ITcon

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

A TOOL-BASED SYSTEM ARCHITECTURE FOR A DIGITAL TWIN: A CASE STUDY IN A HEALTHCARE FACILITY

SUBMITTED: August 2022
REVISED: January 2023
PUBLISHED: February 2023
EDITOR: Robert Amor
DOI: 10.36680/j.itcon.2023.006

Ashit Harode, Ph.D. Candidate
Department of Building Construction, Virginia Tech, Blacksburg, VA, USA;
ashit02@vt.edu

Walid Thabet, Ph.D., CM-BIM
W.E. Jamerson Professor, Department of Building Construction, Virginia Tech, Blacksburg, VA, USA
thabet@vt.edu

Poorvesh Dongre, Ph. D. Student
Department of Computer Science, Virginia Tech, Blacksburg, VA, USA
poorvesh@vt.edu

SUMMARY: *Changes in the local and global markets are forcing A/E/C/FM (Architecture, Engineering, Construction, and Facility Management) organizations to deliver more robust and innovative operational BIMs (Building Information Models). It is hypothesized that BIMs will transform from a static 3D model to a Digital Twin providing a truly digital representation of the physical asset or the building it represents. This transformation to a dynamic Digital Twin will allow the A/E/C/FM industry to visualize, monitor, and optimize operational assets and processes to support better inspection and analysis for a more efficient facility operations and maintenance. To support the adoption and implementation of Digital Twin in A/E/C/FM, the authors have defined two clear objectives. First, we discuss requirements for a functionality-based canonical architecture to create a digital twin followed by proposing two tool-based system architecture options for its implementation. Second, we use a case study approach to develop a proof-of-concept Digital Twin of an operating room in a healthcare facility using Power BI Desktop and Azure Services. The prototype aims to monitor room air quality as per INAIL (National Institute for Insurance against Accidents at Work) and ISO (International Organization for Standards) standards. Multiple sensors connected to a Raspberry Pi 4 are used to capture real-time data for various air quality parameters including temperature, humidity, airflow, particulate contamination, and Nitrous Oxide (N_2O) gas. Multiple dashboards are also created to visualize, monitor, and analyze the data harnessed from the OR sensors. The implementation addresses critical issues including security, data storage, visualization, processing, data streaming, collection, and analysis. As an initial validation, the Digital Twin prototype was presented and discussed with a healthcare BIM manager. Initial feedback from the industry expert indicated that the prototype could decrease the required time to respond to facility maintenance issues such as decreased air flow due to possible obstructions.*

KEYWORDS: *Levels of Digital Twins, Facility Management, System Architecture, Power BI, Azure Services, IoT Devices, Raspberry Pi 4.*

REFERENCE: *Ashit Harode, Walid Thabet, Poorvesh Dongre (2023). A tool-based system architecture for a digital twin: a case study in a healthcare facility. Journal of Information Technology in Construction (ITcon), Vol. 28, pg. 107-137, DOI: 10.36680/j.itcon.2023.006*

1. INTRODUCTION

With the increase in size and complexity of building projects, the focus on facility management efficiency and operational safety issues is becoming more important to understand and analyze (Peng et al., 2020). A successful understanding of facility management and the operational need of a building requires collecting and analyzing data related to the function of building systems. The emergence of technologies, such as the IoT (Internet of Things) and data capture and reporting, provides new opportunities for facility managers to automatically log a large quantity of operational data on multiple system parameters (Walker and Strathie, 2016). Research continues to be conducted on the advantage of Big Data in the A/E/C/FM *(*Architecture, Engineering, Construction, and Facility Management) industry. A case study conducted for a Shanghai apartment project showed how analyzing schedule data from prior similar construction projects was beneficial for improving efficiency and productivity (Ma and Wu, 2019). Data collected from smart cities is being used to develop more efficient, sustainable, and productive cities (Kitchin, 2014). Using computer vision and video data collected from construction sites, research has been carried out to detect unsafe construction behavior (Ding et al., 2018).

However, capturing and analyzing accurate data is still a struggle for many, and the research shows that this information gap is responsible for costing the industry $177.5 billion annually (Brown, 2021). As the industry demands better and more precise process improvement to mitigate risk by learning from past data, large amounts of data are collected from CMMS (Computer Maintenance Management Systems) and sensors. This data, often presented in an unstructured form, needs to be analyzed using techniques such as data mining and machine learning (Ahmed et al., 2018).

