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

A CDE ECOSYSTEM FOR THE ARCHITECTURE, ENGINEERING & CONSTRUCTION (AECO) SECTOR

SUBMITTED: July 2025
PUBLISHED:February 2026
EDITOR: Frédéric Bosché
DOI: 10.36680/j.itcon.2026.006

Robert Doe, Dr
University of South Australia Creative, Adelaide, Australia
https://orcid.org/0000-0002-8246-3003
robert.doe@unisa.edu.au

Karamjit Kaur, Dr
University of South Australia STEM, Adelaide, Australia
https://orcid.org/0000-0003-0255-1060
Karamjit.kaur@unisa.edu.au

Matt Selway, Dr
University of South Australia STEM, Adelaide, Australia
https://orcid.org/0000-0001-6220-6352
matt.selway@unisa.edu.au

Markus Stumptner, Professor,
University of South Australia STEM, Adelaide, Australia;
https://orcid.org/0000-0002-7125-3289
markus.stumptner@unisa.edu.au

SUMMARY: *Building Information Modelling (BIM) workflows are increasingly managed by multiple Common Data Environments (CDE) which effectively comprise distributed sources of truth. However, interoperability of the AECO ecosystem is constrained by normative contractual relationships and by technical incompatibilities between systems and applications. Our aim is to improve interoperability in the AECO sector by introducing a system of systems (SoS) approach - a CDE ecosystem that combines standards-based messaging services with API adapters to enable object-based exchanges of information. Our research design has evolved over the past five years through literature reviews, fieldwork with industry partners, and implementation of a proof-of-concept CDE ecosystem. Engagement with AECO industry partners GHD and DBM Vircon focused research design and helped refine the research question. This paper also leverages prior research by the authors to improve interoperability in the process industry. CDE ecosystem benefits which support the AECO sector's complex interactions and distributed organisational workflows include: vendor-agnosticism, allowing organisations to choose their preferred or specialised software tools; fine-grained transactions, which anticipate IFC5 inter-sector (buildings and infrastructure) interoperability; connectivity across multiple systems; event-driven, timely and reliable, many-to-many messaging services.*

KEYWORDS: *interoperability, System of Systems (SoS), ecosystem, Common Data Environment (CDE), many-to-many, partial exchanges, fine-grained, Building Information Modelling (BIM).*

REFERENCE: *Doe, R., Kaur, K., Selway, M., & Stumptner, M. (2026). A CDE ecosystem for the architecture, engineering & construction (AECO) sector. Journal of Information Technology in Construction (ITcon), 31, 129-148. https://doi.org/10.36680/j.itcon.2026.006*

1. INTRODUCTION

This paper, and our research to date, aims to improve interoperability in the AECO sector. Here we summarise our research design and present a proof-of-concept solution, a novel approach which connects enterprises and integrates business with project processes. Presently in the AECO sector CDEs and applications are challenged to exchange information without misunderstandings, errors, or data loss occurring. A key contributor to poor interoperability is lack of integration between software vendors' products and services, as documented by (Lee, 2011, Afsari, Eastman and Shelden, 2017). The impacts of poor interoperability on the AECO sector were identified by (Gallaher *et al.*, 2004) in their report for the National Institute of Standards and Technology (NIST) which estimated costs of US$15.8 billion per annum in the U.S. sector alone. Close examination of the report by (Chapman, 2005) revealed that ineffective business processes were a significant contributory factor. As the principles of ISO 19650 with BIM workflows managed by CDEs are implemented we are again reminded of the urgent need to improve collaboration and interoperability in the AECO sector (Steel *et al.*, 2012, Sacks *et al.*, 2018). The CDE ecosystem demonstrates a distributed, many-to-many, SoS approach which is able to connect enterprise, business and project concerns. A simulated setting for the proof-of-concept was prompted by recent research (Doe *et al.*, 2024) and a key question asked by industry partners which targeted a need to connect disparate CDEs. Although its core activity is project based, the AECO sector remains fragmented and dispersed. Hence, the challenge is to connect the gaps between enterprise, business and project processes. The proof-of-concept CDE ecosystem proposes an SoS solution to bridge these gaps.

1.1 Significance

This paper contributes a novel solution by demonstrating an SoS approach where messaging services combined with API adaptors connect systems, CDEs and applications to allow for fuller utilisation and optimisation of information. The proof-of-concept solution demonstrates that many-to-many relationships are possible between stakeholders, multiple tools and services. This approach is open standards based and vendor-agnostic allowing for stakeholder choice. The CDE ecosystem's event-driven, fine-grained transactions anticipate improved productivity, as required by the generation of digital twins, smart buildings and smart cities, and to allow for wider connectivity with Artificial Intelligence (AI) services.

1.2 Purpose & aims

This paper integrates the purpose and aims of our research methodology as follows,

- Previous research (comparative, qualitative) summarised in Section 1.4
- Literature Review in Section 2 - framing the research question (RQ)
- Method in Section 3 describes the proposed implementation process
- Simulation of a proof-of-concept CDE ecosystem (heuristic, quantitative) in Section 4

1.2.1 Research Question

- How is it possible for multiple CDEs to communicate and share data with each other?

In response, we have focussed on a technical solution to improve interoperability which we believe will also improve relationships between enterprises, businesses and project teams.

1.3 Structure

Section 2 – Literature Review

The RQ is framed by a review of AECO sector standards and solutions which aim to improve interoperability at ecosystem level. We identify gaps in current research to support and reinforce further enquiry by the RQ.

Section 3 – Method

We describe the proposed scientific contribution and the method implemented.

Section 4 – Simulation

We describe the simulation approach, including: setting; testing; workflow.

Section 5 – Discusssion

This section assesses the goals of the simulation: connectivity; vendor-agnostic; event-driven; fine-grained. We also discuss the limitations of the research design and the simulation, and the challenges of scaling-up, security and adoption by the sector.

Section 6 - Conclusion

We report the advantages of a CDE ecosystem, describe future research, and summarise the paper.

1.4 Previous Research

UniSA STEM authors are involved in ongoing research and implementation to improve interoperability in the Process industries, initially demonstrated via a series of Pilot projects (Mayer *et al.*, 2013, Selway *et al.*, 2015, Selway *et al.*, 2017). In (Doe *et al.*, 2022) we compared interoperability across sectors, including AECO, Process industries, Geospatial, and Manufacturing and Engineering. For example, we noted that, in contrast to AECO and the Process industries, the Manufacturing & Electronics sectors' economic power and long-term working relationships have enabled them to drive the development of proprietary, closed software systems suitable for design, modelling, fabrication and collaboration throughout a product's lifecycle. Our recommendations to improve interoperability in the AECO sector included:

1. A hybrid Interoperability Ecosystem to implement object-based data exchanges and transactions.
2. Standardised IFC adaptors (event-driven, object-based) to exchange data either directly between systems or via a federated server.
3. Endorsement of a modular approach to the development of adaptor and API standards, as proposed in the *'Technical Roadmap'* (buildingSMART, 2020, pp. 17)

In (Doe *et al.*, 2024) we surveyed AECO sector industry partners' views and opinions to extend understanding of interoperability challenges and solutions for meeting these challenges. Research questions were framed by a literature review, survey, semi-structured interviews and focus group discussions. Qualitative input was appropriate when dealing with the complexity of issues encountered and helped formulate meaningful solutions. When prompted, industry partners GHD and DBM Vircon, identified a key interoperability problem,

> *…we have multiple CDEs on a project, and connecting information from one CDE to another is somewhat problematic. (Doe et al., 2024)*

Table 1: Comparison of ISBM and LBD ecosystem characteristics.

Ecosystem Characteristics	ISBM	LBD
Ecosystem structure: distributed.	●	●
Decision-making: centred on a project-based community.	●	
Decision-making: decentralised across sub-communities.		●
Links data: across AECO sector domains.	●	
Links data: across the 'semantic web'.		●
Standards: AECO sector global, industry agreed (ISO, bSI…).	●	
Standards: internet community by the World-Wide Consortium (W3C).		●

Further recommendations included:

4. Development of Germany's Publicly Available Specification (PAS), DIN SPEC 91391 series, 'Common Data Environments (CDE) for BIM projects'. We proposed that function sets and open data exchange between platforms of different vendors be promptly developed into a global ISO standard or incorporated into ISO 19650 series to provide clarity for the sector and for software vendors.

5. Implementation of a proof-of-concept, network-centric, ISBM, a novel middleware approach to improve interoperability in the AECO sector.

ISBM and Linked Building Data (LBD) were compared as candidate solutions for AECO sector ecosystem interoperability (Table 1). LBD is under review by bSI as a means to improve AECO sector interoperability, but no firm proposal for LBD technology exists yet (Pauwels, Zhang and Lee, 2017).

This paper describes implementation of recommendation 5 above, a proof-of-concept CDE ecosystem, a network-centric, middleware approach which aims to improve AECO sector interoperability.

2. LITERATURE REVIEW

We frame and revisit the RQ by reviewing standards and solutions relevant to achieving connectivity between distributed and fragmented systems and applications. Whilst nations, regions and industry all publish standards we focus on global standards published by bSI and the International Organization for Standardisation (ISO) which hierarchically influence the rest. As the proof-of-concept includes systems originally demonstrated in the Process industry we briefly review its standards which are published by Mimosa (MIMOSA, 2021). We also review cross-sector technical solutions developed to connect enterprises, business and projects.

