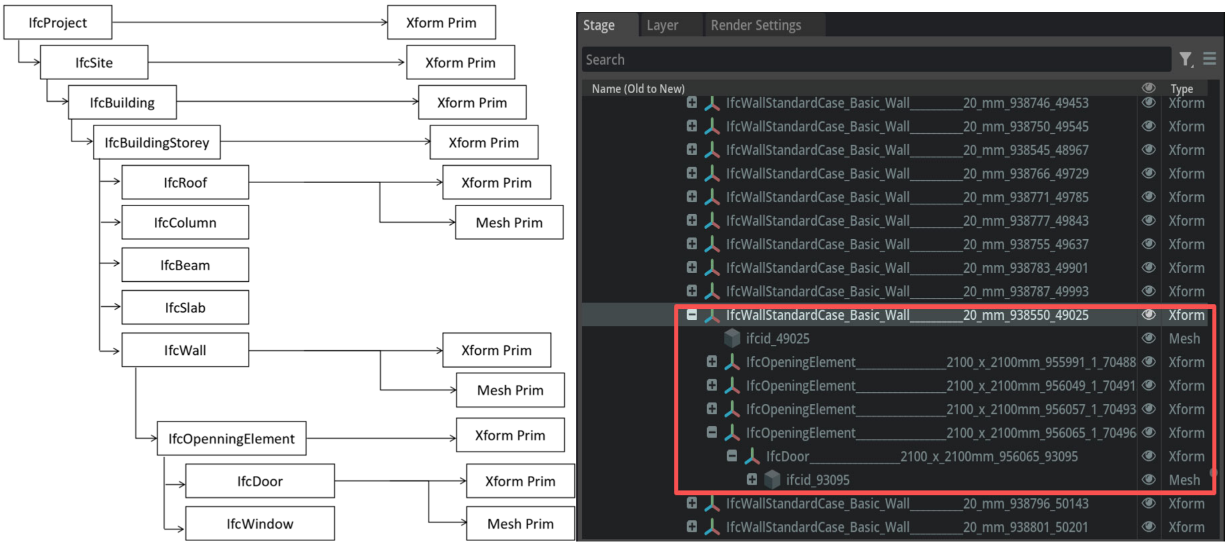


Figure 6. IFC to USD structure mapping.

3.2.2. Geometric Information

The IFC standard constructs 3D solid models using approaches such as boundary representation, swept representations, and constructive solid geometry. For geometry-related content, USD represents scene elements as polygonal meshes with associated transforms and attributes. In the proposed pipeline, IFC geometry is tessellated into triangle meshes and written as USD Mesh prims, while keeping the overall element extents and shapes consistent with the source model. Tessellation parameters are selected to balance the curvature fidelity and data volume, so that small and curved components (e.g., door handles) are preserved without noticeable artefacts, but the resulting USD meshes remain lightweight enough for real-time use. The focus is on ensuring that the geometric representation in USD remains compatible with real-time engines, while preserving the semantics of the IFC elements at the level required by the information-delivery specification, rather than on the specifics of any particular meshing library or API.

3.2.3. Appearance Attribute Mapping

IFC provides a material representation that covers geometric materials and textures while maintaining traceability. Elements are linked to material definitions via associations that can describe layered assemblies, parametric profiles, mixed-material constituents, or single-material assignments. Rendering styles are attached through IfcStyledItem, and for the scenarios used in this study, we focus on surface styles (IfcSurfaceStyle) that define colors and transparency for visible model surfaces.

USD, by contrast, employs a physically based rendering (PBR) material model in which appearance is described by a small set of parameters such as base color, metallic, roughness, and opacity. In the proposed pipeline, IFC surface-style information is mapped to these USD PBR descriptors, so that basic appearance semantics are transferred consistently across engines (Figure 7). Because most AEC authoring workflows do not export full texture maps into IFC, the mapping concentrates on commonly used attributes (e.g., surface color and transparency) and provides a reusable bridge from IFC material constructs to USD materials.

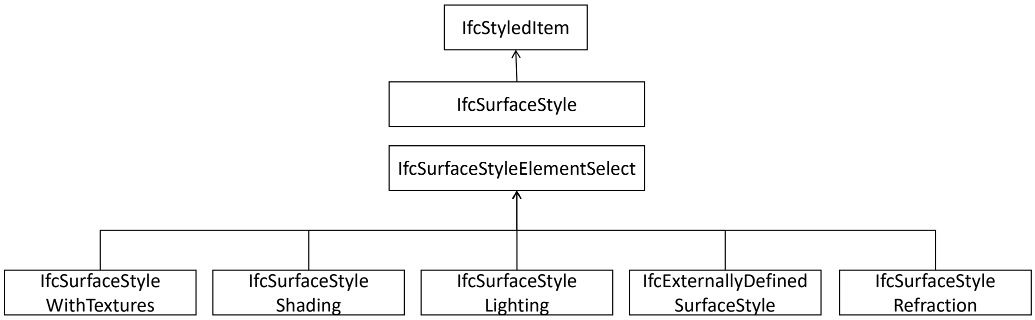


Figure 7. Geometric rendering materials and textures are associated with IfcStyledItem.