One way to centralize the storage of this required data is through a BIM (Building Information Modelling) environment. BIM assists facility managers by providing a graphic platform with 3D building elements to organize, analyze, and process data in a common platform (Peng et al., 2020). BIM platforms can also be used to assist with basic activity recording of assets and their maintenance by connecting to CMMS and sensors. Despite the initial utility of BIM in visualizing and documenting building projects, there are limitations to using current BIM tools for facility management, including:

1. BIMs provide a digital representation of objects and buildings with embedded static data. Linking dynamic real-time operational data to the model would provide additional advantages not offered by BIM.
2. Real-time dynamic sensory data from various sources like IoT sensors and edge devices are difficult to stream and visualize into BIM using current mainstream BIM tools (e.g., Revit or Navisworks).
3. Real-time dynamic sensory data is often needed for further analysis to draw insights and make corrective operational decisions. Conventional BIM tools lack the features to perform such analysis.

These limitations have pushed the A/E/C/FM industry to seek advancements by increasing ways for data integration and analysis within BIM, deepening the use of collaborative data models in operational strategies through Digital Twin adoption. The advancement in computing technologies such as the IoT, data transfer speed, development of high-resolution 3D models, data analysis techniques, human-computer interaction, and robotics have made this implementation possible (Akanmu et al., 2021).

A Digital Twin for facility management is envisioned to be a highly complex dynamic virtual representation of a physical asset or space. In a Digital Twin, sensors connected to a physical asset or space are set up to capture data from that asset or space and transfer the information (e.g. humidity, temperature, air quality, etc.) to the corresponding virtual representation (Madubuike et al., 2022). In addition to initial asset data (record data) embedded in the BIM, real-time operational (dynamic) data provides the ability to report on the location, and functional/operational status of the facility. This allows users and operators to virtually better inspect, analyze and adjust its operability from its virtual representation (Madubuike et al., 2022).

By applying Digital Twins for facility management, building project stakeholders can ensure that data related to the building lifecycle can be maintained, verified, validated, and analyzed as the building evolves (Lu et al., 2020, Stanford-Clark et al., 2019). A Digital Twin can also assist in the evaluation of alternative options for building systems through simulation. It can act as a digital mock-up of physical building systems assisting in testing the efficiency of the proposed building system. Simulations using Digital Twins allow users to test the implication of the decisions/changes they make to the asset and run multiple what-if scenarios before actually implementing the decisions. Using the IoT, Digital Twins can assist in monitoring asset performance in near real-time, allowing facility managers to make informed decisions about asset maintenance and management (Menassa, 2021). Visual management and predictive algorithms linked to a Digital Twin allows for scheduling preventive maintenance in the real world (Peng et al., 2020). A Digital Twin can be developed for a specific asset or can be expanded to represent the entire building system digitally.

1.1 Classification Levels of Digital Twins and Current State of the Art

To support the adoption of Digital Twin in the A/E/C/FM industry, it is necessary and critical to develop a deeper understanding of how Digital Twin development and deployment can mature to reach the desired goals. Real-time two-way communication and synchronization between the physical and digital twin, increased capacity to learn, update and communicate between the two counterparts enables a variety of prognostic and diagnostic opportunities. This helps the Digital Twin gradually behave more like the real asset. Agrawal et al. (2022) discussed that the inability to evaluate and identify appropriate technological capabilities of Digital Twin was one of the challenges faced during the implementation among practitioners. Kritzinger et al. (2018) classified Digital Twins into three main categories based on the level of data integration and data flow between the digital and physical counterparts:

1. Digital Model: The digital representation of the existing or proposed physical object that does not utilize any form of automated data exchange between the physical object and its digital counterpart. Any changes to the state of the physical object do not translate to its digital representation, as live data from the physical object is not integrated into the Digital Twin and vice versa.
2. Digital Shadow: Data flow is unidirectional. Data is automatically moved from the physical object to its digital counterpart, but the data flow from the digital representation to the physical object remains manual.
3. Digital Twin: A bi-directional data transfer between the physical object and its digital counterpart is established in a Digital Twin. Any changes to the state of the physical object are reflected in the digital representation and vice versa.

These categories are not exclusive of each other. With an increased level of data integration and automation, a digital model can transform into a Digital Twin. Different A/E/C/FM organizations have provided different interpretations of this transformation through the progression of properties and functionalities across five levels of maturity. Various leading industry organizations and global research communities including ARUP (ARUP, 2019), DPR-VueOps (Arnold and Teicholz, 2021), Autodesk (Tohani and Metcalfe, 2019), buildingSMART (buildingSMART International, n.d.) provided several classifications with varying definitions and descriptions of different maturity levels of Digital Twins. Description of the maturity level of Digital Twin from these three organizations was considered as these are the leading organizations with technological know-how in the A\E\C\FM industry with respect to the Digital Twin implementation. Table 1 provides a summary of required minimum functionalities defined by each organization that a Digital Twin should have across five levels. The published interpretations of levels of Digital Twins by the three organizations were reviewed to develop the summary table. The column titled 'Functionality' lists property/functionality requirements provided in a Digital Twin at a specific level of maturity. If a certain property or functionality for a given level of Digital Twin is defined by one organization, the cell at the intersection of the functionality and Digital Twin level is denoted by an 'X'. A brief description of a few selected functionalities is listed below.