2.1 Standards

2.1.1 Common Data Environment (CDE)

CDEs for the AECO sector were initially described by Publicly Available Specification (PAS) 1192 in the UK in 2011 which formalised a framework to achieve Building Information Modelling (BIM) Level 2. PAS 1192 was upgraded to global standard ISO 19650 series in 2018 ((ISO), 2018). The CDE specifies a Database Management System (DBMS) capability to manage an 'Information Container' with attributes and meta data, and a transmittal capability to issue update notices to team members and maintain an audit trail of information handling. The Information Container transfers information common to domains and is present in four states – WIP (Work-in-Progress), Shared, Published, Archive (Figure 1).

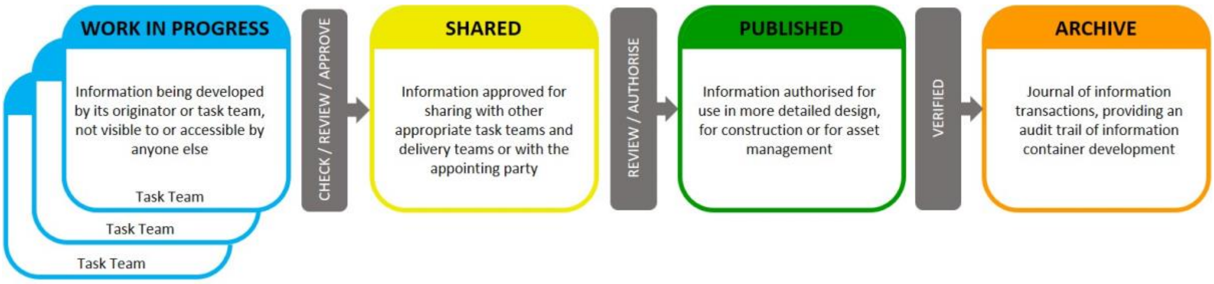

Figure 1: AS ISO 19650.1: 2019, 'Figure 10 – Common Data Environment (CDE) concept' (adapted by authors).

According to ISO 19650, the CDE should manage an information container's state, status and revision. Regarding the latter, (ZBIM) confirm that 'no revision numbering schema or standardised status codes' are defined by ISO 19650, thus delivery teams are advised to agree terminology and the best solution for the project. (Jaskula *et al.*, 2024) report that CDEs are now widely used by the sector and assert that, 'solutions to integrate data between multiple CDEs must be investigated'.

2.1.2 OpenCDE

The bSI's openCDE Technical Team are currently defining openCDE APIs which will streamline the process of exchanging files with a CDE. As noted by (buildingSMART, 2025c), Documents API is a member of the openCDE API family which enables the user to upload and download documents of all file types to and from the client and server, thus facilitating exchange of data between client (user) and server (CDE). Solutions which enable exchange of data between users and multiple CDEs are not yet available,

> *2.2.4. Automatic Syncing of Documents Between Two or More CDEs. This use case is not yet supported. It will be added in the future. (buildingSMART, 2025a)*

Germany's PAS DIN SPEC 91391 ((DIN), 2019) series released 2019 provides a comprehensive definition of expected open data exchanges between multiple CDEs which 'should be able to mutually exchange data without losses'. PAS DIN SPEC 91391 supports exchange of large binary files (e.g. CAD, BIM models) capable of managing information containers as the smallest unit of information, in accordance with CDE BIM Level 2,

> *They structure information in information containers as the smallest unit of information for data exchange. ((DIN), 2019)*

But PAS DIN SPEC 91391 does not yet support exchange of fine-grained data, in accordance with CDE BIM Level 3, which would be capable of referencing object properties and attributes as the smallest unit of information,

> *The logical implications of fine-grained cooperation according to BIM Level 3 have not yet been translated into viable concepts (e.g. ontological perspectives). ((DIN), 2019)*

2.1.3 Data Security

Across connected CDEs, PAS DIN SPEC 91391 (2019) defines security requirements as follows,

Intellectual property (IP):

1. Reliable authentication procedures for CDE users.
2. Suitable encryption methods for exchanges of data between users and CDEs.
3. Compliance with data security standards to reduce risks.
4. Reliablility checked by regular independent penetration tests.

CDE Dataset (project information, communication flows and content, logs, version and status information):

1. Mirroring of data centres to mitigate risk of loss.
2. Seamless monitoring of hardware.
3. Availability of standard measures and resources.
4. Temporally and geographically staggered replication plans.

Data security for OIIE ISBM is provided in several ways, as noted by (Kaur *et al.*, 2018),

1. The use of SSL (Secure Sockets Layer)/TLS (Transport Layer Security) for communication security.
2. Security tokens to manage the authorisation of systems to communicate across channels of the ISBM.
3. Role-based security for both people and systems in the OIIE.
4. Management provided by the OIIE Ecosystem Administrator.

An alternative means of ensuring privacy and security for IP and datasets may be offered by ISO/TC 307 'Blockchain and distributed ledger technologies' (ISO, 2025) which is currently developing a global standard. (Dounas, Hall and Kifokeris, 2025) propose implementation of Blockchain's public ledger of transactions to ensure privacy and security across the fragmented and decentralised European Union's (EU) Digital Building Logbook (LBD).

2.1.4 IFC5 & Partial Exchanges

IFC promises sector-wide interoperability by exchanging data via a common language. Developed since 1994 by bSI it is now defined by ISO 16739-1:2024 (buildingSMART, 2025b). Fifteen years ago, (Eastman *et al.*, 2010) refined discussion of IFCs intrinsic but limited interoperability capability with concepts defining BIM data for each domain or use case including, model view definitions (MVD), information delivery manuals (IDM), exchange models (EM) and exchange objects (EO). The forthcoming IFC5 will facilitate internet connectivity and allow users to exchange IFC data using JSON (JavaScript Object Notation), XML (eXtensible Markup Format), RDF (Resource Description Framework), and HDF5 (Hierarchical Data Format), independent of the current EXPRESS schema (Berlo *et al.*, 2021).

According to (Domer and Bernadello, 2023) IFC5 will be the preferred method of integrating data into applications and CDEs. IFC5 will eventually facilitate use cases like working with connected CDEs, and business concepts such as digital twins, smart cities, and smart buildings (Berlo *et al.*, 2021). IFC5 is influenced by the visual effects (VFX) sector, where World Pixar developed a standard called Universal Scene Descriptor (USD). IFC5 transactions may be made at the level of the smallest possible dataset called a 'component', a sub-part of a single object e.g. rebar or concrete could have component IDs and version IDs and be part of an entity ID called a column (buildingSMART, 2023). USD's influence extends to consideration of a distributed relationship between components and the applications which create them, a profound change from present relationships which embed applications into the way that objects are made,

> *Anyone can add a component and it doesn't need to come from the same author. (buildingSMART, 2023 (Greg Schleusner))*

IFC5 will allow transactions to be tracked and versioned with a high level of granularity where differences may be 'queried' by a 'diff' (difference) function and a 'version control system' (VCS), as demonstrated by (Postle, 2023) using BlenderBIM and GitHub. (Charest and Rogers, 2020) suggest that such transactional updates are best effected by APIs and shared messaging services,

> *APIs and Messaging system methods more easily support transactional updates to avoid constant bulk resynchronization and are likely better options in this scenario [dataset updates and replacement]. (Charest and Rogers, 2020)*

2.1.5 Open Industrial Interoperability Ecosystem (OIIE)

The OIIE specification is described in ISO 18101 (Automation & Integration). It makes best-practice use of other standards such as ISA-95 Part 6: Messaging Service Model (MSM) which describes a set of messaging services for information exchanges. OpenO&M ISBM (Open Operations & Maintenance, Information Service Bus Model) is an open standard that provides a vendor-neutral interface to the communication infrastructure of the OIIE™ Architecture. It can be used in any industry as it allows the transmission of any information model, including MIMOSA CCOM, ISO 15926, MESA B2MML and OAGIS, and its APIs provide a wide range of interfaces which serve as an integrated communication backbone for an ecosystem (MIMOSA, 2021).

