2022 European Conference on Computing in Construction
Ixia, Rhodes, Greece
July 24-26, 2022

DATA INTEGRATION FOR DIGITAL TWINS IN THE BUILT ENVIRONMENT

Jorge Merino[12], Xiang Xie[12], Nicola Moretti[12], Janet Yoon Chang[1], and Ajith Parlikad[12]
[1]University of Cambridge, Cambridge, United Kingdom
[2]Centre for Digital Built Britain, University of Cambridge, Cambridge, United Kingdom

Abstract

Digital Twinning is an emerging technology in AEC/FM for the efficient operation of the built environment where data is often isolated in BIM, Building Automation Systems (BAS) and documental databases. This paper proposes a method based on federated data models and ontologies to standardise the integration of construction data (IFC), BAS (Brick Schema), and IoT using data lake architectures while keeping the delegation of responsibilities of data owners. A case study in the Alan Reece building demonstrates the approach by enabling Fault-Detection-and-Diagnosis of the HVAC system for facility management. Integration of all built environment data is crucial for efficient operation.

Introduction

Efficient building operation is one of the main challenges in the Architecture, Engineering, Construction and Facility Management (AEC/FM) sectors nowadays. Net-zero carbon objectives force restrictive goals and constraints for the efficient use of energy in the built environment (Kazmi et al. 2014, Ufuk Gökçe & Umut Gökçe 2014). Traditionally, building operations are characterised by independent systems functions with the focus on the effective run of equipment. Even that huge improvements in the efficient use of energy have been identified, most focus in individual systems performance by single-point or distributed monitoring (Kazmi et al. 2014). Other authors have addressed the lack of communication between the operations of the different systems of a building, but the full potential remains to be seen (Ufuk Gökçe & Umut Gökçe 2014, Dong et al. 2014, Oti et al. 2016). Digital twin technologies are rising as the facilitators of integrated building operations in AEC/FM throughout the life-cycle of buildings, particularly during the operation and maintenance (O&M) phase (Dong et al. 2014).

Digital twins represent the natural convergence of emerging technologies in the AEC/FM industry. Building Information Modelling (BIM) changed the way built environment information is created, stored and exchanged between involved stakeholders (Howell & Rezgui 2018). The Industry Foundation Classes (IFC) brought methods for construction data sharing which are being adopted industry-wide (Autodesk Inc. 2021, Perttula & Suchocki 2020, Building Smart Int. 2022, ISO 2015). Both industry and academia have started considering the representation of assets and systems in the built environment as part of BIM (Dave et al. 2018, Tomasevic et al. 2015). For instance, Brick Schema, Haystack, and Building Topology Ontology provide standardised representation of the physical, logical, and virtual assets in buildings and the relationships between them (Balaji et al. 2016, Haystack 2021, Rasmussen et al. 2021). Recently, digital technologies like the Internet of Things (IoT) and Artificial Intelligence (AI) are pushing BIM towards adjacent research areas throughout the entire built environment life-cycle, at building, infrastructure and city levels (Boje et al. 2020, Sotres et al. 2017).

The main challenge in the development of a digital twin for a built environment is the natural segregation of data storage(Hu et al. 2016). Data is modelled to meet independent systems requirements (e.g., building management system, asset management system, occupancy, design and construction data, etc.) rather than as part of the overarching built environment entity (Corry et al. 2015). Furthermore, systems and buildings components information is often outdated, incomplete, and inaccurate when exchanged along the assets' life cycles (e.g., from design and construction to operations and management) (O'Donnell et al. 2013). Data integration and management become crucial for digital twinning of the built environment.

Data Lakes architectures are considered as a comprehensive approach for data management in distributed information systems (Kumar et al. 2018). Data transformations are conducted in data pipelines in lake architectures, which are processes to prepare, clean, and give access to data considering the individual information requirements of the data consumers and applications. Numerous digital twins have successfully been implemented as data lakes in the literature, but they can quickly turn into data swamps without appropriate management (Raj & Surianarayanan 2020). Extract, Transform and Load (ETL/ELT) has been present in industrial systems for decades, however, data transformations in the AEC context cannot be achieved effortlessly (Adnan & Akbar 2019, Hu et al. 2016). The completeness and accuracy of the geometries and semantics are common issues in the ETL process for construction data (Sani & Rahman 2018). The need for data fidelity (i.e., preserving raw data to avoid information loss) creates multiple versions of data, which induces a high risk of data inconsistency in data lakes (Sawadogo & Darmont 2021). On-demand schema mapping on large variety of sources in a data lake is also an arduous effort while integrating data in the pipelines (Nargesian et al. 2019).

Semantic web approaches drive data integration while achieving broad classification and description of built environment entities (Pauwels et al. 2017, Corry et al. 2015). Ontologies like ifcOWL or Brick Schema have standardise the way in which construction and systems data is modelled, but neither incorporate all the intricacies and complexities of buildings. Still, the effort of creating an ontology that accommodates all domains (e.g., BIM, BAS, ...) becomes unmanageable when new domain data is incorporated in the data lake. These ontologies are extended and extended until they become hard to understand (Zhe et al. 2006). The data pipelines that integrate data based on these ontologies need re-engineering when those extensions and changes happen. Federated data models promote the domain-specific independence of data sources while finding appropriate links between ontologies to standardise integration of data (O'Donnell et al. 2013, Gerrish et al. 2017).