(1) Material Mapping Basics

Within *IfcSurfaceStyleShading*, *IfcSurfaceStyleRendering* defines additional material attributes, which are mapped to the predefined properties of the USD Material prim (Figure 8). The mapping rules from IFC model to USD appearance properties are as follows:

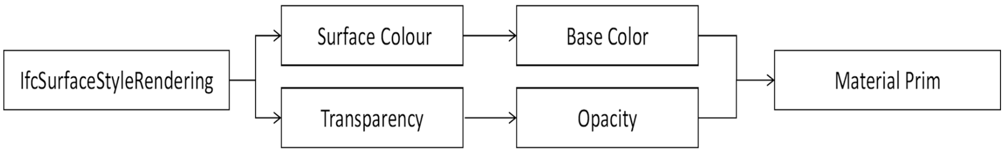


Figure 8. Material attributes mapped to the USD Material prim.

- The base color of a Material prim defines the material's fundamental color. In the IFC, the *SurfaceColour* attribute under *IfcSurfaceStyleRendering* specifies the RGB color used for surface rendering. When converting from IFC to USD, the RGB values of *SurfaceColour* in *IfcSurfaceStyleShading* can be mapped to the base color RGB values of the Material prim.
- Transparency is a key visual attribute that determines an object's see-through level (commonly for glass). The field ranges from 1.0 (fully transparent) to 0.0 (fully opaque); if unspecified, the default is 0.0 (opaque). During the process, the transparency value defined in *IfcSurfaceStyleRendering* is used to set the Material prim's opacity property, preserving the intended appearance.
- For other predefined Material prim properties, such as Metallic and Roughness, default values of 0.0 and 0.5 are assigned, respectively.

(2) Texture Assignment

Material mappings are defined by the IFC class, with support for rules derived from data standards. Based on an IFC class and specific attribute fields, parameters such as the base color, transparency, metallic, roughness, and texture maps can be specified. Different textures and maps are assigned for different IFC classes. To meet project visualization requirements, users may author an .mtlx file that declares parameters and texture references. In this work, base color, specular_color, metalness, and diffuse_roughness with their texture files are mapped to the USD Material prim's Base Color, Opacity, Metallic, Roughness, and Diffuse Map (Figure 9). Materials are bound per class by attaching them to the material:binding relationship on the target Mesh prims, thereby specifying which material is applied to each mesh.

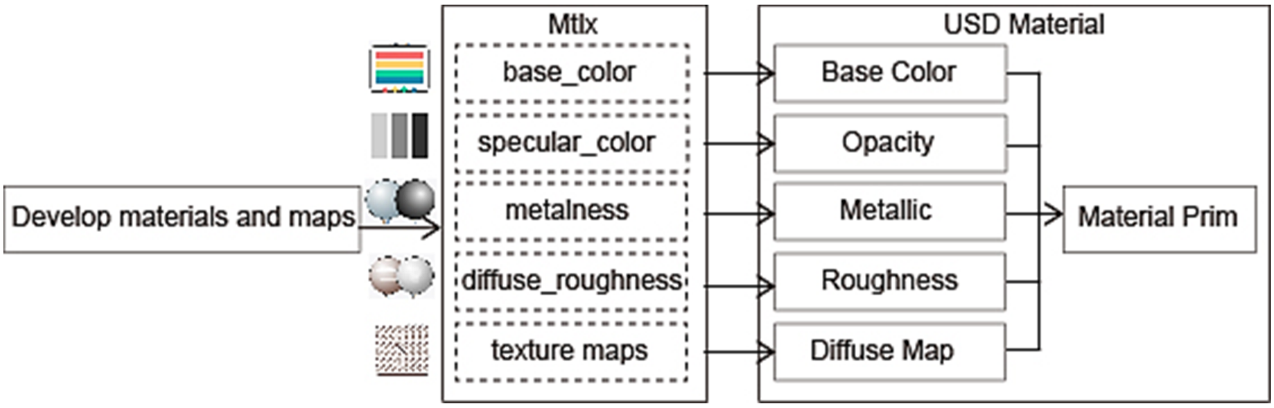


Figure 9. Develop materials and maps to USD.

3.2.4. Attribute Binding

Considering that attributes may be lost when USD data are integrated with other systems and that some third-party systems do not support reading USD attributes, this work adopts a model–data separation strategy to balance the attribute completeness and engineering flexibility for downstream integration. In IFC, the GlobalId is inherited from the root entity and is expressed as a GUID with global uniqueness, serving as the persistent identifier across projects, software platforms, and lifecycle stages. It ensures global traceability and is the basis for object identification in data exchange. IFC entities are bound to USD prim nodes, while preserving each element's GlobalId. Semantic attributes are then extracted and written to structured JSON files, achieving model–data separation that is well-suited for data analytics, cross-platform exchange, and digital-twin applications.

Using IfcOpenShell, the IFC file is loaded, and all element entities are iterated to extract basic attributes such as the *GlobalId*, *Name*, and *ObjectType*. In accordance with the information delivery specification, the process traverses the *IfcRelDefinesByProperties* relationships, matches property-set names, extracts the corresponding properties, and exports them to a JSON file. During the geometric conversion described earlier, *GlobalId* and Name are written to both the *Xform* prim and the Mesh prim to provide persistent identifiers for subsequent cross-platform exchange or system-integration development. Downstream applications can therefore synchronize USD geometry with the external JSON attributes by joining on the shared GlobalId, allowing attribute updates to be propagated by editing the JSON records while keeping the USD geometry unchanged. In summary, IFC hierarchy information is bound to USD Xform prims, geometric representations to Mesh prims, material and surface styles to Material prims, while semantic attributes are exported to external JSON files indexed by GlobalId.