1. Receives Sensors Data: Real-time data from built-in sensors in the physical asset installed to monitor various parameters can provide information regarding the operational conditions of the asset or space being monitored. This information can assist in verifying that the installed asset is working as per plans and specifications.
2. Ability to analyze and predict: Received static and real-time data is analyzed to predict the future performance of the asset and possible future maintenance requests (Lu et al., 2020). This helps in reduction of maintenance costs, assist in inventory management, reduces downtime, and increases the reliability of assets.
3. Simulation: Continuously updated real-time data from a wide array of processes (e.g. supply air system) is simulated to identify critical issues from more vantage points and provide operational improvements (IBM, n.d.).
4. Interconnected incorporation of lower-level twins: Using this functionality, a higher-level Digital Twin can act as a central decision-making network, gathering resources and information from numerous independent lower-level Digital Twins. This functionality can be highly useful in smart city designs (ARUP, 2019).

TABLE 1: Digital Twin levels

S. No.	Functionality	Digital Twin Levels				
		Level-1	**Level-2**	**Level-3**	**Level-4**	**Level-5**
1	Static data supporting asset Maintenance and Operation	X	X	X	X	X
2	Receive and interpret real-time sensor data	X	X	X	X	X
3	Receive information from external sources		X	X	X	X
4	Intelligence to interpret data and provide feedback		X	X	X	X
5	Ability to learn analyze and predict using the data			X	X	X
6	Autonomy in a given domain			X	X	X
7	Simulation				X	X
8	Full autonomy					X
9	Interconnected incorporation of lower-level twins					X

The implementation of level 4 and level 5 Digital Twins seems far-reaching as the A/E/C/FM industry is still trying to navigate through reasoning models, machine consciousness, and full autonomy (ARUP, 2019). However, research on implementing a lower-level Digital Twin is gaining momentum. Real-time monitoring, performance prediction, and strategic improvements have been facilitated by the implementation of Digital Twin in the built environment (Menassa, 2021). ARUP China built a city-scale information model (CIM) platform operating system capable of performing 3D modeling and spatial analysis, visualization of simulation data and statistics, building data dashboards, parametric design modules, and real-time data visualization and analysis. This platform was tested on the Digital Twin of Hong Kong city called Neuron City (ARUP, 2019). Digital twins of existing buildings have also been created and utilized to evaluate Net Zero Energy Buildings (Kaewunruen et al., 2018). To monitor the comfort level of students on campus while meeting the sustainability goals of a smart campus, a Digital Twin of the campus was created to monitor key performance factors (Zaballos et al., 2020). Using the capability of Digital Twins to integrate project information from different stages of the life cycle into the model, the lifecycle assessment of a subway station was evaluated (Kaewunruen et al., 2020). Peng et al. (2020) utilized a hospital building as a case study to show continuous lifecycle integration using Digital Twins. BIM-enabled Digital Twins was also created to simulate a manufacturing work cell assembly line to identify required components in the work cell, process duration, and interdependencies (Ensafi et al., 2021). Smart infrastructure Digital Twin was also created with the objective to turn passive infrastructure into data-centric systems which can collect performance data that assists in future design decisions of the infrastructure to make them more efficient, resilient, and sustainable (Broo et al., 2022). Use of Digital Twin to support smart city facility management has also been explored with the Digital Twin solution receiving heterogeneous data originating from different urban data providers (Bujari et al., 2021). The effectiveness of the structural Digital Twin was also tested to see if by using IoT-based sensors and actuators the Digital Twin could respond in real-time to the performance of its physical counterpart (Chiachío et al., 2022). Similarly, a Digital Twin of temporary structures has been created to enhance monitoring and preventing potential structural failures (Yuan and Anumba, 2020).Use of Digital Twin to support anomaly detection in support of asset monitoring using IFC-based data structure has been tested for centrifugal pumps in the heating, ventilation, and air-cooling system (Lu et al., 2020). Use of Digital Twin to support heritage sites maintenance has also been explored by monitoring relative humidity inside the sites using IoT devices and developing ventilation system by running simulation in the Digital Twin environment (Zhang et al., 2022). Further research has also been conducted in the use of Digital Twin in aiding healthcare facility management (Song and Li, 2022), construction safety (Shariatfar et al., 2022), and audio-based autonomous management of roadway construction (Deria et al., 2022).