This paper proposes a method based on federated data models to integrate construction and system information based on data pipelines. A data lake architecture is used to manage the flow of information from the original data sources to the OM&R applications and data users. The technical aspects and methods for the integration of data in a digital twin in the built environment are the main focus of the paper. Techniques are demonstrated in a case study conducted on the digital twin of the Alan Reece building of the University of Cambridge. The case study demonstrates the integration of construction data, systems and asset information, and sensor readings for a fault detection and diagnose (FDD) application of the building Heating, Cooling and Air Conditioning (HVAC) system.

Methodology

The method presented in this section assumes that data is structured using federated data models, and stored in independent data repositories. The data lake architecture and data sources are part of the Digital Twin data platform and the processes for integration are driven by data pipelines.

Digital twin data platform

The digital twin for the built environment platform is composed of a number of data services for data ingestion, management and access, shaping the architecture of a data lake. This type of architectures enables data ingestion from diverse sources using independent Extract, Transform, Load (ETL/ELT) processes.

The architecture proposed in (Brazauskas et al. 2021), called Adaptive City Platform (ACP), was implemented to ingest and manage data from multiple real-time sources (see figure 1), including IoT sensors (e.g., LoRaWan, radio-frequency, WiFi) and a building management system that governs the mechanical, electricity and plumbing systems of the target building. The ACP is engineered towards minimising the end-to-end latency for real-time data, averaging a few milliseconds between a data entry (i.e., when it is ingested) and exit (i.e., when it is available for use). This is particularly important in the built environment to visualise the real state of assets and spaces and for the early identification of potential problems in Operation Maintenance and Repair (OM&R) (Boje et al. 2020).

Other data sources may coexist in the data lake, but they are not necessarily ingested through the ACP. Reference data of the built environment consists of static floor plans, maps and/or 3D models (e.g., CAD model) of the building structures; blueprints of mechanical, electrical, and plumbing systems (e.g., HVAC system), or other representations of their functional dependencies like the BrickSchema; and asset catalogs to record all the built assets. Transactional data refers to semi-static information about status of assets, maintenance work orders, such as the inspection condition and date of assets. All data (real-time and static) is stored in its raw format as it converges to the data lake (i.e., ELT). Data lakes architectures facilitate the ingestion of high-variety data as well as high and low velocity data.

Many data lakes use a standardised but flexible data model to structure data across different sources, and engineer it for integration. This is particularly effective for metadata since it often comes in proprietary or open formats that are designed for visualisation/inference rather than fast querying. The ACP suggests a data modelling strategy governed by crates (Brazauskas et al. 2021) to model building information. A crate is an entity (e.g., a sensor, an equipment, a space) with its own attributes plus zero or more parents (see figure 2). Parents are referenced in the crate approach rather than been nested. Thus, crates form a hierarchical structure, but every crate is uniquely identified through an indexed key for quick access. This is particularly effective to represent the topologies and functional hierarchies of buildings, their systems as well as sensor data. Documental databases are chosen to hold the crate model, and JavaScript Object Notation (JSON) format is used in the ACP. Listing 1 shows an example of the crate model for sensor data. Transformed data into the crate model is also stored in the ACP through a message filer that collects data into a day-level for every sensor (i.e., ELTL). Raw data is kept for traceability and repeatability.

Listing 1: Crates model example for sensor data

```
"sensor-temperature-123456": {
  "acp_id": "sensor-temperature-123456",
  "type": "sensor",
  "features": ["temperature"],
  "parents": [ { "parent_id": "ifm-space-01",
      "parent_type": "space"} ] },
"sensor-vibration-789012": {
  "acp_id": "sensor-vibration-789012",
  "type": "sensor",
  "features": ["x", "y", "z"],
  "parents": [{ "parent_id": "ifm-pump-01",
      "parent_type": "equipment" } ] }
```

The ACP enables access to data in real-time both through HTTP POST to the desired http destination URL, and by accepting data subscriptions through websockets (see figure 1). Access to other sources of data needs to be enabled by APIs. In both cases, data is engineered through data pipelines in a data lake.