3.3. Prototype Design

The development in this study adopts a client–server architecture and is organized into three layers (Figure 10): the resource layer, the algorithm layer, and the application layer. The resource layer comprises the imported IFC model data and material definition files, as well as the exported USD models and attribute files. The algorithm layer includes IFC data ingestion, semantic-mapping rules, scene composition implemented via the USD API, and *GlobalId*-indexed attribute export. The application layer supports path configuration, executes model conversion, and visualizes conversion progress. The overall architecture is shown in the figure below.

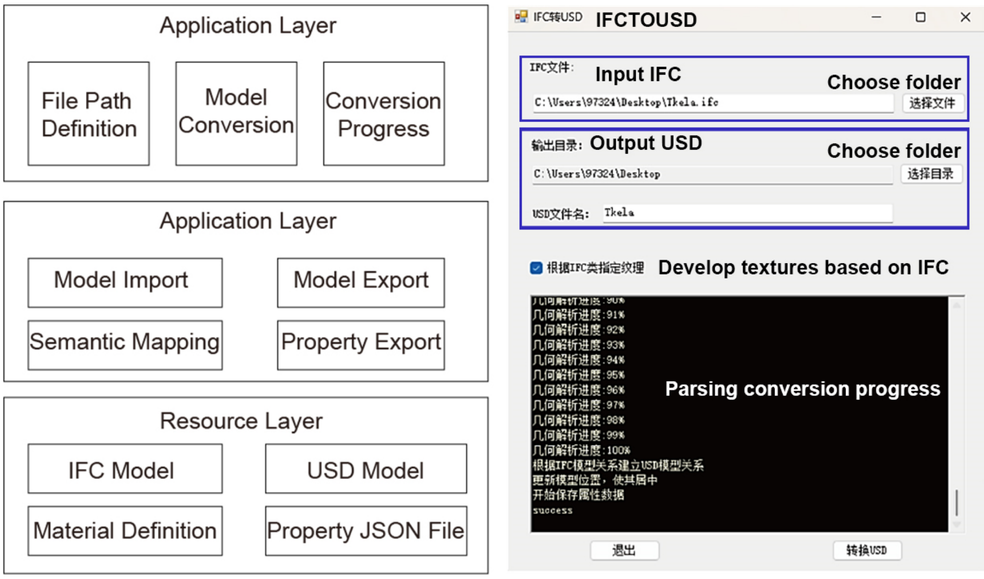


Figure 10. Client–server architecture and tool developed in this study.

4. Experiment

This study is grounded in the Technical Guidelines for Modeling and Delivery of Existing Significant Buildings—Building Engineering Volume issued by the Shenzhen Housing and Construction Bureau, which specify component naming, classification codes, component numbering, coordinate requirements, delivery accuracy for component-level model units, and component-level attribute tables for the target delivery scenarios Multiple software models are converted and tested to assess feasibility and implementation effectiveness. Tests are conducted on IFC models exported from Revit and Tekla across civil, MEP, and steel-structure disciplines, with analyses focusing on the visual quality, model size, and attribute completeness (Tables 1 and 2 and Figure 11).

Table 1. Data used in case study experiment.

No	Project Type	Discipline	Model Size	Entity Count
1	Commercial Building	Civil Work	20.80 MB	1203
2	School	Electrical	4.00 MB	1314
3	An Airport Runway	Structure	1.19 MB	1005

Table 2. Component-level model unit attribute information table based on Shenzhen Engineering Construction Delivery Standard (● = required, ○ = optional).

No.	Info Category	Info Name	Detailed Design	Unit
1	Identity Info	Name	●	/
2	Identity Info	Code	●	/
3	Location Info	Building Unit Name	●	/
4	Location Info	Floor	●	/
5	Location Info	Space Name	●	/
6	Location Info	Base Point Coordinate X	●	m
7	Location Info	Base Point Coordinate Y	●	m
8	Location Info	Base Point Coordinate Z	●	m

Table 2. *Cont.*

No.	Info Category	Info Name	Detailed Design	Unit
9	Location Info	Footprint Length	●	mm
10	Location Info	Footprint Width	●	mm
11	Location Info	Footprint Height	●	mm
12	System Info	System Category Level 1	●	/
13	System Info	System Category Level 2	●	/
14	System Info	System Category Level 3	●	/
15	Technical Info	Model/Specification	●	/
16	Technical Info	Fire Rating	●	/
17	Technical Info	Enclosure Protection	●	/
18	Technical Info	Material	●	/
19	Technical Info	Mass	●	kg
20	Technical Info	Installation Method	●	/
21	Technical Info	Installation Height	●	m
22	Manufacturing Info	Manufacturer Name	○	/
23	Manufacturing Info	Product Standard	●	/
24	Manufacturing Info	Certification Scheme	●	/
25	Manufacturing Info	Date of Manufacture	○	/
26	Manufacturing Info	Ex-Factory Price	○	/