The remainder of the paper is organized as follows: Section 2 provides a summary of the literature review and analysis approach used. Section 3 discusses the authors' four-step research implementation methodology and describes two options for a tool-based systems architecture based on a functionality-based canonical architecture. In sections 4 and 5, the authors discuss the proposed functionality-based and tool-based system architectures respectively, and example tools that can be utilized to develop Digital Twins. Section 6 provides an overview of the case study of an OR and the 3D model used to develop the Digital Twin. Section 7 details how the proof-of-concept Digital Twin is created using Power BI as the central platform to integrate the BIM and the telemetry data captured using several sensors connected to a Raspberry Pi 4 computer configured as an edge device. It also explains how software platforms and services like VCAD, and Microsoft Azure are utilized in creating the Digital Twin and proposes a standard naming convention for utilizing the Azure services to better support future implementation and adoption of a Digital Twin. Section 8 discusses various Power BI dashboards created to visualize and monitor sensor air quality data. Validation of the developed dashboard is discussed in section 9. Finally, the paper concludes with sections 10 and 11 providing a discussion and conclusion respectively including proposed future work.

2. LITERATURE REVIEW AND ANALYSIS APPROACH

To further gauge the extent of current research work being conducted targeting Digital Twin implementation, a structured literature search and analysis process was used over 3 phases as shown in Fig. 1. The goal of Phase 1 is to search literature through multiple data sources using keyword searches. A Term Co-Occurrence analysis is conducted with the goal to identify gaps for the implementation of Digital Twins in the A/E/C/FM industry. In Phase 2, identified papers were classified by different research domains that focused on Digital Twin usage and implementation in the A/E/C/FM industry. A gap analysis was conducted in Phase 3 to determine needed future research directions.

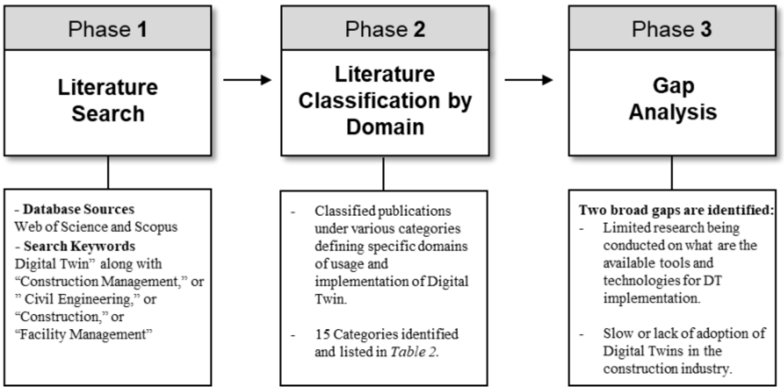

Phase 1
Literature Search
- Database Sources
Web of Science and Scopus
- Search Keywords
Digital Twin" along with "Construction Management," or " Civil Engineering," or "Construction," or "Facility Management"
Phase 2
Literature Classification by Domain
- Classified publications under various categories defining specific domains of usage and implementation of Digital Twin.
- 15 Categories identified and listed in *Table 2*.
Phase 3
Gap Analysis
Two broad gaps are identified:
- Limited research being conducted on what are the available tools and technologies for DT implementation.
- Slow or lack of adoption of Digital Twins in the construction industry.

FIG. 1: Structured literature search and analysis approach used.