Data Pipelines

A data pipeline is a piece of software that sits between the data lake sources and the data consumers to extract, transform, and integrate available data on demand. Data

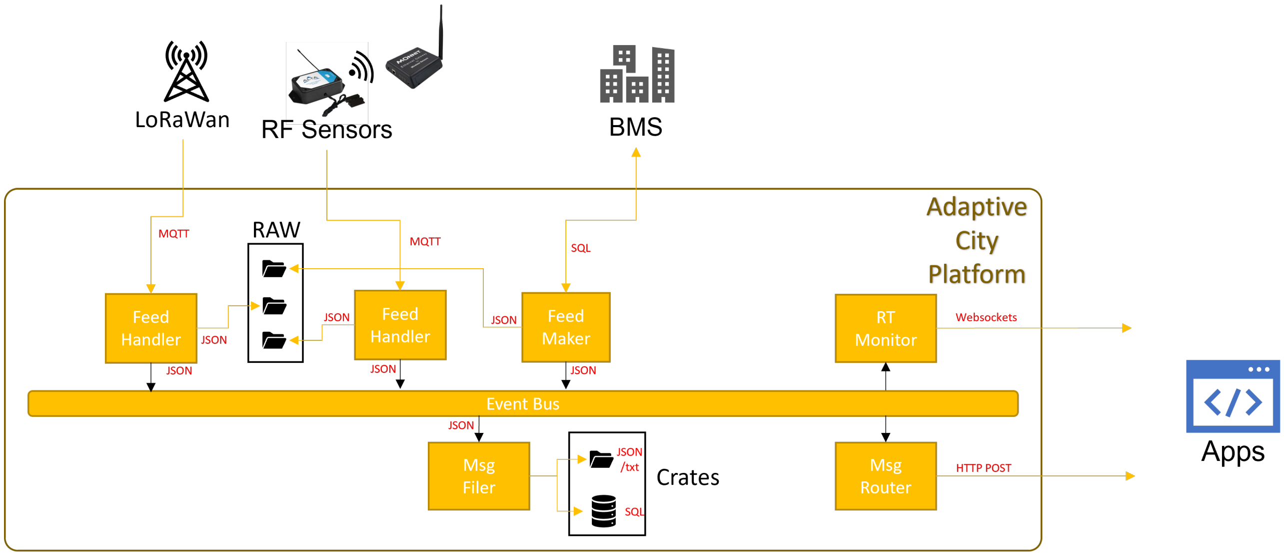

Figure 1: Adaptive City Platform (ACP) (Brazauskas et al. 2021)

crate_id	features	crate_type	parents
sensor_temperature-123456	[Temperature]	Sensor	[ifm-space-01, ifm-zone-z1]
ifm-space-01	[]	Space	[ifm-floor-GF]
ifm-zone-z1	[]	Space	[ifm-floor-GF]
ifm-floor-GF	[]	Space	[ifm]
sensor_vibration-789012	[X, Y, Z]	Sensor	[ifm-pump-a]
ifm-pump-a	[]	Equipment	[ifm-hvac]

Figure 2: Adaptive City Platform (ACP) crates data model (Brazauskas et al. 2021)

consumers/applications identify information requirements to run (e.g., required data points, input format, pace, etc.) which is fulfilled by data pipelines. A data pipeline can be reused by different applications and different data pipelines can be combined to serve a new application if required (see figure 3).

Sensor data needs of the static sources to add contextual information (e.g., sensor location, built environment subsystem, etc.) before applications make use of it. Without the context, sensor data lacks meaning since the data they generate is a measure of a feature or an event that belongs to other entities. For instance, an IoT sensor produces temperature readings, but that temperature is the property of an space. Similarly, a vibration sensor in the Building Management System produces vibration frequencies in the X, Y, Z to represent the movement of a pump in the HVAC system.

The creation of data pipelines starts with the identification of information requirements (Pishdad-Bozorgi et al. 2018). It is essential to understand what data the data consumers and applications use, how often and what format it is required (Kang & Choi 2015, Becerik-Gerber et al. 2012). Sometimes, existing pipelines can be reused adding a new layer of data transformation (e.g., if a different format is needed) or integration (e.g., if additional data needs to be combined). If no existing pipeline can provide it, then a new one must be designed from scratch. In such case, it is also necessary to identify the sources of data in the lake for the data pipeline to extract, transform and combine data. Ultimately, the data access is made available by data pipeline through an API in the format and pace required. In this research, authentication is not considered. Figure 3 visualises how information requirements become the inputs and the API becomes the output of data in a data pipeline. This process is also known as data engineering.

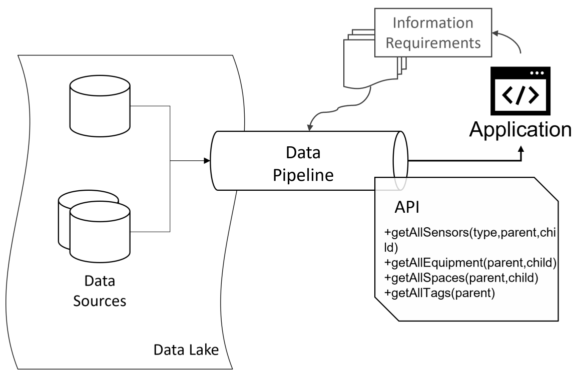

Figure 3: Data pipeline design: Inputs and outputs

Ontologies are widely used to drive the integration of diverse sources (Corry et al. 2015, Hu et al. 2016). Even that the ACP data strategy is used to govern the flow of real-time data, there are two ontologies that help understanding sensor data in the built environment: the ISO 12006 (ISO 2015) (Industry Foundation Classes - IFC data model) and Brick Schema.