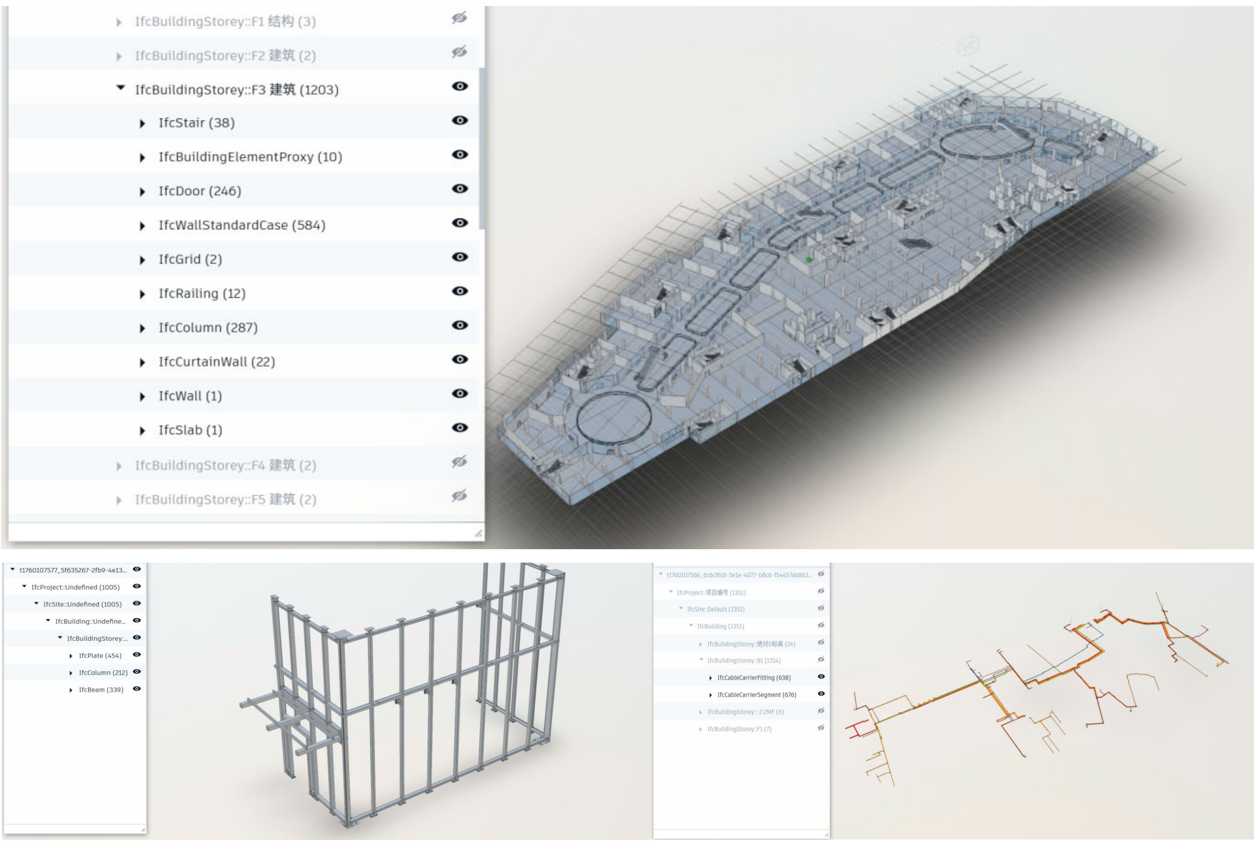


Figure 11. Model used in case study experiment using IFC model viewer.

4.1. Model Checking

Before scene generation, a machine-verifiable quality check is performed on the IFC models intended for 3D scene delivery based on IDS, providing trustworthy semantics for scene composition. The IDS and IFC files are loaded to run entity scanning and property validation, identifier consistency (unique GlobalId, compliant naming), spatial hierarchy, geometric validity (closed solids, coherent units and coordinates), and standards conformance, which matches to IDS expressions or enumerations, as well as materials and properties meeting specified rules. Files that pass are labeled and proceed to the next stage with required data locked for USD semantic mapping and binding.

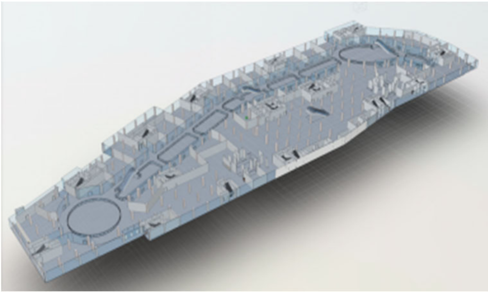
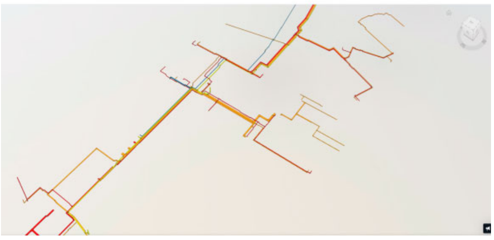
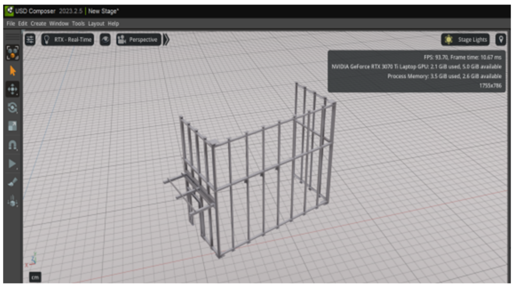
4.2. Visual Performance

Table 3 summarizes the visual results obtained by mapping IFC models to USD with the proposed system. IFC models were inspected in Autodesk Viewer [45], and the generated USD scenes were viewed in NVIDIA Omniverse [46] and Unreal Engine 5 [47]. Surface color and transparency values were extracted from IFC and mapped to PBR materials using the USD Preview Surface model. In addition to overall visual inspection, targeted checks were performed on geometric and material fidelity. For the commercial building model, small and curved elements such as door handles and locks showed no mesh breakage or visibly faceted circular parts, indicating tessellation quality sufficient for the intended application scenarios. For glass elements, IFC parameters (e.g., Surface Color = (0, 128, 192), Transparency = 0.9) were mapped to an equivalent base color and opacity in USD, and side-by-side rendered views showed visually faithful reproduction.