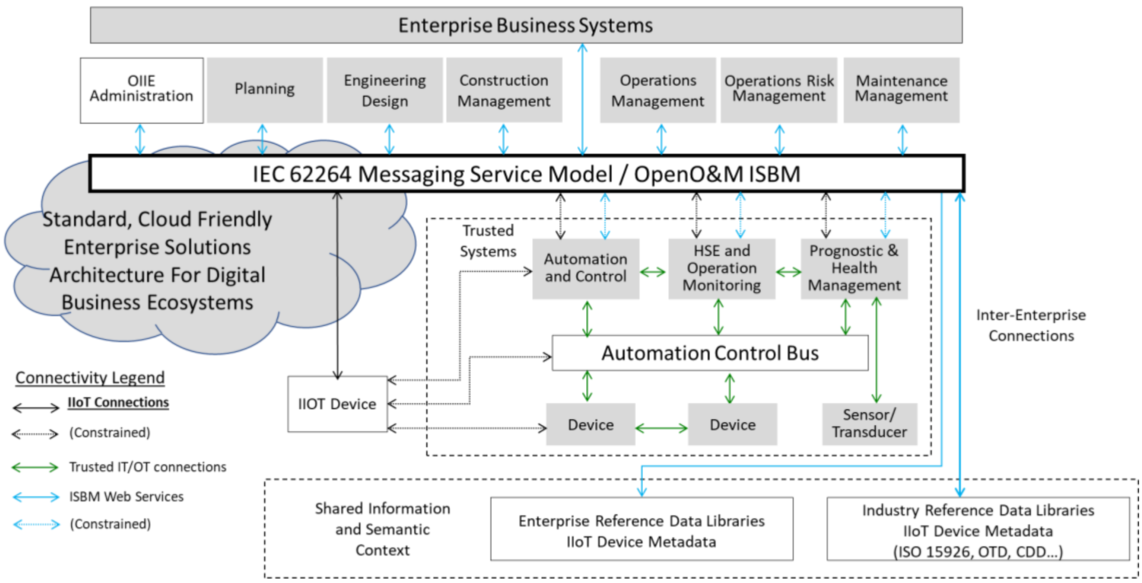

Figure 2: OIIE Architecture, from (Kaur et al., 2018).

Figure 2 illustrates the overall OIIE Architecture (Kaur *et al.*, 2018):

- Top level; different activities or systems involved in lifecycle operations - each a system, or SoS, that can communicate directly with one another, or through the ISBM.

- Middle level; the many connections that may occur between components, IIoT devices and the ecosystem as a whole through ISBM.
- Lower level; connections to industry-wide, and enterprise specific reference data libraries (RDL) which provide semantic connectivity across databases.

According to (Kaur *et al.*, 2018), the OIIE core set of features address all of the principles of Industry 4.0, as defined by (Hermann, Pentek and Otto, 2015):

- Interoperability: standards-based, one-to-many, many-to-many.
- Virtualisation: of a digital twin.
- Decentralisation: an SoS - a federated, distributed system.
- Real-time capability: for data analysis and automated decision-making.
- Service orientation: ISBM provides intra- and inter-enterprise data exchanges through web services.
- Modularity: a federated, distributed system which aims for plug-and-play capability
- Security: as described in Section 2.1.3.

2.2 Solutions

In this section we review enterprise and business level information exchange systems which aim to leverage a fully digital ecosystem. (Gallaher *et al.*, 2004) confirmed that poor business processes contributed most to AECO sector interoperability failures. (Grilo and Jardim-Goncalves, 2010) asserted that, 'evolution towards a highly distributed and fully digital ecosystem of players (i.e., architects, structural engineers, and HVAC engineers) is technically feasible'.

2.2.1 Enterprise Service Bus (ESB) & Message Service Bus (MSB)

An Enterprise Service Bus (ESB) is implemented in software that operates between business applications, and enables communication among them (Chappell, 2004),

> *An ESB is a standards-based integration platform that combines messaging, web services, data transformation, and intelligent routing to reliably connect and coordinate the interaction of significant numbers of diverse applications across extended enterprises with transactional integrity. (Chappell, 2004)*

Internet communication no longer restricts the term 'enterprise' to a corporate entity. (Flurry, 2018) defines an ESB as an architectural pattern called Service-Oriented Architecture (SOA) supplying loosely coupled connectivity between service requesters and service providers in service-oriented solutions and allowing 'discrete separation of temporal, technological, and organisational processes'.

A Message Service Bus (MSB) augments an ESB by providing decentralised, message distribution and a publish/subscribe interaction pattern: its message exchange is decoupled and anonymous, 'supporting a many-to-many style of communication' (Jacobsen, 2018). As noted in Section 2.1.5, the OIIE architecture provides the ISBM standard, a version of MSB, which defines a method for implementing SOA principles via the Simple Object Access Protocol (SOAP) API which '...provides a standard interface or an abstract layer to any ESB' (MIMOSA, 2025).

2.2.2 Linked Building Data (LBD)

The 'Semantic Web' is a term coined by Tim Berners-Lee which defines a holistic approach based on computers' ability to find and link semantic meaning in data, and to use rules to infer reason through logic (Berners-Lee, Hendler and Lassila, 2001). In 2007 a pioneering LBD approach was implemented at the Sydney Opera House, NSW, Australia which combined IFC and the semantic web using Resource Description Framework (RDF) and Web Ontology Language(OWL) to enable '…loosely coupled software applications to collaborate as if they were one application' (Schevers *et al.*, 2007). bSI's Linked Data Working Group (LDWG) launched 2015, developed ifcOWL which allowed '…building data to be easily linked to material data, geographic information system (GIS) data, product manufacturer data, sensor data, classification schemas, social data, and so forth'. Ultimately, due to the language differences between EXPRESS and OWL, the conclusion was reached that ifcOWL 'was an academic exercise' and 'is full of exceptions and particularities that make it hard to use in practice' (buildingSMART, 2025d).

With LBD, instead of files, information is represented in 'structured graphs' which can be integrated with different sources of data and repositories (Pauwels, Zhang and Lee, 2017). (Werbrouck *et al.*, 2019) propose a decentralised CDE: they compare *Social Linked Data,* an LBD ecosystem, with the AECO sector's ecosystem noting similarities which include , '…standardised data representations, role- or actor-based authorisation and authentication and the need for modular and extensible applications dedicated to a specific use case'. The paper presents a proof-of-concept which envisages a decentralised CDE bridging '…multiple data stores of different project stakeholders and the end user'. In previous work we compared the characteristics of LBD and ISBM, as noted in Section 1.4.

2.2.3 Application Programming Interface (API)

APIs focus on vertical integration between an enterprise's back-end systems and third parties (systems of engagement), whilst ESBs provide horizontal integration between an enterprise's back-end systems (systems of record) (Widjaja, 2025). An API is a standard way of connecting applications by getting data from someone else's server or service into your application via a set of constraints and protocols (Goodwin, 2025). An API comprises a variety of functions or procedures that an application program can access, 'as well as data structures, constants, and various definitions needed to describe system resources'. (Harry, 2009). Protocols commonly used include Representational State Transfer (REST), a set of API constraints which uses Hypertext Transfer Protocols (HTTP) with the information requests POST, GET, PUT and DELETE.

By decoupling client and server applications APIs allow technology changes to systems to occur independently (IBM, 2025). For example, bSI's openCDE Documents API will enable an Oracle Aconex CDE (project management, information workflow) to connect with a Solibri CDE (quality assurance checking - geometry analysis, clash detection, code compliance) so that the same information is available in both CDE applications, thus saving time, increasing the quality of information exchanges, and removing silos which exist between domains (buildingSMART, 2025c). Documents API and BCF API can be used together by a client application that has implemented both APIs. For example, a user working with a client application that implements both APIs can download models directly from the CDE using the Documents API and manage issues relating to those models using the BCF API. A BCF file is commonly exchanged via API web services where an 'issue' is assigned and tracked through statuses 'Open', 'In Progress', and 'Closed' (buildingSMART, 2021). In this scenario communication of data and issues occurs asynchronously between users, and the process is decoupled from the applications and services which will eventually resolve the issue. (Kandler, Heiß and Rüppel, 2025) note that information recorded by BCF is unstructured thus limiting the ability to analyse or reuse it; the authors develop a strategy for making BCF data structured using natural language processing (NLP). In a systematic literature review (Borkowski, 2025) report on BCF in the context of BIM representation, data exchange and decision support as follows:

- Benefits: a lightweight, tool-agnostic issue container, linking discussion, viewpoints, and element identifiers to specific model contexts.
- Improvements: specification of stable identifiers and versioning across CDEs, extended issue schemas with requirement/risk links, and design governance patterns that prevent drift between issues and CDE approved model states.

2.3 Gaps

The findings of the review support and reinforce further enquiry by the RQ.

- No global or industry standard is available yet which defines the requirements for achieving connectivity between multiple CDEs, though this is anticipated by IFC5, ISO 19650 and DIN SPEC 91391.
- No solution has been identified to provide connectivity between multiple CDEs, though two candidate solutions exist, LBD and MSB (ISBM).
- No standard or solution exists for fine-grained resolution of data exchanges between multiple CDEs, though this is anticipated by IFC5, ISO 19650 and DIN SPEC 91391.

As noted earlier in Section 1.4, LBD has been widely reported by others as a candidate solution and we have compared LBD with ISBM in our paper (Doe *et al.*, 2024). The method and simulation sections describe a novel approach for the AECO sector - a proof-of-concept MSB (ISBM) solution.

3. METHOD

The implementation followed the method illustrated in (Figure 3), also described in more detail in Appendix A. The following systems and applications were integrated:

- ISBM, open standards-based.
- Adaptors, custom designed by the authors, open standards-based.
- Authoring tools and CDE services, developed by others, open-source, free.
- IFC models and data, open-source, free.

3.1.1 ISBM

ISBM was hard-coded into BlenderBIM and BIMserver with 'session-ids'. These identifiers provide a lightweight mechanism for user authentication and enable traceability of user actions across the workflow. As noted above in Section 2.1.5, ISBM is an open standard that provides a vendor-neutral interface to the communication infrastructure of the OIIE™ Architecture. As noted in Section 2.1.5 ISBM is a version of an MSB and can be used in any industry as it allows the transmission of any information model, thus serving as an integrated communication backbone for an ecosystem.