IFC is a standard building data model widely used for interoperability between built environment design applications and systems. Model View Definitions (MVD) improve the data exchange between BIM and Building Automation

Systems. As an example, some of the Building Automation and Control Networks (BACnet) assets are modelled as instances in IFC (Tang et al. 2020) to enable better visualisation of the automated systems (e.g. HVAC room diffusers). IFC excels on 3D modelling (i.e., drawing all the structures and components with a lot of detail), and thus it has a complex representation the topologies and the architectural hierarchies, which can be inferred by traversing the IFC elements and relationships. (Moretti et al. 2020) shows how IFC meta-information about hierarchy and topology of the built environment can be extracted. An IFC2ACP data pipeline was developed to read the IFC files and infer the topologies and architectural hierarchies using the IFCOpenShell python API (*IfcOpenShell* 2021). Then it transforms that information into the ACP crates model and store it in memory in a JSON object to enable indexed access to all the elements.

Listing 2: Example of the ACP representation of an IFCSpace

```
"103": {
  "crate_id": "103",
  "crate_type": "space",
  "acp_ts": 1629813271.922424,
  "acp_localtime": "2021-08-24T14:54:31.9224",
  "ifc_id": "0UIH5Blo19ohldZ0jJVrWM",
  "ifc_type": "IfcSpace",
  "parent_crate_id": "GF-basement",
  "ifc_geometry": {
    "ifc_geometry_type": "IfcExtrudedAreaSolid",
    "ifc_location": [71474.9958300488,29669.3674420791,-150.0],
    "ifc_depth": 2375.0,
    "ifc_sweptarea": {
      "type": "IfcArbitraryClosedProfileDef",
      "points": [
        [-2099.99980792596,-1137.49952716191],
        ...,
        [-2099.99980792596,-1137.49952716191]]
    }
  }
}
```

Brick Schema facilitates the comprehensibility of the functional hierarchies of the systems in a built environment from a human perspective. The design of Brick ontology mainly focused on defining the physical, logical, and virtual building assets with the emphasis on building operations, such as equipment or sensors in lighting, submetering, and HVAC systems (Balaji et al. 2016). Leveraging its strong expressability of building system hierarchies, the Brick Schema has been adopted in real cases like (Xie et al. 2021) to represent building metering system hierarchies, and to connect sub-metering readings with spatial characteristics for fine-grained energy analysis. However, Brick Schema models are networks that contain many cycles to determine connections between entities. Traversing these cyclical networks in real-time can be costly in latency. (Xie et al. 2021) shows how Brick Schema meta-information on the systems in the built environment can be transformed into the ACP crates data model for real-time applications. The Brick Schema files (i.e., turtle or TTL) are read in the Brick2ACP data pipeline using the brickschema python API (*Brick Ontology Python package* 2021), and transformed into the ACP crates data model to avoid that increase in latency. The ACP crates version of the Brick Schema meta-information is stored in JSON in memory for quick access.

Listing 3: Example of the ACP representation of a BrickSchema location

```
"103": {
  "location_name": "http://ifm.cam.ac.uk/demo_building#103",
  "type": "https://brickschema.org/schema/Brick#Room",
  "number_of_points": 0,
  "number_of_equipment": 0,
  "parents": [{
      "parent_id": "http://ifm.cam.ac.uk/demo_building#Floor_1",
      "type": "https://brickschema.org/schema/Brick#isPartOf",
      "parent_type": "location"
    },{
      "parent_id": "http://ifm.cam.ac.uk/demo_building#Zone_103",
      "type": "https://brickschema.org/schema/Brick#isPartOf",
      "parent_type": "location"
    },{
    "parent_id": "http://ifm.cam.ac.uk/demo_building#LZone",
      "type": "https://brickschema.org/schema/Brick#isPartOf",
      "parent_type": "location"
    }]
}
```

The lingering question is how to integrate both ontologies since some applications may need to make use of metadata from both IFC and Brick Schema. Luckily, there is no need to create a meta-ontology to combine IFC and Brick Schema since they both handle the concept of a Space (in IFC) or Location (in BrickSchema) and Sensors (in IFC) or Points (in Brick Schema) which can be used as a nexus for the integration. A IFCxBrick data pipeline was implemented to integrate the tailored ACP versions IFC and Brick Schema data based on the common elements found in both ontologies. Transparent access to integrated data is enabled through an API. The API allows data consumers to access all elements in the building. It is also possible to query the elements inside the known element (i.e., children; e.g., all the sensors in a location) as well as the elements to which the known elements belongs to (i.e., the parents; e.g., the location of an asset/equipment).