The system supports defining material-mapping rules by IFC class, allowing the mappings to be specified according to data the standards and visual-platform requirements. Users can customize materials and textures and bind them by class. In the steel-structure test scenario, a brushed stainless-steel preset (from an AMD GPU Open MaterialX library, provided as a .mtlx file with base_color (0.53, 0.51, 0.50), metalness = 1, diffuse_roughness = 0.5) was assigned to columns and beams and mapped to USD Preview Surface shaders (Figure 12). These class-based mappings demonstrate that the pipeline preserves the spatial hierarchy, geometry, and appearance attributes in the USD scenes without observable geometric degradation.

Figure 12. Texture maps and rendering effective of the USD model.

Table 3. Data models and scene composition results.

Model	IFC Model in AUTODESK Viewer	USD in Omniverse/Unreal Engine
Civil Works Model		
MEP Model		
Tekla Steel Structure Model		

4.3. Model Volume

.usda is a readable text format, convenient for review and debugging but larger and slower to read and write. It is an optimized binary format—smaller and faster to load, suitable for asset publishing and large scenes; *.usd* is a generic suffix that can denote either ASCII or binary encoding. In this work, the system exports *.usd* for digital-asset publishing and BIM platform applications; for manual inspection, the binary layer can be converted to *.usda* text with supporting tools for viewing and fine-tuning.

After semantic mapping and scene composition, the resulting USD datasets were substantially reduced in size. While preserving the intended semantic structure, the pipeline employs binary encoding to achieve smaller files and faster I/O and exports attribute data separately to further decrease USD payloads. Because the dataset size is coupled to the use context, scene composition should select an encoding strategy appropriate to the application, balancing the readability, file size, and read and write performance to achieve optimal results while expressing the IFC semantics. To provide a lightweight indication of conversion performance, three representative case-study models were also converted using four different IFC-to-USD workflows (IFC → Revit → USD, IFC → CAD Converter → USD, IFC → FBX/OBJ → USD, and the proposed IFC → IFC2USD pipeline). All tests were executed on the same workstation (12th Gen Intel Core i9-12900H CPU, NVIDIA GeForce RTX 3070 Ti Laptop GPU with 8 GB VRAM, 16 GB RAM). The proposed pipeline achieves comparable or lower end-to-end conversion times than the multi-stage alternatives, while avoiding intermediate exchange formats and preserving IDS-validated identifiers and attributes for downstream use.

4.4. Model Properties

The system preserves each element's unique identifier GlobalId (GUID), extracts semantic attributes with precision, and generates structured JSON data, making the model–data separation suitable for data analytics, cross-platform exchange, and R&D on visual platforms. Using Microsoft Visual Studio, the exported JSON files can be loaded to inspect per-element attributes and compared against the GUID and associated properties shown in Viewer for the IFC file. For the school electrical model, the exported JSON attributes were cross-checked against the IFC metadata for the required property sets defined by the project's delivery specification, and no missing fields were observed for the targeted business attributes. In parallel, spot checks on the USD Xform hierarchy confirmed that the project–site–building–story–element chain inherited from IFC was preserved for the sampled elements. The JSON output includes GUID, ObjectType, identification data, dimensional annotations, model properties, constraints, and related attributes. For example, in the school electrical model, each JSON record starts with a small header containing 'IfcId', 'GlobalId', 'Name', and 'ObjectType', followed by grouped key–value pairs for the relevant property sets. The system preserves the attribute integrity while providing flexibility for subsequent engineering application.

4.5. Comparison with Existing IFC → USD Methods

In addition to runtime and file-size measurements (Tables 4 and 5), the comparison in this section also treats semantic retention as a small benchmark across five criteria—identifier preservation, hierarchy consistency, geometric completeness, material mapping, and attribute retention—using a three-level scale for each workflow. Existing IFC to USD workflows provide good geometric translation but only partial semantic retention. In the BIM to USD route, the IFC file is first reinterpreted by software and then exported; so, attribute fields and parts of the original IFC hierarchy may be dropped or remapped to tool-specific categories, and the resulting USD scene follows the software's internal structure rather than the delivery hierarchy. The Omniverse CAD Converter to USD path preserves the overall project–site–building–story structure and is convenient for mixed CAD/IFC ingestion, but it does not fully use IFC's native hierarchy, introduces generic nodes, and, in the version tested for this study, does not export attributes; appearance semantics are also incomplete, as glass elements in the building model retained only the RGB color while losing IFC transparency due to USD opacity mapping. The IFC to FBX/OBJ to USD path relies on intermediate formats that were designed for mesh exchange; so, component hierarchies are flattened, properties and classifications are discarded, and materials must be re-authored or provided via separate files, resulting in limited semantic continuity and lower overall efficiency. By contrast, the proposed IFC2USD tool starts from delivery specification-validated IFC, maps the spatial hierarchy one-to-one to USD Xform prims, derives Mesh prims and PBR parameters from IFC surface styles, and exports attributes as GlobalId-indexed JSON instead of embedding them in the scene. As summarized, the schema-aware pipeline achieves comparable or lower conversion times than the multi-stage workflows while maintaining a more complete set of delivery semantics for downstream use. In qualitative terms, this comparison acts as a small benchmark across identifier preservation, hierarchy consistency, geometric completeness, material mapping, and attribute retention, where only the proposed IFC2USD route maintains all five aspects simultaneously for the case-study models.