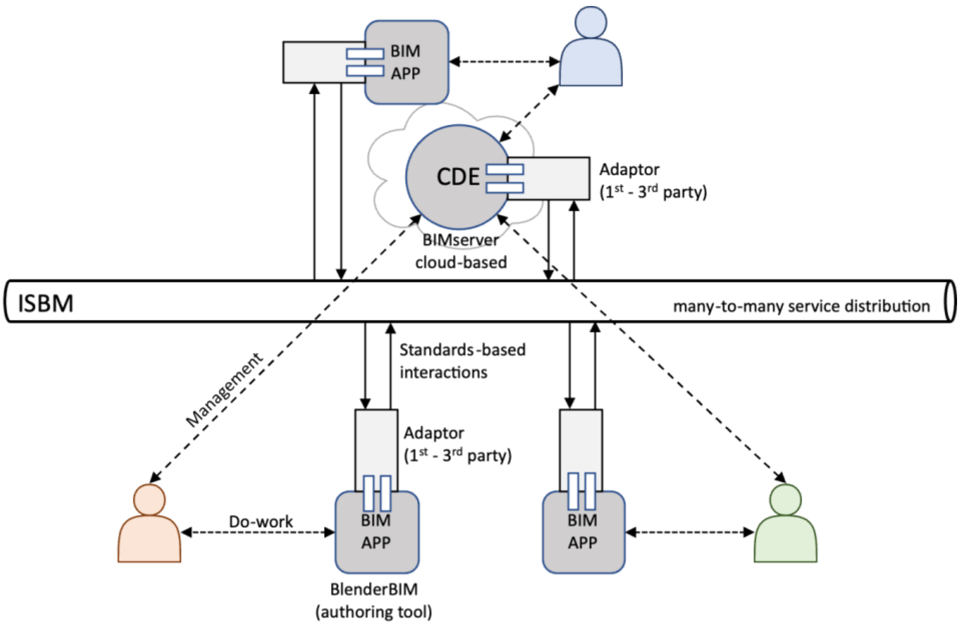

Figure 3: Implementation method.

3.1.2 Adaptors

Custom adaptors were developed to connect systems and applications to their respective APIs. Adaptors allow for decoupling of connections giving greater flexibility and scalability if modular updates, changes or deletions are required. They also enable fine-grained, event-driven messaging and data exchange - see Section 4.2.2. The long-term intention is that first or third party vendors will produce standards-compliant, commercial off-the shelf (COTS) adaptors, enabling 'plug-and-play' interoperability between heterogenous systems.

3.1.3 Authoring tools and server (CDE)

BlenderBIM served as the participants' authoring environment. BlenderBIM (now Bonsai) is an open-source, native IFC authoring tool originally developed by Dion Moult and available as an add-on from IfcOpenShell

(2025). BIMserver acted as the CDE, providing IFC based storage and information management. Its services include model-checking, versioning, project structuring, clash detection and merging. Of significance is its ability to query, merge and filter a BIM model and to quickly generate IFC files. BIMserver was developed by Jacob Beetz, Léon van Berlo and others (Beetz *et al.*, 2010).

3.1.4 IFC models and data

BlenderBIM's functionality is built on IfcOpenShell, an open-source software library for developers working with IFC. The IFC format provided by IfcOpenShell allowed for read, write, and modify capabilities. We sourced IFC data created by the US Army Corps & Engineers (USACE, 2011) for a 2-storey, duplex apartment building separated into architectural, mechanical, electrical and plumbing (MEP) models.

4. SIMULATION

In this section, we describe the use case setting, and testing of the CDE ecosystem simulation, and provide a video hyperlink to a 4-minute workflow sequence. The goals of the proof-of-concept CDE ecosystem were:

1. Connectivity, across distributed CDEs and applications, supporting legacy and existing systems.
2. Vendor-agnostic, based on industry and global standards.
3. Event-driven, timely information sharing via messaging services.
4. Fine-grained, object-based, partial exchanges and synchronisation of data.

Aims 1 and 3 were tested heuristically and by observation. Aim 2 is discussed in Section 5. Aim 4 was tested experimentally and by observation.

4.1 Setting

The simulated setting involved users, systems, and services undergoing a sequence of actions during an ISO 19650 Work-in-Progress (WIP) state, as follows:

1. MEP 1 engineer loads a model of a project from the CDE into an authoring tool - the model is also shared between an Architect and MEP 2 engineer.
2. MEP 1 engineer updates the model and publishes changes to the CDE - a 'minimal changeset' only.
3. The CDE receives the changes, reconciles them and their metadata, and incorporates the changes into the project revision.
4. Assuming that the revision meets quality, workflow, coordination and other necessary criteria, the CDE then publishes the revised information and changes to all interested parties.
5. All parties review the published updates and take action if required:
    a. MEP 2 engineer checks for changes and receives notification through their authoring tool that the project has been revised:
        i. Reviews the changes, accepts and merges them into their model.
        ii. May make other changes and publish their updated model to the CDE.
    b. The Architect checks for clashes and receives notification through their authoring tool that a clash has occurred:
        i. Reviews the changes and the clash between elements.
        ii. May ask MEP 1 engineer to move their element to avoid the clash, or decides on alternative action.

Event-driven messaging allowed for a dynamic workflow, thereby leveraging 'best-of-breed' solutions for performing tasks.

4.2 Testing

4.2.1 Connectivity

The BlenderBIM source repository was cloned from GitHub and its functionality extended by an 'add-on' panel with customised buttons (Figure 4).

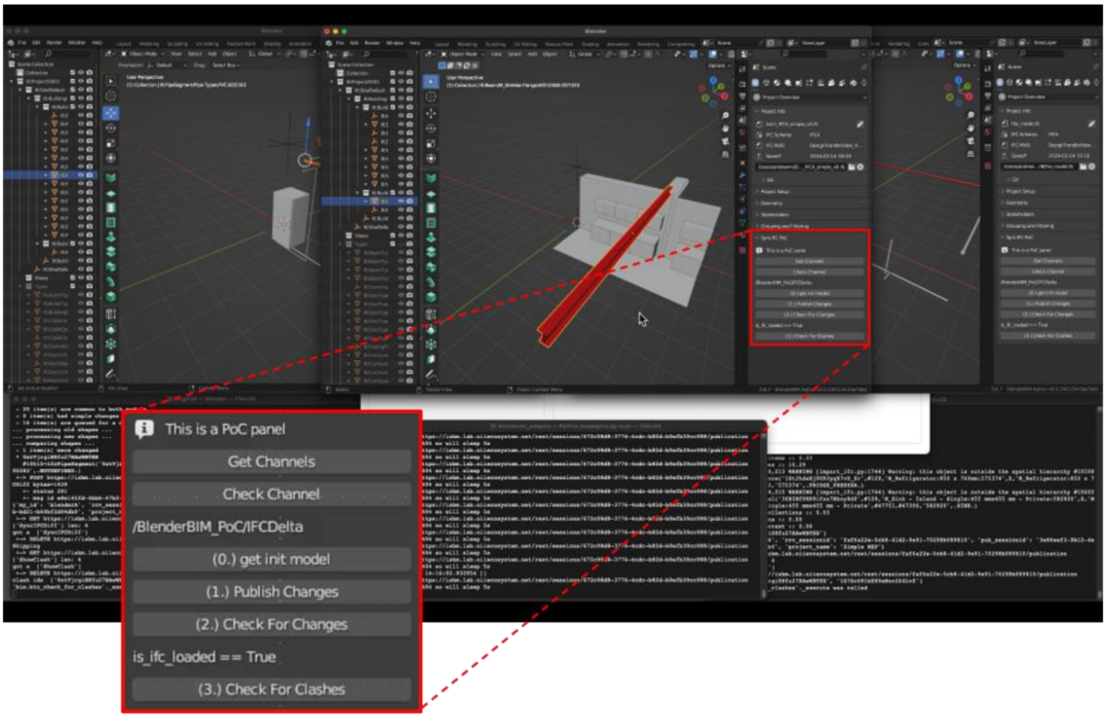

Figure 4: Proof-of-concept panel, custom buttons (screen shot: 3 x BlenderBIM, BIMServer, ISBM running concurrently).

Three instances of BlenderBIM were assigned to one architect and two MEP engineers. Correspondingly, an IFC file named 'Simple Arch' and two named 'Simple MEP' were created. These sub-models were converted to Git repositories by the '(0.) get init model' custom button. Git repositories facilitated comparison between model versions, assisted by a SyncIFCDiff function.

The add-on panel's custom buttons were ordered top to bottom relative to the demonstration sequence. In the following list, 'channels' refers to pathways for the flow of interrelated messages:

Get channels – Checks communication with the ISBM instance.

Check channel – Checks that the channel is present in the demonstration.

Buttons (0) to (3) are used after opening an IFC project file in BlenderBIM:

(0.) get init model - Used by BlenderBIM A and B to download the same version of their model initially and convert the IFC folder to a Git repository. The BIMserver adaptor should be running.