Listing 4: Example of the integration of IFC and Brickschema through the ACP data model

```
"103": {
    "acp_id": "103",
    "crate_type": "space",
    "crate_id": "103",
    "ifc": {
      "crate_id": "103",
      "crate_type": "space",
      "parent_crate_id": "GF-basement",
      ... see IFC listing ...,
    },
```

```
    "brick": {
      "location_name": "http://ifm.cam.ac.uk/
          demo_building#103",
      "type": "https://brickschema.org/schema/
          Brick#Room",
      ... see BRICK Schema listing ...
    }
}
```

With all three parts of the puzzle available and indexed, any application can open a websocket client connected to the ACP platform to subscribe to sensor data in real-time (see figure 4) and request contextual metadata from the IFCxBrick API.

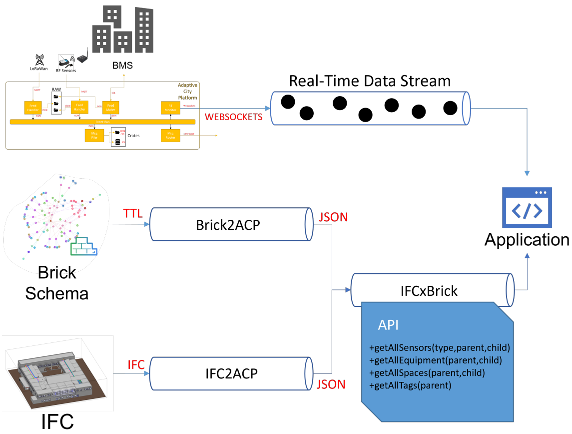

Figure 4: Data pipelines: ifc2acp, brick2acp and ifcxbrick, and the real-time connection with websockets

Whereas semantic web approaches like centralised triple stores focus on data exploration, this approach is tweaked towards real-time data reporting. Provided APIs can also be used for exploration, but they are not specifically designed for data discovery. Linked Building Data approaches like the BOT ontology (Rasmussen et al. 2021) could also be adopted as additional federated models.

Case Study: Institute for Manufacturing

This approach has its practical application in the digital twin pilot of Alan Reece building at the West Cambridge site. The Alan Reece building is a 3-storey building and stands over a 3800-square-meter comprehensive area, including spaces for teaching, office, research, laboratory, canteen, etc. Figure 5 shows the 3D model of the Alan Reece building. The digital twin is geared to support building operations and asset management. Among the applications enabled by the digital twin, this paper focuses on fault detection and diagnosis functionality for building HVAC systems. Some of the spaces in the Alan Reece building are conditioned with Variable Refrigerant Flow (VRF) system, connecting to multi-zone indoor air conditioning units in a multi-split manner. Functionally, the VRF and indoor units provide heating and cooling the building, serving multiple seminar rooms and lecture theatres.

The practical setting for this case study is an HVAC zone comprising two seminar rooms where an automated FDD

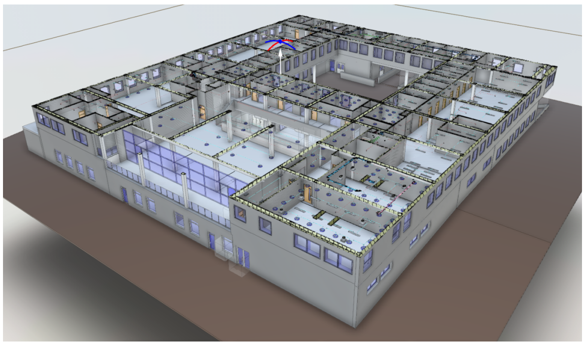

Figure 5: 3D Model of the Alan Reece building

application identifies anomalies in the comfort temperature of the spaces (see figure 6) using real-time analytics based on sensor and contextual data (i.e., IFC and BrickSchema). The air conditioning system of the seminar rooms is pictured in figure 7, and it is composed of an VRF that feeds the indoor units connected to the seminar rooms in the target HVAC zone. This application make use of temperature, humidity and dew-point, open-closed (for windows and doors) data from IoT sensors in the seminar rooms, plus operational data of the HVAC system from the BMS. Figure 7 shows the sensors (or points) in the BMS monitored by this application.

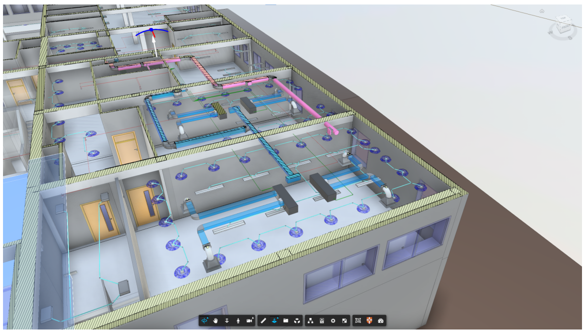

Figure 6: 3D model of the Seminar Rooms 2 and 3

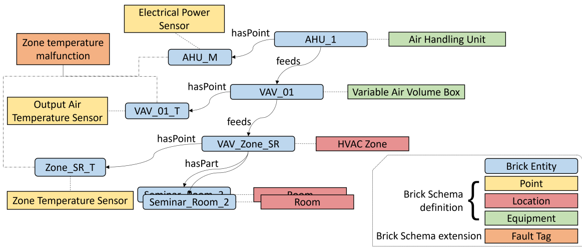

Figure 7: BrickSchema model of the HVAC system feeding Seminar Rooms 2 and 3

For simplicity, this case study targets the Zone Temperature Malfunction fault (see figure 7). This fault refers to anomalies found on the comfort temperature of the seminar rooms' HVAC zone. If the temperature monitored by the IoT sensor exceeds the comfort interval, the FDD triggers an investigation using available data. First of all, it checks the status of the windows and doors of the space, and the HVAC Zone temperature. If there is an anomaly,