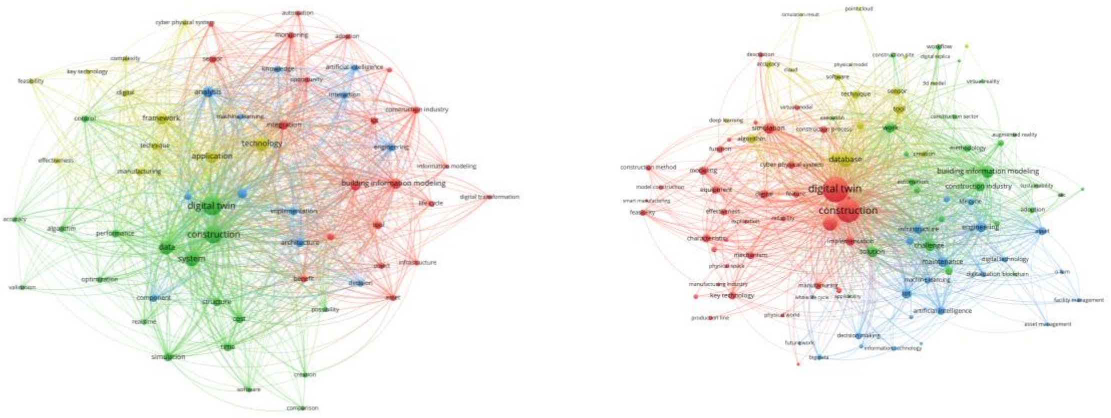
2.1 Literature Search

A Term Co-Occurrence analysis is conducted with the goal to identify gaps in current research for the implementation of Digital Twins in the A/E/C/FM industry. Two database sources are used: *Web of Science*, and *Scopus*. Web of Science database was searched first to identify all the publications that included the words "Digital Twin" along with "Construction Management" or" Civil Engineering" or "Construction" or "Facility Management" in their abstract, title, keywords, or topics. Additional filters are used to only show the results of papers published between 2012 to 2022. Based on this search, 432 publications were identified. Using VOSviewer (Van Eck and Waltman, 2011), a term co-occurrence map is generated to analyze the current focus of research on the use of Digital Twins in the A/E/C/FM industry. Terms that had less than ten occurrences in the database are removed and all the filtered terms are included in the final co-occurrence map.

Fig. 2(a) shows the co-occurrence map generated by the VOSviewer for the *Web of Science database*. The map shows a strong co-occurrence between the words "Digital Twin" and "data" with 147 co-occurrences, and "Digital Twin" and "building information modeling" with 81 co-occurrences, indicating that publications discussing Digital Twins also addressed data and BIM. On the contrary, there are only 46 co-occurrences of the pair "Digital Twin" and "tool", 23 co-occurrences of the pair "Digital Twin" and "creation", and 21 co-occurrences of the pair "Digital Twin" and "software." This co-occurrence analysis indicates that although the research on the utilization of Digital Twins in the A/E/C/FM industry is increasing, little discussion is taking place to identify the required tools and software platforms that can be used to create Digital Twins.

A similar term co-occurrence analysis was conducted using the *Scopus* database. A total of 674 publications were identified. Terms that had less than ten occurrences in the database are removed and all the filtered terms are included in the final co-occurrence map.

Fig. 1(b) shows the co-occurrence map generated by the VOSviewer for the *Scopus* database. The map shows that there is a strong co-occurrence between the words "Digital Twin" and "database" with 163 co-occurrences, and "Digital Twin" and "building information modeling" with 126 co-occurrences indicating that publications discussing Digital Twin also addressed data and BIM. On the contrary, there are only 76 co-occurrences of the pair “Digital Twin” and “tool”, 31 co-occurrences of the pair “Digital Twin” and “creation”, and 43 co-occurrences of the pair “Digital Twin” and “software”. This co-occurrence analysis is proportionally similar to the co-occurrence analysis of the *Web of Science* database and a similar conclusion can be drawn from this analysis as well.



(a)

(b)

FIG. 2: Co-occurrence map of terms related to Digital Twin in the A/E/C/FM industry. (a) Web of Science database (b) Scopus database

2.2 Literature Classification by Domain

To further understand the state of the art of Digital Twin usage and implementation research, publications were classified under various categories defining a specific domain of digital twin. As shown in Table 2, 15 categories were identified for research domains focused on Digital Twin usage and implementation in the A/E/C/FM industry. The second column lists the literature identified that focus on the specific research domain.

TABLE 2: State of the art for Digital Twin research

Research Domain	Literature
Benefits of Digital Twin to the construction industry	(Madubuike et al., 2022), (Akanmu et al., 2021), (Boje et al., 2020), (Khajavi et al., 2019)
Emerging technology and trends to support evolution from BIM to Digital Twin	(Menassa, 2021), (Chiachío et al., 2022), (Jiang et al., 2022), (Li et al., 2021), (Pan and Zhang, 2021), (Seghezzi et al., 2021), (Tariq et al., 2022), (Zhang et al., 2022), (Almatared et al., 2022), (Deria et al., 2022)
Digital Twin knowledge transfer	(Elsehrawy et al., 2021)
Cybersecurity for Digital Twin	(Alshammari et al., 2021)
Cyber-Physical System	(Alshammari et al., 2021), (Akanmu et al., 2021), (Jiang et al., 2022), (Pregnolato et al., 2022), (Pan and Zhang, 2021), (Yuan and Anumba, 2020)
Smart Infrastructure Digital Twin	(Broo et al., 2022), (Pregnolato et al., 2022)
Simulation	(Chiachío et al., 2022), (Yuan and Anumba, 2020), (Zhang et al., 2022)
Real-time information exchange	(Jiang et al., 2022)
Workflow for the development of Digital Twin	(Pregnolato et al., 2022), (Pan and Zhang, 2021), (Broo et al., 2022), (Sobhkhiz and El-Diraby, n.d.)