(1.) Publish Changes. Used only by BlenderBIM A. It publishes the SyncIFCDiff message.

(2.) Check for Changes. Used only by BlenderBIM B. It consumes the SyncIFCDiff message.

(3.) Check for Clashes. Used on BlenderBIMs A,B,C to get the last clash status.

BIMserver was cloned from GitHub so that changes could be independently made. Sub-projects named 'Simple Arch' and 'Simple MEP' were set-up within BIMserver and linked to BlenderBIM. The authors developed a BIMserver adaptor function as a basic server process to facilitate initial model checkout from BIMserver, and to centrally process the handling of diff messages which identified changes between versions of models.

4.2.2 Event-driven

Event-driven messaging was tested during WIP workflow between the three users. Messaging services published and consumed via ISBM open channels were triggered by the users' BlenderBIM add-on panel:

- MEP1 triggered the initial event-driven exchange by clicking on the add-on panel's 'Publish Changes' button (Figure 4).
- MEP 2 engineer then clicked the 'Check for Changes' button and consumed changes made in the sub-model, assisted by an IfcDiffSync function developed by the authors,.
- Lastly, the architect clicked the 'Check for Clashes' button to trigger BIMserver's clash detection services and consume the changes into their sub-model.

Through observation of these event-driven exchanges we were able to refine their outcomes so that they were meaningful for the users involved. The workflow is explained and depicted more fully in Section 4.3.

4.2.3 Fine-grained

A simple IFC model with slab elements was created in BlenderBIM (Figure 5). We used this model to test and confirm partial exchange capability between ISBM, adaptors, and applications by querying objects at fine-grained level. The 'object' was the coordinate system related to the slab element. The BlenderBIM add-on panel's '(0.) get init model' button (Figure 4) converted IFC files to Git repositories using a SyncIFCDiff function developed by the authors. This confirmed fine-grained differences between versions of the IFC model.

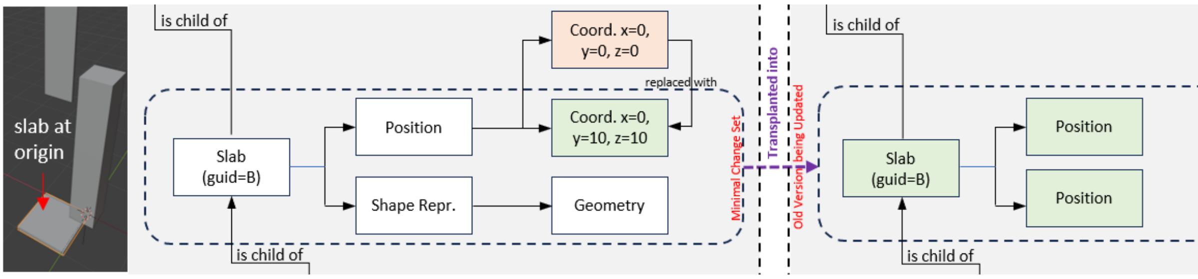

Figure 5: Fine-grained change set – slab object moved (0,0,0 to 10,10,10).

4.3 Workflow

4.3.1 Sequence

Sequential images from the video capture moments in the CDE ecosystem simulated setting where alterations are made and checked 'on the fly' (Figure 7). Resulting changes are reviewed asynchronously by team members.

A 4-minute video recording (Figure 6) was presented to industry partners and others. It depicts the simulated setting of the WIP stage of ISO 19650, as described in Section 4.1. The video illustrates the potential of a CDE ecosystem supported by reliable communication via event-driven data exchanges between each team member, applications and services.

CTRL | click

Figure 6: Proof-of-concept video (the green button is hyperlinked to a restricted YouTube channel).

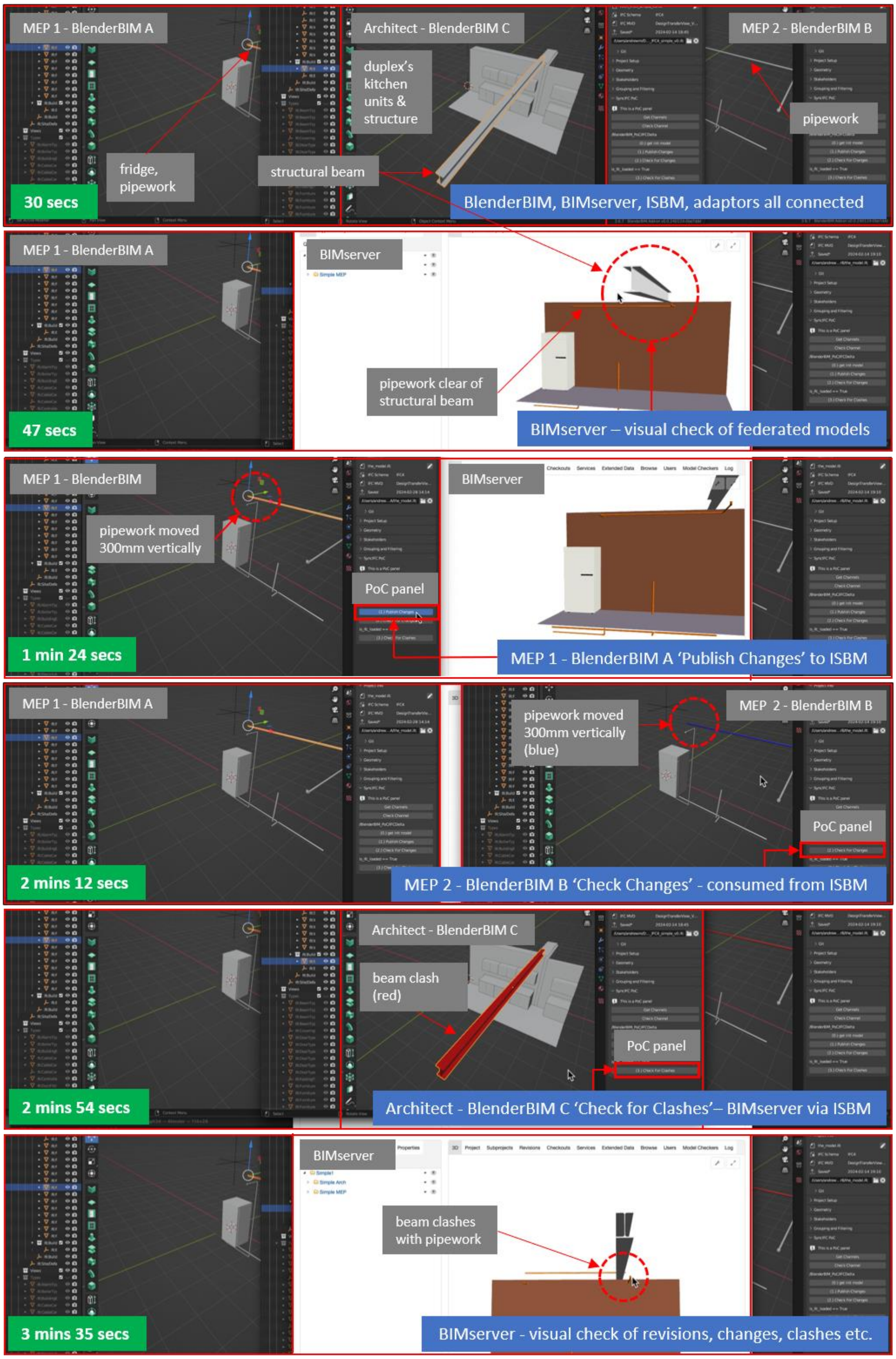

Figure 7: Proof-of-concept typical ISO 19650 WIP workflow sequences – annotated screenshots.

5. DISCUSSION

We assess the goals of the simulated proof-of-concept CDE ecosystem,

1. Connectivity, across distributed CDEs and applications.
2. Vendor-agnosticism, based on open, global standards.
3. Event-driven, timely, messaging services.
4. Fine-grained, object-based, partial exchanges.

We also discuss the limitations of the research design simulation methodology, and the challenges of scaling-up the proof-of-concept.

5.1 Connectivity

Connectivity between users and systems is vital because a CDE is a decentralised source of partial truth distributed across fragmented users and systems, rather than a centralised single source of truth. This was illustrated in Section 2.1.5 and Figure 2 by the multiplicity of standards-based connections depicted for the OIIE Architecture. Though our short-term goal is to develop the proof-of-concept towards a scaled-up version of a CDE ecosystem our longer-term goal is to demonstrate similar levels of standards-based connectivity essential for an AECO ecosystem. Hence, we illustrate a version of the OIIE Architecture adapted for the AECO sector (Figure 8) which similarly predicts the multitude of connections between systems, users and IIoT devices required for the operation and maintenance of digital twins, smart buildings, and smart cities.