Table 4. Data size comparison for different conversion workflows.

Project Type	Software	Discipline	IFC Model Size	IFC → USD Workflows USD Size			
				IFC → Revit → USD	IFC → Omniverse CAD Converter → USD	IFC → FBX/OBJ → USD	IFC → IFC2USD → USD
Commercial Building	Revit	Civil Work	20.80 MB	10.5 MB	14.0 MB	8.7 MB	11.4 MB
School	Revit	Electrical	4.00 MB	0.4 MB	1.62 MB	0.9 MB	1.24 MB
An Airport Runway	Tekla	Structure	1.19 MB	0.7 MB	0.6 MB	1.9 MB	0.5 MB

Table 5. Performance before and after conversion for different models.

Type	Discipline	IFC → USD Workflows			
		IFC → Revit → USD(s)	IFC → Omniverse CAD Converter → USD(s)	IFC → FBX/OBJ → USD(s)	IFC → IFC2USD → USD(s)
Commercial Building	Civil Work	37	38	12	10
School	Electrical	17	9	5	4
Airport Runway	Structure	7	4	7	3

5. Conclusions

openBIM addresses data standards and semantic consistency within AEC deliverables, whereas OpenUSD excels at cross-domain cross-tool collaboration and efficient composition of complex 3D scenes. Fusing the two enables interoperability that goes beyond geometry, allowing BIM data to be transmitted and utilized in 3D environments in accordance with information-delivery requirements. Recent industry developments, including the liaison between IFC and USD and early IFC5 work that explores USD-inspired mechanisms, suggest a complementary pattern of IFC as the authoritative data and USD as the operational scene; the proposed pipeline provides a project-level realization of this direction for digital-twin and XR workflows.

This study develops a specification-driven IFC to USD pipeline that treats information-delivery requirements as a scene contract and prioritizes semantic continuity. IDS-validated IFC models are checked and normalized and then mapped into a USD scene graph, where spatial hierarchy is preserved via Xform prims, geometry is expressed as Mesh prims, PBR appearance is derived from IFC surface styles, and semantic attributes are externalized as GlobalId-indexed JSON. Across three representative case-study models and multiple workflows, the results show that this schema-aware route achieves comparable or lower conversion times, produces lighter USD payloads under binary encoding, and retains identifiers, hierarchies, geometry, materials, and attributes more completely than connector and converter-based alternatives. In this sense, the research questions of whether delivery specifications can drive semantically continuous scenes and whether this can be achieved with practical performance for AEC/XR workflows, are positively answered at the prototype scale.

Several limitations remain. The current evaluation is restricted to three medium-scale models from civil, electrical, and steel-structure disciplines and to a unidirectional IFC to USD flow; larger multi-discipline federated models and round-tripping back to IFC will require further integration with authoring tools. Dynamic scene streaming, multi-user editing, and version control across USD layers are also not yet addressed. Future work will therefore extend testing to larger and more diverse datasets and BIM platforms, integrate MaterialX/OpenPBR libraries more systematically, and deploy the pipeline within digital-twin platforms to assess its behavior under continuous update and operational use.

Author Contributions: Data curation, C.H.; Formal analysis, G.R. and C.H.; Funding acquisition, G.R.; Methodology, C.H., G.R. and T.S.; Project administration, G.R. and T.S.; Resources, G.R. and C.H.; Supervision, G.R. and T.S.; Validation, C.H. and G.R.; Writing—original draft, C.H., G.R. and T.S.; Writing—review and editing, G.R. and T.S. All authors have read and agreed to the published version of the manuscript.