it is necessary to diagnose the sources, including potential impact to critical assets further up in the hierarchy of the HVAC system. Thus, operational status of the indoor units feeding the seminar rooms and the VRF needs to be checked accessing the BMS.
The information required for the FDD application consist of the real-time from the IoT sensors and BMS sensing points, the relationships between faults, assets and spaces in the HVAC from BrickSchema, and the topology of the rooms from IFC. Real-time data is ingested and managed by the Adaptive City Platform (ACP) which runs in a custom server. Reference data (i.e., IFC and Brick Schema models) is also stored in the same custom server in its original format. The IFC2ACP, Brick2ACP, and IFCxBrick data pipelines are responsible for the integration and online APIs are made available for all three data pipelines. The ACP enables real-time data requests and subscription through websockets.
Figure 8 shows the usage of the IFCxBrick data pipeline to discover and diagnose the zone temperature malfunction fault. The FDD application monitors the sensors related to the zone temperature malfunction fault. The sequence starts with the FDD application querying the function getAllSensors indicating the fault identificator. The function will return the list of sensors related to this fault. Subsequently, the FDD application can subscribe to the real-time data of the sensor list. Then, on every message that arrives to the FDD the following steps may be triggered:

1. The FDD application needs to know the type of sensor that the message is coming from. The type of sensor is in the body of the message coming from the ACP, but it can also be queried with the function getAllSensors of the IFCxBrick indicating the sensor id. In this example, we assume it is an IoT sensor which are located only in spaces and never in equipment in the Alan Reece Building.

2. The FDD application queries the function getAllSpaces of the IFCxBrick API with the sensor id to know what space (i.e., seminar room 2) that IoT sensor belongs to (i.e., its parent). If the temperature of the seminar 2 exceeds the comfort interval, the investigation is triggered.

3. The FDD application will check other sensors in that space by querying the function getAllSensors of the IFCxBrick API with the list of spaces returned in the previous step. It also requests the last readings from the ACP through the websocket.

4. A crosscheck including the original reading and the new sensor readings is performed to identify the source of the problem. All the sensors reporting a problem in the crosscheck must be investigated. The functions getAllEquipment and getAllSpaces from the IFCxBrick API can be queried again to know more about the HVAC system functional relationships (e.g., what HVAC Zone the space belongs to, what indoor unit feeds the HVAC Zone, which VRF feeds the corresponding indoor units, etc.; see figure 7), as well as the topology and the hierarchy of the building (e.g., what sensors are in a space, or an equipment, what spaces belong to other spaces, etc.; see figure 6).

5. Similarly to step 3, the last sensor readings of the spaces and equipment that are subject to further investigation can be requested to the ACP. The last two steps are repeated until the source of the fault is identified.

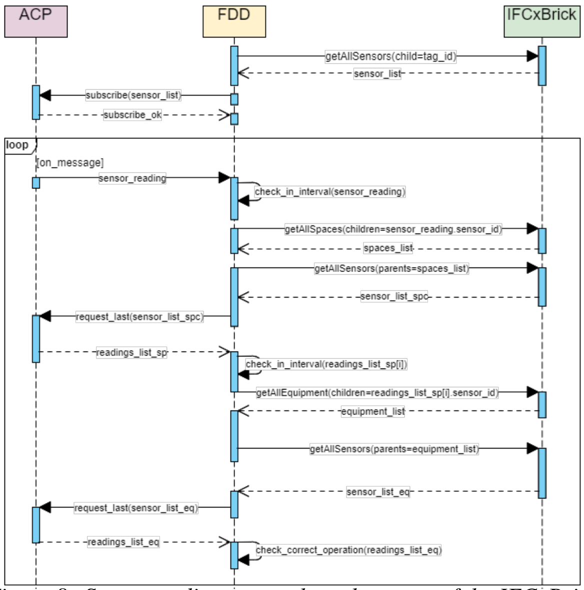

Figure 8: Sequence diagram to show the usage of the IFCxBrick API and the ACP websockets in an FDD application

The faults discovered by the FDD and the diagnosis can be reported through the required methods to assist facility management. In this case, the FDD reports via email to the facility manager, and the faults are visualised in the 3D model of the building by flashing the affected assets and spaces. This 3D model is hosted in a virtual environment in the university network.
Important lessons were learned during this case study. BAS in buildings are managed by different roles and even departments, and serviced by different companies, reducing data availability. In most cases data is only available as spreadsheets or documents that require pre-processing (e.g., asset maintenance records, hand-over documents of buildings). Information changes are not often documented and original documents are still used as reference instead, affecting data accuracy and timeliness. It is easier to get a one-time dump of the information, but not on-demand. Even when on-demand access is enabled, the technical aspects of data engineering appear. The high-variety of BAS, the manifold applications and users requirements make the design of pipelines an ordeal. Further, the duplication of pipelines is likely if there is not appropriate planning (e.g., shared features identified). Federated data models helped

to manage this complexity enabling the design of tailored models in the development of manageable data pipelines.