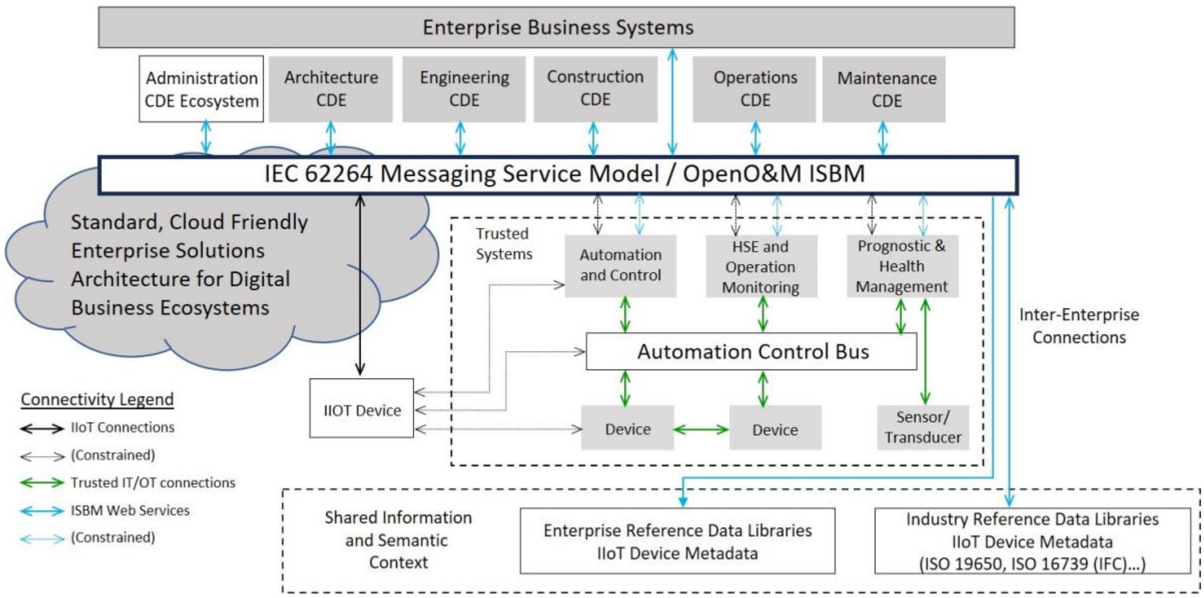

Figure 8: CDE ecosystem for the AECO sector.

The top level defines different activities or systems involved in AECO sector lifecycle operations, the middle level shows the many connections that may occur via ISBM between components, IIoT devices and the ecosystem as a whole, whilst the lower level shows the many industry-wide connections, and enterprise specific RDLs which provide shared information and semantic connectivity across databases.

Connectivity which links enterprises, businesses and project processes should reduce the costs of poor interoperability, as noted by (Chapman, 2005) - see Section 1. We note here that future research should examine this issue more closely.

5.2 Vendor-agnostic

Open standards-based ISBM and COTS plug-and-play adaptors allow vendor-agnostic choices by AECO organisations because the CDE ecosystem understands and exchanges any vendor's data models and document formats. Nevertheless, the goal for bSI and the AECO sector, as noted in Section 2.1.4, is that IFC5 becomes the standard or common language for error-free, reliable exchange of data at object or component level. Though bSI and the IFC standard foster this approach vendors often decide the compliance level of their products' IFC export and import capabilities. Consequently, the CDE ecosystem allows for transition from the status quo where any model format may be exchanged with limited IFC compliance, to one where model formats adhering to open standard IFC5 will ensure that only error-free, reliable data is shared. As previously noted in Section 2.1.5, ISBM has been implemented in the Oil & Gas sector through the OIIE which provides a standard interface where a federation of systems is implemented from multiple software vendors working together to support business processes. ISBM offers interoperability via application-to-application communications for every version and format of message exchange system. Similarly, we believe that AECO sector standards-based interoperability will leverage the impetus needed to support vendor-agnostic choices by organisations. Certainly, the growing frequency of jurisdiction's (national, regional, local) mandates requiring contractors to implement open, standards-based IFC imports and exports will foster such change.

5.3 Event-driven

The simulation demonstrated that event-driven changes to a model may effectively be published and consumed by others - see Section 4.2.2. Checking and updating was resolved by manually triggered event-driven messaging using a customised panel in BlenderBIM. One of the goals with a scaled-up demonstration would be to include implementation of automatically triggered event-driven messaging. In such a case, an ‘event-listener’ would detect change and notify interested parties. Event-driven messaging facilitates control of published and subscribed data flows, enabling time for considered action and better decision-making. As also noted in Section 3.1.2 we envisage that standards-based, event-driven COTS adaptors would be developed by first and third parties to connect ISBM, or similar messaging services, to CDEs and applications. A natural extension to the CDE ecosystem would include AI-powered messaging services to leverage the value of data stored by CDEs, thus adding automation, personalisation, and intelligence to shared messaging workflows.

5.4 Fine-grained

The simulation also demonstrated fine-grained exchange of data using diff functions and Git repositories which facilitate version control, as noted in Section 2.1.4. An added benefit of diffs is that they foster collaboration by allowing team members to understand each other’s contributions, and by enabling errors to be more easily tracked and fixed thus providing a complete record of the project’s evolution over time. Version control, may be contrasted with revision control in the AECO sector which developed to track changes to large binary files (e.g. CAD, BIM models) related to workflow stage approvals (e.g. Planning, Design Development, Tender) and contractual milestones (e.g. 50%, 70% complete etc.). Revision control is recorded on the drawing sheet itself,

> *An accurate record of all changes made to released drawings is tracked via a revision block. This is important so that the sources of all changes may be understood, verified, and approved. (Giesecke et al., 2023)*

Currently, CDE BIM Level 2 capability aligns with revision control where the information container represents the smallest unit of information i.e. the output of a domain (e.g. architect or MEP engineer). The simulation demonstrated CDE BIM Level 3 capability which refers to the ‘object’ as the smallest unit of information. ISO 19650’s aim for ‘exchanging, recording, versioning and organising for all actors’ ((ISO), 2018) points to its goal for a future which includes such fine-grained exchanges. Similarly, bSI and IFC5 aim to align with this approach where the level of the smallest unit of information is a ‘component’ i.e. a sub-part of a single object. Hence, the transition from file to object exchange, and revision to version control is gaining pace. Finally, we note that these represent significant technical and cultural changes for the AECO sector needing careful management of project teams’ awareness and understanding.

5.5 Limitations & Challenges

5.5.1 Research design

We acknowledge that adversarial contracts also contribute towards poor interoperability in the AECO sector, a factor which restricts the sharing of information. Correspondingly, we support open, transparent and risk sharing contracts as fostered by Integrated Project Delivery (IPD), Partnering (NEC etc.), and ConsensusDocs (USA). However, this subject area was outside of the scope of our research.

5.5.2 Simulation

Simulation revealed heuristic insights rather than scientific outcomes. We simulated an ISO 19650 setting focussed on a specific use case in which users change design parameters and check outcomes on the fly. Hence, we arrived at solutions by trying different actions and checking outcomes. We have provided details of the setting, simulation and tools to allow for replication by others. Testing and validation of an experimental scaled-up version will improve and aim to prove the accuracy and authenticity of results.

5.5.3 Scalability & Adoption

The scalability and adoption of a CDE ecosystem would be a stepwise process. For example, our next step is to aim to secure research funding for testing and validating a simulated CDE ecosystem, and to identify COTS opportunities for adaptors and other services, in collaboration with industry partners. Steps towards AECO sector-wide scalability are dependent on development of mulitiple CDE interoperability in alignment with the ISO 19650 standard (Section 2.1.1), and development of IFC5 (Section 2.1.4). Further steps towards scalability include our recommendation that an AECO sector-wide implementation should make reference to the OIIE Interoperability Pilot, 'a public interoperability test-bed jointly run by MIMOSA in cooperation with multiple other industry associations' (MIMOSA, 2020). Similarly, bSI and AECO industry associations could jointly run an AECO Interoperability Pilot which would feed back lessons learned from the implementation of a variety of use cases.

Adoption of a CDE ecosystem by the AECO sector should consider recently introduced global standards and services. In 2018 the ISO 19650 series was published then, between 2020-2025, bSI's Technical Roadmap (buildingSMART, 2020) published the openBIM workflow standards about: requirements (Information Delivery Specification (IDS)); production (IFC4.3); checking (BIM Collaboration Format (BCF)); and, delivery (openCDE APIs). Services supporting these standards include: buildingSMART Data Dictionary (bSDD), Use Case Management (UCM), Validation (IFC compliance), and a Professional Certification course. An AECO Interoperability Pilot, which incorporates the CDE ecosystem and these foundational standards and services, could support wider adoption by facilitating collaboration between participants and demonstrating the capability of systems and software products to support use cases in settings such as ISO 19650 CDE workflow states.

5.5.4 Security

A distributed CDE ecosystem increases data security issues, as noted in Section 2.1.3. PAS DIN SPEC 91391 identifies measures to protect IP and the CDE Dataset, whilst OIIE ISBM defines security processes for communication and authorisation between people and systems, overseeen by an OIIE Ecosystem Administrator. Correspondingly we have identified the role of a CDE Ecosystem Adminstrator (Figure 10), for the AECO ecosystem. Broadly, we recommend that more research is required to identify data security measures appropriate for a distributed CDE ecosystem. As references, we point to ISO 27001:22 for establishing an Information Security Management System (ISMS) and the EU's General Data Protection Regulation (GDPR).