Funding: This project is supported by the National Natural Science Foundation of China (Project No.:52308031), the sub-project of the National Key R&D Program (Project No.: 2024YFC3809903), and the Fundamental Research Funds for the Central Universities of China.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: There are no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Jeon, K.; Lee, G.; Kang, S.; Roh, H.; Jung, J.; Lee, K.; Baldwin, M. A relational framework for smart information delivery manual (IDM) specifications. *Adv. Eng. Inform.* **2021**, *49*, 101319. [CrossRef]
2. Zhang, H.; Zhao, W.; Han, Z.; Zhu, J.; Zhu, Q.; Xu, Z.; Feng, D.; Song, Y.; Song, S.; Zhang, B.; et al. Knowledge-guided digital twin modeling method of generating hierarchical scenes for a high-speed railway. *Trans. GIS* **2023**, *27*, 2017–2041. [CrossRef]
3. Rehman, S.U.; Kim, I.; Hwang, K.-E. Advancing BIM and game engine integration in the AEC industry: Innovations, challenges, and future directions. *J. Comput. Des. Eng.* **2025**, *12*, 26–54. [CrossRef]
4. Jiang, Y.; Liu, X.; Wang, Z.; Li, M.; Zhong, R.Y.; Huang, G.Q. Blockchain-enabled digital twin collaboration platform for fit-out operations in modular integrated construction. *Autom. Constr.* **2023**, *148*, 104747. [CrossRef]
5. Qiu, Q.; Lau, D. Real-time detection of cracks in tiled sidewalks using YOLO-based method applied to unmanned aerial vehicle (UAV) images. *Autom. Constr.* **2023**, *147*, 104745. [CrossRef]
6. Shi, J.; Pan, Z.; Jiang, L.; Zhai, X. An ontology-based methodology to establish city information model of digital twin city by merging BIM, GIS and IoT. *Adv. Eng. Inform.* **2023**, *57*, 102114. [CrossRef]
7. Information Delivery Specification IDS—BuildingSMART Technical. Available online: https://technical.buildingsmart.org/projects/information-delivery-specification-ids/ (accessed on 12 October 2024).
8. Son, S.; Lee, G.; Jung, J.; Kim, J.; Jeon, K. Automated generation of a model view definition from an information delivery manual using idmXSD and buildingSMART data dictionary. *Adv. Eng. Inform.* **2022**, *54*, 101731. [CrossRef]
9. Cerovšek, T.; Omar, M. Advancing Semantic Enrichment Compliance in BIM: An Ontology-Based Framework and IDS Evaluation. *Buildings* **2025**, *15*, 2621. [CrossRef]
10. de Marco, G.; Slongo, C.; Siegele, D. Enriching Building Information Modeling Models through Information Delivery Specification. *Buildings* **2024**, *14*, 2206. [CrossRef]
11. Agbossou, I. *Urban Built Environment Visual Features Modeling for 3D GeoSimulation Using USD Standard Specifications BT—Innovation in Urban and Regional Planning*; Marucci, A., Zullo, F., Fiorini, L., Saganeiti, L., Eds.; Springer: Cham, Switzerland, 2024; pp. 169–182.
12. PIXAR. USD White Papers. 2021. Available online: https://openusd.org/docs/USD-3rd-Party-Plugins.html (accessed on 1 January 2025).
13. Apple Inc. Pixar, Adobe, Apple, Autodesk, and NVIDIA form Alliance for OpenUSD to Drive Open Standards for 3D Content. 2023. Available online: https://www.apple.com.cn/newsroom/2023/08/pixar-adobe-apple-autodesk-and-nvidia-form-alliance-for-openusd/ (accessed on 1 December 2025).
14. Tang, M.; Nikolaenko, M.; Alrefai, A.; Kumar, A. Metaverse and Digital Twins in the Age of AI and Extended Reality. *Architecture* **2025**, *5*, 36. [CrossRef]
15. Benning, P.; Dumoulin, C. Interoperability Through Standards: IFC, Concepts and Methods. In *Building Information Modeling: Shared Modeling, Mutual Data, the New Art of Building*; Wiley: Hoboken, NJ, USA, 2023; pp. 51–75. [CrossRef]
16. Yu, Y.; Kim, S.; Jeon, H.; Koo, B. A Systematic Review of the Trends and Advances in IFC Schema Extensions for BIM Interoperability. *Appl. Sci.* **2023**, *13*, 12560. [CrossRef]
17. Okonta, E.D.; Vukovic, V.; Hayat, E. Prospective Directions in the Computer Systems Industry Foundation Classes (IFC) for Shaping Data Exchange in the Sustainability and Resilience of Cities. *Electronics* **2024**, *13*, 2297. [CrossRef]
18. BuildingSMART. BuildingSMART Technical. 2021. Available online: https://technical.buildingsmart.org/ (accessed on 1 December 2025).