Research Domain	Literature
Blockchain-based smart contract	(Hunhevicz et al., 2022)
Risk assessment	(Kamari and Ham, 2022)
Facility Management/Operation and Maintenance	(Lu et al., 2020), (Seghezzi et al., 2021), (Lu et al., 2019), (Xie et al., 2020), (Abdelrahman et al., 2022), (Abdelrahman and Miller, 2022), (Song and Li, 2022)
Project Management	(Salem and Dragomir, 2022), (Deria et al., 2022)
Smart Cities	(Bujari et al., 2021), (Kim and Ham, 2022)
Construction safety	(Shariatfar et al., 2022), (Li et al., 2021), (Teizer et al., 2022)

The adoption of Digital Twins in the A/E/C/FM industry remains low as compared to other industries. The slow rate of adoption can be attributed to three factors (Madubuike et al., 2022):

1. Complex and technically demanding steps for setting up a Digital Twin.
2. Required multiple endpoints with sensors to collect massive amounts of data create potential areas of security vulnerability and scalability issues.
3. High costs of implementation and lack of budget.

2.3 Gap Analysis

The literature reveals two broad gaps in the current state of the art. First, there is a lack of discussion on "how" to create a Digital Twin. There seems to be limited research being conducted on what are the available tools and technologies that can be utilized to create an effective Digital Twin to support the needs of owners and facility managers. Consequently, there also seems to be little discussion on how available tools and technologies can be integrated to define different options of system architectures that can be used for the development and implementation of digital twins. The second gap identified relates to the slow or lack of adoption of Digital Twins in the construction industry.

This paper addresses the first gap identified and has two main objectives.. First, the paper discusses requirements for functionality-based cononical architecture to create Digital Twins. Using this architecture as the basis, commercially available tools, and software platforms were reviewed to explore how they can be integrated to create alternative tool-based system-architectures that can assist in implementing Digital Twins while maintaining a scalable, cost-effective, and secure solution. Second, using a case study approach, the paper aims to create a proof-of-concept Digital Twin that adopts one of the tool-based system architectures proposed and applied to a health care facility.

The gaps identified in this research corroborate with the findings of Durão et al. (2018), who in their paper concluded that research in the field of Digital Twin is now shifting more focus towards actual implementation of Digital Twins.

3. RESEARCH IMPLEMENTATION METHODOLOGY

Takyar (n.d.) proposed three stages for implementation of a Digital Twin:

Design: This stage involves determining how physical assets will be integrated with their Digital Twin counterparts. Modeling software such as Autodesk Revit can be used to create a 3D representation of the assets. Users also need to think about what type of information is required during the asset life cycle and how it can be integrated, accessed, and shared using the 3D model. Real-time data capture and analysis solutions usually include several components that span from device communication and management to event processing and data analysis.

Operation: The function and purpose of the Digital Twin need to be defined. When developing a digital twin, it is essential to define the purpose for which the Digital Twin is being developed. The Digital Twin may be used for asset control, operational changes, monitoring, and data analytics for predictive maintenance. Users need to select appropriate software, sensors, and IoT services to accomplish the desired operational objectives defined.

Augmentation: The created Digital Twin should be cohesive enough to support the future development of Digital Twin research and expansion to its initial scope of use. While creating a Digital Twin, users need to make sure that additional future functionality can be added while maintaining performance.

Our research implementation methodology focuses on the design stage proposed and described by (Takyar, n.d.)and includes a four-step process (Fig. 3).