6. CONCLUSIONS

The proof-of-concept CDE ecosystem demonstrates an interoperable SoS solution for managing multiple CDEs. It is a method appropriate to the AECO sector's fragmented and distributed nature, capable of improving collaboration and therefore relationships between team members. A novel feature is an MSB (ISBM) which supports shared messaging interactions between CDEs, authoring tools, other applications and services across a single interface for communication. The content and meaning of these communications would be understood by all as they are based on accepted global standards including ISO 19650 (BIM) series and ISO 16739-1:2018 (IFC). Key benefits of the CDE ecosystem include:

1: Vendor Neutrality, through promotion of choice for organisations which may continue to use their preferred and/or specialised tools in projects connected via standards based ISBM and COTS plug-and-play adaptors.

2: Supports IFC5, inter-sector (buildings and infrastructure) interoperability through transactions at object or component level, facilitating connected CDEs, digital twins, smart cities, and smart buildings.

3: Many-to-Many Connectivity, which is configurable and capable of supporting the sector's complex interactions and organisational workflows which change according to the state and stage of the project.

4: Supports Industry 4.0, through a distributed CDE ecosystem, integrating with business concepts such as digital twins, smart cities, and smart buildings, and fostering interaction with AI.

5: Coordinated or Mediated, supporting a variety of workflows either through a primary CDE at specific ISO 19650 workflow states (Shared, Published, Archived), by another CDE, or anything in between.

6: Distributed Communication, between all cloud-based and dispersed design, construction, operations and facilities management team members connecting any system or application's distributed sources of truth.

7: Supports Legacy Capabilities, through incremental integration with existing capabilities (i.e. specifications, standards, and technologies) to realise improved outcomes, and support a desired future state.

6.1 Future Research

In future research, we aim to scale-up the CDE ecosystem through continued engagement with industry partners, as noted in Section 5.5.3. Our intention is to collaborate with a regional jurisdiction which, because of its procurement and development role, is capable of leveraging change by mandating contractors' and suppliers' compliance with openBIM and ISO 19650 standards. The outcome of further testing and validation by industry and academic partners would determine the CDE ecosystem's functionality and contribution towards achieving better collaboration and interoperability. Hence, development of the proof-of-concept towards an experimental research methodology would identify the causal effects of key variables on outcome measurements including: performance, interoperability costs, user acceptance, and error rates.

6.2 Summary

The proof-of-concept CDE ecosystem contributes a novel solution for achieving better interoperability for the AECO sector. It should be of interest to all stakeholders aiming to ensure interoperability across building lifecycles and wherever accountability for the reliability of data and information flows is required.

ACKNOWLEDGEMENTS

Thanks to industry partners: GHD (QLD), and DBM Vircon for their feedback and insights: Department of Infrastructure and Transport (DIT), South Australia and members of the DCCIF BIM Working Group.

Author contributions: R.D., M.Se., conceptualisation; R.D., M.Se., methodology; R.D., writing and original draft preparation; R.D., M.Se., K.K., M.St., review and editing; R.D., M.Se., visualisation; M.St., overview. Andrew McRae, software development, video production. All authors have read and agreed to the published version of the manuscript.

Funding: This proof-of-concept research received no external funding. It was supported by internal seed funding from the UniSA Industrial AI Reseach Centre.

Data Availability Statement: IFC data is from the Duplex Apartment project first published in Germany and provided to the US Construction Engineering Research Laboratory. These files were hosted by the US National Institute of Building Sciences from 2009 – 2020 and were originally provided and are used here under a Creative Commons Licence (CC BY 4.0).

Declaration of generative AI and AI-assisted technologies in the writing process: No AI technologies were used.

Other data presented in this study is available on request.

Conflicts of Interest: The authors declare no conflict of interest.

REFERENCES

(DIN), D.I.f.N. (2019) 'DIN SPEC 91391-1:2019-04'. Common Data Environments (CDE) for BIM projects Germany.

(ISO), I.O.f.S. (2018) 'ISO 19650-1:2018'. Organization and digitization of information about buildings and civil engineering works, including BIM

Afsari, K., Eastman, C. and Shelden, D.R. (2017) 'Building Information Modeling data interoperability for Cloud-based collaboration: Limitations and opportunities'. International Journal of Architectural Computing, 15 (3), pp. 187-202.

Beetz, J. et al. (2010) Published. 'BIMserver.org - an open source IFC model server'. CIB W78 27th International Conference on Applications of IT in the AEC Industry (CIB-W78), 2010 Cairo, Egypt. International Council for Research and Innovation in Building and Construction (CIB), pp.1-8.

Berlo, L.v. et al. (2021) 'Future of the Industry Foundation Classes: towards IFC 5 (preprrint)'. CIB W78 2021

Berners-Lee, T., Hendler, J. and Lassila, O. (2001) 'The Semantic Web'. Scientific American, 284 (5), pp. 34-43.

Borkowski, A.S. (2025) 'Theoretical mechanisms of Building Information Modelling (BIM): information representation, data exchange, and decision support'. Journal of Civil and Hydraulic Engineering, 3 (3), pp. 159–167.

buildingSMART (2020) 'Technical Roadmap buildingSMART - Getting ready for the future'. Available at: https://www.buildingsmart.org/wp-content/uploads/2020/09/20200430_buildingSMART_Technical_Roadmap.pdf.

buildingSMART (2021) BIM Collaboration Format. Available at: https://www.buildingsmart.org/standards/bsi-standards/bim-collaboration-format-bcf/ (Accessed: 04/04).

buildingSMART (2023) TR3 - Report on IFC 5 session from last General Assembly. Available at: https://youtu.be/VilimUSSuOQ?si=muwhnK5TQJGtXIPU (Accessed: 17/10).

buildingSMART (2025a) Documents API. Available at: https://tinyurl.com/m34ceajt.

buildingSMART (2025b) Industry Foundation Classes. Available at: https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/.

buildingSMART (2025c) openCDE API. Available at: https://tinyurl.com/5y22pmy7 (Accessed: 1/7).

buildingSMART (2025d) What is ifcOWL? Available at: https://github.com/buildingSMART/technical.buildingsmart.org/blob/main/ifcOWL.md.

Chapman, R.E. (2005) 'Inadequate Interoperability: A Closer Look at the Costs'. International Symposium on Automation and Robotics in Construction (ISAARC). Ferrara, Italy, International Association for Automation and Robotics in Construction (IAARC).

Chappell, D.A. (2004) Enterprise Service Bus. Sebastopol, CA: O'Reilly.

Charest, G. and Rogers, M. (2020) Data Exchange Mechanisms and Considerations. Available at: https://enterprisearchitecture.harvard.edu/data-exchange-mechanisms (Accessed: 11/06).

Doe, R. et al. (2022) 'Interoperability in AECO Sector and the Oil & Gas Sectors: Object-based Standards and Systems.'. The Journal of Information Technology in Construction (ITcon), (Special Issue from The Eastman Symposium), pp.

Doe, R. et al. (2024) 'Ecosystem interoperability for the AECO sector'. The Journal of Information Technology in Construction (ITcon), 29 347-376.

Domer, B. and Bernadello, R. (2023) Interoperability. Lausanne, Switzerland: EPFL Press.

Dounas, T., Hall, D. and Kifokeris, D. (2025) 'Digital building logbooks on the blockchain: first conceptualisation and future research directions'. Blockchain, Smart Contracts and Distributed Ledger Technologies in the

Built Environment: Key concepts, technologies, and applications. The Institute of Engineering and Technology, pp. 147-175.

Eastman, C. et al. (2010) 'Exchange Model and Exchange Object Concepts for Implementation of National BIM Standards'. Journal of Computing in Civil Engineering, 24 (1), pp. 25-34.

Flurry, G. (2018) 'Enterprise Service Bus'. In: Liu, L. and Özsu, M.T. (eds.) Encyclopedia of Database Systems. New York, NY: Springer New York, pp. 1308-1311.

Gallaher, M.P. et al. (2004) 'Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry'.

Giesecke, F. et al. (2023) Technical Drawing with Engineering Graphics. 16th edn. Australia: Peachpit Press.

Goodwin, M. (2025) What is an API? . Available at: https://www.ibm.com/think/topics/api.

Grilo, A. and Jardim-Goncalves, R. (2010) 'Value Proposition on Interoperability of BIM and Collaborative Working Environments'. Automation in Construction, 19 522-530.

Harry, H. (2009) Encyclopedia of Computer Science and Technology, Revised Edition. New York: Facts on File, Inc.

Hermann, M., Pentek, T. and Otto, B. (2015) Design Principles for Industrie 4.0 Scenarios: A Literature Review.

IBM (2025) What is a REST API? Available at: https://www.ibm.com/think/topics/rest-apis (Accessed: 2/7).

IfcOpenShell (2025) IfcOpenShell - The open source IFC toolkit and geometry engine Available at: https://ifcopenshell.org/ (Accessed: 2025).

ISO (2025) 'Blockchain and distributed ledger technologies'. ISO/TC 307. ISO.

Jacobsen, H.-A. (2018) 'Publish/Subscribe'. In: Liu, L. and Özsu, M.T. (eds.) Encyclopedia of Database Systems. New York, NY: Springer New York, pp. 2933-2937.