19. Di Biccari, C.; Calcerano, F.; D'Uffizi, F.; Esposito, A.; Campari, M.; Gigliarelli, E. Building information modeling and building performance simulation interoperability: State-of-the-art and trends in current literature. *Adv. Eng. Inform.* **2022**, *54*, 101753. [CrossRef]
20. Forth, K.; Borrmann, A. Semantic enrichment for BIM-based building energy performance simulations using semantic textual similarity and fine-tuning multilingual LLM. *J. Build. Eng.* **2024**, *95*, 110312. [CrossRef]
21. Guntermann, L.; Hagedorn, P.; König, M. Enhancing IFC Models for Circular Design: Validation of Required Information through an IDS Framework. In Proceedings of the 42nd International Symposium on Automation and Robotics in Construction (ISARC 2025), Montreal, QC, Canada, 28–31 July 2025. [CrossRef]
22. Zhu, X. Smart Compliance Checking Frameworks for BIM Standards. Ph.D. Thesis, Cardiff University, Cardiff, UK, 2024. Available online: https://orca.cardiff.ac.uk/id/eprint/169050 (accessed on 1 December 2025).
23. Piras, G.; Agostinelli, S.; Muzi, F. Smart Buildings and Digital Twin to Monitoring the Efficiency and Wellness of Working Environments: A Case Study on IoT Integration and Data-Driven Management. *Appl. Sci.* **2025**, *15*, 4939. [CrossRef]
24. Scandurra, S.; di Luggo, A. Bsdd to Document State of Preservation of Architectural Heritage in Open-Hbim Systems. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *XLVIII-M-2*, 1427–1434. [CrossRef]
25. Horn, R.; Ebertshäuser, S.; Di Bari, R.; Jorgji, O.; Traunspurger, R.; von Both, P. The BIM2LCA Approach: An Industry Foundation Classes (IFC)-Based Interface to Integrate Life Cycle Assessment in Integral Planning. *Sustainability* **2020**, *12*, 6558. [CrossRef]
26. Aragón, A.; Alberti, M.G. Limitations of machine-interpretability of digital EPDs used for a BIM-based sustainability assessment of construction assets. *J. Build. Eng.* **2024**, *96*, 110418. [CrossRef]
27. Sarton, J.; Zellmann, S.; Demirci, S.; Güdükbay, U.; Alexandre-Barff, W.; Lucas, L.; Dischler, J.M.; Wesner, S.; Wald, I. State-of-the-art in Large-Scale Volume Visualization Beyond Structured Data. *Comput. Graph. Forum* **2023**, *42*, 491–515. [CrossRef]
28. Esser, S.; Vilgertshofer, S.; Borrmann, A. Version control for asynchronous BIM collaboration: Model merging through graph analysis and transformation. *Autom. Constr.* **2023**, *155*, 105063. [CrossRef]
29. Li, H.; Rieckenberg, G.; Karanam, N.; Vo, E.; Hurley, K. Optimizing Assets for Authoring and Consumption in USD. In Proceedings of the ACM SIGGRAPH 2024 Talks, Denver, CO, USA, 28 July–1 August 2024; ACM: New York, NY, USA, 2024; pp. 1–2.
30. Esser, S.; Vilgertshofer, S.; Borrmann, A. Graph-based version control for asynchronous BIM collaboration. *Adv. Eng. Inform.* **2022**, *53*, 101664. [CrossRef]
31. Afzal, S.; Ghani, S.; Hittawe, M.M.; Rashid, S.F.; Knio, O.M.; Hadwiger, M.; Hoteit, I. Visualization and Visual Analytics Approaches for Image and Video Datasets: A Survey. *ACM Trans. Interact. Intell. Syst.* **2023**, *13*, 1–41. [CrossRef]
32. Xu, K.; Ottley, A.; Walchshofer, C.; Streit, M.; Chang, R.; Wenskovitch, J. Survey on the Analysis of User Interactions and Visualization Provenance. *Comput. Graph. Forum* **2020**, *39*, 757–783. [CrossRef]
33. Castro, C.; Leiva, V.; Basso, F. A Data-Driven Systematic Review of the Metaverse in Transportation: Current Research, Computational Modeling, and Future Trends. *Comput. Model. Eng. Sci.* **2025**, *144*, 1481–1543. [CrossRef]
34. AOUSD. Alliance for OpenUSD. 2025. Available online: https://aousd.org/ (accessed on 1 December 2025).
35. Beach, T.; Yeung, J.; Nisbet, N.; Rezgui, Y. Digital approaches to construction compliance checking: Validating the suitability of an ecosystem approach to compliance checking. *Adv. Eng. Inform.* **2024**, *59*, 102288. [CrossRef]
36. Tuhaise, V.V.; Tah, J.H.M.; Abanda, F.H. Technologies for digital twin applications in construction. *Autom. Constr.* **2023**, *152*, 104931. [CrossRef]
37. Xu, L.; Veeramani, D.; Zhu, Z. Automated physics-based modeling of construction equipment through data fusion. *Autom. Constr.* **2024**, *168*, 105880. [CrossRef]
38. Xu, L.; Liu, H.; Xiao, B.; Luo, X.; Veeramani, D.; Zhu, Z. A systematic review and evaluation of synthetic simulated data generation strategies for deep learning applications in construction. *Adv. Eng. Inform.* **2024**, *62*, 102699. [CrossRef]
39. Kosmopoulos, P.; Dhake, H.; Kartoudi, D.; Tsavalos, A.; Koutsantoni, P.; Katranitsas, A.; Lavdakis, N.; Mengou, E.; Kashyap, Y. Ray-Tracing modeling for urban photovoltaic energy planning and management. *Appl. Energy* **2024**, *369*, 123516. [CrossRef]
40. Koprov, P.; Starly, B. Industrial metaverse meets IIoT: Low-code platforms for machine-to-machine and human-to-machine integration. *Manuf. Lett.* **2025**, *44*, 1254–1265. [CrossRef]
41. Balin, S.; Bolognesi, C.M.; Borin, P. Immersive representation exercise in BIM through the integration of prerequisites and requirements: Structuring of an artefact applied to a case study. *Procedia Comput. Sci.* **2025**, *253*, 155–166. [CrossRef]
42. Lee, Y.-C.; Solihin, W.; Eastman, C.M. The Mechanism and Challenges of Validating a Building Information Model regarding data exchange standards. *Autom. Constr.* **2019**, *100*, 118–128. [CrossRef]
43. NVIDA. Omniverse Connect. 2025. Available online: https://docs.omniverse.nvidia.com/connect/latest/revit.html (accessed on 30 September 2025).
44. Du, J.; Zou, Z.; Shi, Y.; Zhao, D. Zero latency: Real-time synchronization of BIM data in virtual reality for collaborative decision-making. *Autom. Constr.* **2018**, *85*, 51–64. [CrossRef]
45. Autodesk Viewer | Free Online File Viewer. Available online: https://viewer.autodesk.com/ (accessed on 1 December 2025).

46. Omniverse Platform for OpenUSD | NVIDIA. Available online: https://www.nvidia.com/en-us/omniverse/ (accessed on 1 December 2025).
47. Unreal Engine 5. Available online: https://www.unrealengine.com/en-US/unreal-engine-5 (accessed on 1 December 2025).