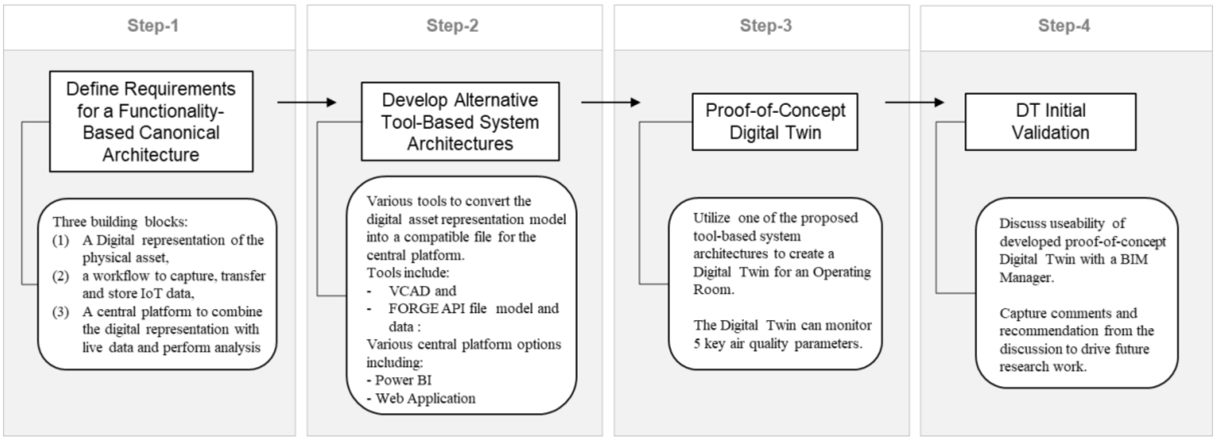

Step-1
Step-2
Step-3
Step-4
Define Requirements for a Functionality-Based Canonical Architecture
Develop Alternative Tool-Based System Architectures
Proof-of-Concept Digital Twin
DT Initial Validation
Three building blocks:
(1) A Digital representation of the physical asset,
(2) a workflow to capture, transfer and store IoT data,
(3) A central platform to combine the digital representation with live data and perform analysis
Various tools to convert the digital asset representation model into a compatible file for the central platform.
Tools include:
- VCAD and
- FORGE API file model and data :
Various central platform options including:
- Power BI
- Web Application
Utilize one of the proposed tool-based system architectures to create a Digital Twin for an Operating Room.
The Digital Twin can monitor 5 key air quality parameters.
Discuss useability of developed proof-of-concept Digital Twin with a BIM Manager.
Capture comments and recommendation from the discussion to drive future research work.

FIG. 3: Four-step methodology to define and use a Digital Twin.

Step-1: Discuss requirements for a functionality-based canonical architecture using a unified modeling language representation. The proposed architecture is made up of three building blocks: (1) a Digital representation of the physical asset, (2) an IOT solution to capture, transfer and store dynamic sensory data from edge computing devices, and (3) a central platform to combine the digital representation with live data and perform analysis. These building block are similar to the Digital Twin solution proposed by Zhang et al. (2022) when utilizing Digital Twin to automatically optimize relative humidity in underground heritage sites. They divided their Digital Twin in three building blocks: (1) the physical world, (2) the digital environment including digital representation of asset and data collected through processing and analysis sensor data, and (3) a Digital Twin platform to combine IoT technology and to link the digital and physical environments. The functionality-based architecture addresses how to integrate real-time data from sensors and edge devices with the digital asset representation and other data into a central platform for visualization and analysis.

Step-2: Define several tool-based system architectures based on the functionality-based architecture. The suggested tool-based system architecture options are based on some of the available hardware tools and software platforms identified that can be integrated to create a Digital Twin. Options vary based on their initial costs, license options, and level of coding and programming requirements and are influenced by our review of the technical literature (Cityenith, n.d., SIEMENS, n.d., Autodesk, n.d., Autodesk, 2020, Autodesk University, n.d., Bray, 2021, Broz, 2019, Microsoft, n.d.-a, Microsoft, n.d.-b, Blogic, n.d., Takyar, n.d., Coriani, 2020). The tool-based system architecture utilizes Revit to create a digital representation of the asset, VCAD or Forge API to transfer the digital representation to a central platform, and Power BI (Desktop/Service/Mobile) or user created web application as the central platform. The data from the physical asset is captured in the tool-based system architecture alternatives using sensors either custom built using Raspberry Pi or commercially available. The data from the sensors is transferred to the central platform using proprietary data loggers and software or Microsoft Azure services.

Step 3: A tool-based system architecture option is chosen and the details of its components and technological needs are defined to create the proof-of-concept Digital Twin. The implementation utilizes Revit to create a digital representation of the asset, VCAD to transfer the digital representation to the central platform, and Power BI desktop as the central platform. Real-time data from sensors connected to a Raspberry Pi 4 (edge device) transfer data via Microsoft Azure services. The goal of developing the prototype is to measure and test air quality in an operating room by monitoring five key parameters: temperature, humidity, particulate contamination, airflow, and N_2O concentration. Real-time values for these parameters are visualized using a set of designed dashboards.

Step-4: Perform an initial validation of the Digital Twin implementation through discussions with an experienced healthcare facility BIM manager. Feedback is documented for future research and implementation.