Jaskula, K. et al. (2024) 'Common data environments in construction: state-of-the-art and challenges for practical implementation'. Construction Innovation.

Kandler, B., Heiß, M. and Rüppel, U (2025) 'Optimisation of BIM collaboration format data analysis through advanced classification and information extraction.'. In: Francis, A., Miresco, E. and Melhado, S. (eds.) Advances in Information Technology in Civil and Building Engineering. Springer, pp.

Kaur, K. et al. (2018) 'Towards an open-standards based framework for achieveing condition-based predictive maintenance'. 8th International Conference on the Internet of Things IoT 2018. Santa Barbara, USA, ACM International Conference Proceedings Series.

Lee, G. (2011) 'What information can or cannot be exchanged?'. Journal of Computing in Civil Engineering 25 1-9.

Mayer, W. et al. (2013) Semantic interoperability in the oil and gas industry: a challenging testbed for semantic technologies. US: Association for the Advancement of Artificial Intelligence.

MIMOSA (2020) OIIE Oil and Gas Interoperability (OGI) Pilot. Available at: https://www.mimosa.org/ogi-pilot/ (Accessed: 12/02).

MIMOSA (2021) 'Open Industrial Interoperability Ecosystem (OIIE)'. MIMOSA.

MIMOSA (2025) 'Open O&M ISBM'. MIMOSA.

Pauwels, P., Zhang, S. and Lee, Y.-C. (2017) 'Semantic web technologies in AEC industry: A literature overview'. Automation in Construction, (73), pp. 145-165.

Postle, B. (2023) In BlenderBIM IFC Git collaboration. Available at: https://www.youtube.com/watch?v=cJZhSCSSWdA (Accessed: 1/7).

Sacks, R. et al. (2018) 'Chapter 3 Collaboration and Interoperability '. BIM Handbook: A guide to Building Information Modeling for Owners, Designers, Contractors and Facility Managers, 3rd Edition. Wiley, pp. 1-31.

Schevers, H. et al. (2007) 'Towards Digital Facility modelling for Sydney Opera House using IFC and Semantic Web Technology'. Electronic Journal of Information Technology in Construction, 12 347-362.

Selway, M. et al. (2015) Published. 'A conceptual framework for large-scale ecosystem interoperability'. 34th International Conference on Conceptual Modeling (ER 2015) 2015 Stockholm, Sweden. New York Springer, pp.287-301.

Selway, M. et al. (2017) Published. 'Level-aware ecosystem transformations for industrial lifecycle interoperability'. 36th International Conference on Conceptual Modeling, ER 2017 2017 Valencia, Spain. US Springer, pp.173-181.

Steel, J. et al. (2012) 'Model interoperability in building information modelling'. Software and systems modeling, 11 (1), pp. 99-109.

USACE (2011) Duplex Apartment. Available at: https://github.com/buildingsmart-community/Community-Sample-Test-Files (Accessed: 2/7).

Werbrouck, J. et al. (2019) 'Towards a Decentralised Common Data Environment using Linked Building Data and the Solid Ecosystem'. 36th CIB W78 2019 Conference. Newcastle, UK, Proceedings of the 36th CIB W78 Conference on Information Technology in Construction.

Widjaja, M. (2025) Difference ESB and API. Available at: https://www.itarch.info/2020/05/difference-between-esb-and-api-gateway.html (Accessed: 25/10).

ZBIM ZBIM Handbook Australia and New Zealand guide to ISO 19650 Part 1. Available at: https://brisbim.com/wp-content/uploads/2019/10/ANZ-Guide_ISO19650_Industry-Preview.pdf (Accessed: 11/07).

APPENDIX A

By Andrew Mcrae, 25th March 2024

 # How Proof-of-Concept demo was run

Overall

There are three concurrent instances of BlenderBIM used during the demo along with a server process called the bimserver_adaptor, plus BIM Server itself.

Git revision control

Git is required by the custom code in the addon. make sure command-line git is installed (and in the PATH) and works from any directory.

BlenderBIM setup

Blender 3.6 was used as the base.

The BlenderBIM ("BB") addon zip as at 24-01-24 was installed and then the source modified in place to add custom buttons and functionality.

* https://github.com/IfcOpenShell/IfcOpenShell/releases/tag/blenderbim-240124

This repo contains the modified files plus a large number of unmodified BB files in the bb240124 branch.

* https://github.com/amcrae/BlenderBIM_ISBM_Addon/tree/bb240124

It does not contain (for example) the lib folder and some data files from blender bim.

So...

1. Install the BB addon zip, then

2. git clone the repo and checkout bb240124 branch to temp folder.

3. copy across files from the repo over the top of BB.

...should reproduce BB config.

The version used in the Feb 28 demo still contained one hardcoded reference to the name of the project in BIM server, "Simple MEP", which is only used by the "get init model" button. This hardcoding has since been replaced with a reference to the project_name field of the isbmconfig file.

BIMserver

Java SE (1.8) was already installed.

A copy of BIM server 1.5.184 was installed locally.

* https://github.com/opensourceBIM/BIMserver/wiki/Download

* https://github.com/opensourceBIM/BIMserver/releases/download/v1.5.184/bimserverjar-1.5.184.jar

The design used in the demo was set up as a new project with two subprojects in BIMserver called:

* Simple1

* Simple Arch

* Simple MEP

The MEP project had to be called that name for the "get init model" feature to work from blender.

The Arch and MEP models (ifc files) are then uploaded with "Checkin.." to their matching projects.

BIMserver Adaptor

This is a basic server process which can serve the initial model checkout from BIM server and also do the central processing handling the Diff messages.

This repo contains the source of the bimserver_adaptor

* https://github.com/amcrae/BIM_Diff_Server/tree/main

This should be cloned and set up with a virtualenv.

With the venv active, the required libraries can then be installed with

`pip3 install -r pyreqs.txt`

Note the hpp-fcl takes a long time.

The username and password for the BIMserver needs to be hardcoded in the constants at the top of `bsapitest.py`.

ISBM config

Server address is hardcoded in `isbm_poc.py` (a file appearing in both the BB addon and bimserver_adaptor).

The session ids are assumed to be set up, connected, and stable so they can be used for messaging at any time. The channel and subscriber topics are listed in the `session_configs.txt` in `bimserver_adaptor` repo. These sessions will have to be set up in the ISBM server and the new session-ids copied into configuration files used by the demo in several places.

There are session IDs mentioned in the BB addon source but they are not used by current code.

The config files are called `isbmconfig.json` and need to be in :

* the bimserver_adaptor folder.

* The folder holding Blender A's MEP IFC file.

* The folder holding Blender B's MEP IFC file.

* The folder holding Blender C's Arch IFC file.

The consumer session ID to use is named `rcv_sessionid` in the JSON.

In the case of the Blender A and B config files, it also contains the name of the subproject in BIMServer that holds the MEP model. e.g.

```json
{
"my_id": "blenderA",
"rcv_sessionid": "40b67d96-09e0-4eb6-89e6-b156bfcb5943",
"pub_sessionid": "3e86eef3-9b12-4e1b-bd21-b60bf2d04db0",
"project_name": "Simple MEP"
}
```

As Blender C only checks for clashes it only needs the rcv_sessionid.

IFC file folders

Each instance of blender will need to load an IFC file from a different folder. The folder is also used to make temporary files.

In the case of blender B, whichever IFC file is opened will be overwritten when it receives the diff update.

The IFC files were usually called 'the_model.ifc' as their exact content depended on what they received from the bimserver_adaptor.

Each folder needs an `isbmconfig.json` as described above.

The folder for Blender A must also be a git repo for the diffing to work, and will be converted to git repos by the "get init model" button action.

Example folder structure (without git repo yet) is in `IFCs_Simple.zip`

Running BIMserver Adaptor

With the venv active, running `python3 bsadaptor.py scan` will scan for incoming messages and handle them.

This can be done before the demo to make use of the "get init model" feature which will request a download of the model from a named project in BIMserver and serve it to the channel.

Mainly the server will handle SyncIFCDiff topic messages.

Using the custom buttons in BlenderBIM

When starting blender, run it from the command line to be able to see the diagnostic print statements in the console.

The buttons are ordered top-to-bottom and named according to where they are used in the demo sequence, however not all buttons can be used by all blender instances.

Get Channels. can be used to check communication with the ISBM, as all channel names are dumped to the console.

Check Channel. Used to check the channel used in the demo is present by looking up its name.

The remaining buttons can only be used after opening an IFC project file.

(0.) get init model. Used by Blender A and B to download the same version of their model initially AND convert the ifc folder to a Git repository. Do this for blender A and B before the demo is run. Make sure the bimserver_adaptor is running.

(1.) Publish Changes. This is used only by Blender A. It publishes the SyncIFCDiff message.

(2.) Check for Changes. This is used only by Blender B. It consumes the SyncIFCDiff message.

(3.) Check for Clashes. This is used on Blenders A,B,C to get the last clash status.

The sequence is definitely Publish by A, Check by B. After publishing changes from A you must not consume/Check for changes on A.

Clashes can be checked for at any time but it is best to wait for the adaptor process to finish publishing the ShowClash message. The highlighting will only happen on the receipt of the clash message; the objects are cleared back to white if you try to check for clashes again.