Conclusions

AECO industry is steering in the direction of the digitalisation of buildings from the design and construction phase to ensure efficient building operations. Digital twins are seen as enabler for smart buildings operations, but they can only achieve their full potential when all available sources of data are integrated.

In this sense, one of the biggest challenges is to enable on-demand access to Building Automation Systems (BAS) which ownership and management is often shared. Obtaining real on-demand data becomes an arduous cyclical process of requesting more and more access grants, since managers are not always willing to facilitate it as the main responsible of the data. Senior asset managers must become facilitators in the development of the Digital Twin (Gerrish et al. 2017). The technical challenges of data engineering explode with high-variety data and uses, which may cause duplication of pipelines without adequate planning.

This paper demonstrate how to integrate the functional relationships of building systems, the topology, and the architectural hierarchy of buildings and real-time information to enable dynamic asset management applications. Data lake architectures unlocks the data sitting in silos, while the use of federated data models helps with the delegation of data responsibilities. The data pipeline design method presented illustrates how to elicit information requirements of asset management applications in order to identify original data sources and data transformations and combinations to meet them. The low-latency ACP crates data model is combined with IFC and Brick Schema to create a tailored data model for the pipelines to enable data access in real-time in the case study. It evidences how this integration methods can aid not only condition monitoring and prognosis, but also enhanced visualisation to promote facility management.

The insights and reports provided by the applications can serve as new sources of information, but it is necessary to understand how to re-purpose them to enable new analysis and insights. Federated data models may need to be extended to accommodate newly created asset management knowledge. Whereas data lakes support the design of digital twins for the built environment, data access to original sources will remain as a big challenge because of the chain of responsibility of data. Implementation of DTs in early stages of buildings life-cycle can help eliminate this burden, and can become the first step towards servitisation in operations.

Acknowledgments

This research forms part of the Centre for Digital Built Britain's (CDBB) work at the University of Cambridge within the Construction Innovation Hub (CIH). The Construction Innovation Hub is funded by UK Research and Innovation, UK through the Industrial Strategy Fund.

References

Adnan, K. & Akbar, R. (2019), 'Limitations of information extraction methods and techniques for heterogeneous unstructured big data', *International Journal of Engineering Business Management* **11**.

Autodesk Inc. (2021), 'BIM Interoperability | openBIM & buildingSMART | Autodesk'.
URL: *autodesk.com/industry/aec/bim/interoperability*

Balaji, B., Bhattacharya, A., Fierro, G., Gao, J., Gluck, J., Hong, D., Johansen, A., Koh, J., Ploennigs, J., Agarwal, Y. et al. (2016), Brick: Towards a unified metadata schema for buildings, *in* 'Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments', pp. 41–50.

Becerik-Gerber, B., Jazizadeh, F., Li, N. & Calis, G. (2012), 'Application Areas and Data Requirements for BIM-Enabled Facilities Management', *Journal of Construction Engineering and Management* **138**(3), 431–442. Publisher: American Society of Civil Engineers.

Boje, C., Guerriero, A., Kubicki, S. & Rezgui, Y. (2020), 'Towards a semantic Construction Digital Twin: Directions for future research', *Automation in Construction* **114**.

Brazauskas, J., Verma, R., Safronov, V., Danish, M., Merino, J., Xie, X., Lewis, I. & Mortier, R. (2021), 'Data management for building information modelling in a real-time adaptive city platform', *arXiv preprint arXiv:2103.04924* .

Brick Ontology Python package (2021).
URL: *https://github.com/BrickSchema/py-brickschema*

Building Smart Int. (2022), 'buildingSMART - The International Home of BIM'.
URL: *www.buildingsmart.org*

Corry, E., Pauwels, P., Hu, S., Keane, M. & O'Donnell, J. (2015), 'A performance assessment ontology for the environmental and energy management of buildings', *Automation in Construction* **57**, 249–259.

Dave, B., Buda, A., Nurminen, A. & Främling, K. (2018), 'A framework for integrating BIM and IoT through open standards', *Automation in Construction* **95**, 35–45.

Dong, B., O'Neill, Z. & Li, Z. (2014), 'A BIM-enabled information infrastructure for building energy Fault Detection and Diagnostics', *Automation in Construction* **44**, 197–211.

Gerrish, T., Ruikar, K., Cook, M., Johnson, M., Phillip, M. & Lowry, C. (2017), 'BIM application to building energy performance visualisation and management: Challenges and potential', *Energy and Buildings* **144**, 218–228.

Haystack (2021), 'Project Haystack'.
URL: *https://project-haystack.org/*

Howell, S. & Rezgui, Y. (2018), *Beyond BIM: Knowledge management for a smarter built environment*, BRE Electronic Publications.

Hu, S., Corry, E., Curry, E., Turner, W. J. N. & O'Donnell, J. (2016), 'Building performance optimisation: A hybrid architecture for the integration of contextual information and time-series data', *Automation in Construction* **70**, 51–61.

IfcOpenShell (2021).
URL: *http://ifcopenshell.org/*

ISO (2015), 'ISO 12006-2:2015 Building construction — Organization of information about construction works Part 2: Framework for classification'.

Kang, T.-W. & Choi, H.-S. (2015), 'BIM perspective definition metadata for interworking facility management data', *Advanced Engineering Informatics* **29**, 958–970.

Kazmi, A. H., O'grady, M. J., Delaney, D. T., Ruzzelli, A. G. & O'hare, G. M. P. (2014), 'A Review of Wireless-Sensor-Network-Enabled Building Energy Management Systems', *ACM Trans. Sen. Netw.* **10**(4), 66:1–66:43.

Kumar, S. A. P., Madhumathi, R., Chelliah, P. R., Tao, L. & Wang, S. (2018), 'A novel digital twin-centric approach for driver intention prediction and traffic congestion avoidance', *J Reliable Intell Environ* **4**(4), 199–209.

Moretti, N., Xie, X., Merino, J., Brazauskas, J. & Parlikad, A. K. (2020), 'An openBIM Approach to IoT Integration with Incomplete As-Built Data', *Applied Sciences* **10**(22), 8287.

Nargesian, F., Zhu, E., Miller, R. J., Pu, K. Q. & Arocena, P. C. (2019), 'Data lake management: Challenges and opportunities', *Proc. VLDB Endow.* **12**(12), 1986–1989.

Oti, A. H., Kurul, E., Cheung, F. & Tah, J. H. M. (2016), 'A framework for the utilization of Building Management System data in building information models for building design and operation', *Automation in Construction* **72**, 195–210.

O'Donnell, J., Corry, E., Hasan, S., Keane, M. & Curry, E. (2013), 'Building performance optimization using cross-domain scenario modeling, linked data, and complex event processing', *Building and Environment* **62**, 102–111.

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

Perttula, T. & Suchocki, M. (2020), 'Development Progress on IFC for Infrastructure | Autodesk University'.
URL: *autodesk.com/autodesk-university/es/class/Development-Progress-IFC-Infrastructure-2020*

Pishdad-Bozorgi, P., Gao, X., Eastman, C. & Self, A. P. (2018), 'Planning and developing facility management-enabled building information model (FM-enabled BIM)', *Automation in Construction* **87**, 22–38.

Raj, P. & Surianarayanan, C. (2020), Chapter Twelve - Digital twin: The industry use cases, *in* 'Advances in Computers', Vol. 117 of *The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases*, Elsevier, pp. 285–320.

Rasmussen, M. H., Pauwels, P., Lefrançois, M. & Schneider, G. F. (2021), 'Building Topology Ontology'.
URL: *https://w3c-lbd-cg.github.io/bot/*

Sani, M. & Rahman, A. (2018), GIS and BIM integration at data level: A review, *in* 'International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives', Vol. 42, pp. 299–306.

Sawadogo, P. & Darmont, J. (2021), 'On data lake architectures and metadata management', *Journal of Intelligent Information Systems* **56**(1), 97–120.

Sotres, P., Santana, J. R., Sánchez, L., Lanza, J. & Muñoz, L. (2017), 'Practical Lessons From the Deployment and Management of a Smart City Internet-of-Things Infrastructure: The SmartSantander Testbed Case', *IEEE Access* **5**. Conference Name: IEEE Access.

Tang, S., Shelden, D. R., Eastman, C. M., Pishdad-Bozorgi, P. & Gao, X. (2020), 'BIM assisted Building Automation System information exchange using BACnet and IFC', *Automation in Construction* **110**(November 2019), 103049.

Tomasevic, N. M., Batic, M. C., Blanes, L. M., Keane, M. M. & Vranes, S. (2015), 'Ontology-based facility data model for energy management', *Advanced Engineering Informatics* **29**(4), 971–984.

Ufuk Gökçe, H. & Umut Gökçe, K. (2014), 'Integrated System Platform for Energy Efficient Building Operations', *Journal of Computing in Civil Engineering* **28**(6), 05014005.

Xie, X., Moretti, N., Merino, J. & Chang, J. Y. (2021), Ontology-based spatial and system hierarchies federation for fine-grained building energy analysis, *in* 'Proc. of the Conference CIB W78', Vol. 2021, pp. 11–15.

Zhe, Y., Zhang, D. & YE, C. (2006), Evaluation metrics for ontology complexity and evolution analysis, *in* '2006 IEEE International Conference on e-Business Engineering (ICEBE'06)', pp. 162–170.