4. FUNCTIONALITY-BASED ARCHITECTURE FOR ASSET DIGITAL TWINS

In an asset Digital Twin, static data (or record data) embedded into the 3D BIM is integrated with real-time IoT data captured from building sensors and transmitted using edge devices into a common platform. This allows to visualize, and analyze the information simultaneously to generate actionable information that can inform the physical counterpart.

Fig. 4 illustrates a functionality-based canonical architecture that identifies and describes the required components to integrate graphical and non-graphical information from the BIM with real-time telemetry data generated through an IoT solution into a common central platform user interface (UI). The UI can be configured to view and analyze graphics and data through user-defined dashboards or can be linked to external tools such as machine learning algorithms for further analysis. The architecture of Fig. 4 is defined using the unified modeling language communication diagram. A communication diagram is chosen to represent the proposed system architecture because they show the interaction of the objects that send and receive messages (Booch, 2005).

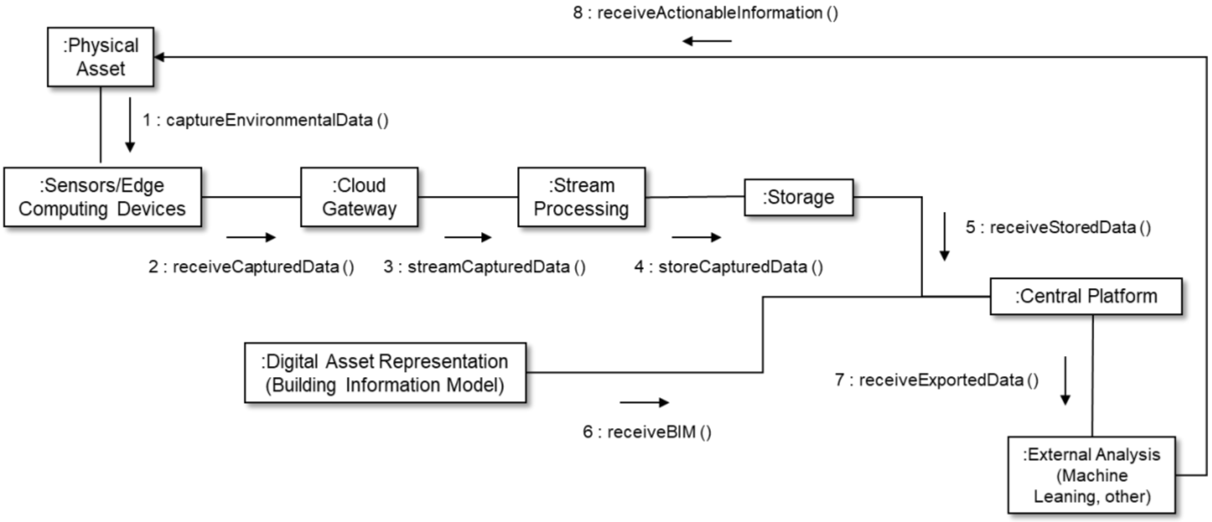

8 : receiveActionableInformation ()
:Physical Asset
1 : captureEnvironmentalData ()
:Sensors/Edge Computing Devices
:Cloud Gateway
:Stream Processing
:Storage
5 : receiveStoredData ()
2 : receiveCapturedData ()
3 : streamCapturedData ()
4 : storeCapturedData ()
:Central Platform
:Digital Asset Representation (Building Information Model)
7 : receiveExportedData ()
6 : receiveBIM ()
:External Analysis (Machine Leaning, other)

FIG. 4: Proposed functionality-based architecture.

With reference to Fig. 4, a digital representation of the physical asset is usually achieved using model authoring tools to create a BIM (Building Information Model) of the asset or building, with its record (static) data integrated in the model. Such historical data is usually collected during the design and construction phases of a facility and its assets. The resulting digital asset representations is imported in the central platform (1) for visualization and analysis.

IoT solutions usually include several components that spans from devices for communication and management to event processing and data ingestion and analysis (Madakam et al., 2015). Data captured from sensors attached to the physical asset are transmitted using edge devices in the form of triggered events representing environmental data. These events can take the form of user-defined time-lapse, movement of occupants, or start and stop of an asset. Captured data is transmitted to a cloud gateway (2) to receive the data and to form a bridge between hardware/software local applications and cloud-based storage. Cloud gateways manage bi-directional communications with devices, queuing events and messages. Once received, the cloud gateway then instructs a stream processing application to stream the captured data (3) and store the data in a storage application (4). Azure Stream Analytics (Microsoft, n.d.-a) is an example of a real-time analytics service designed to build end-to-end IoT serverless streaming pipeline solution.

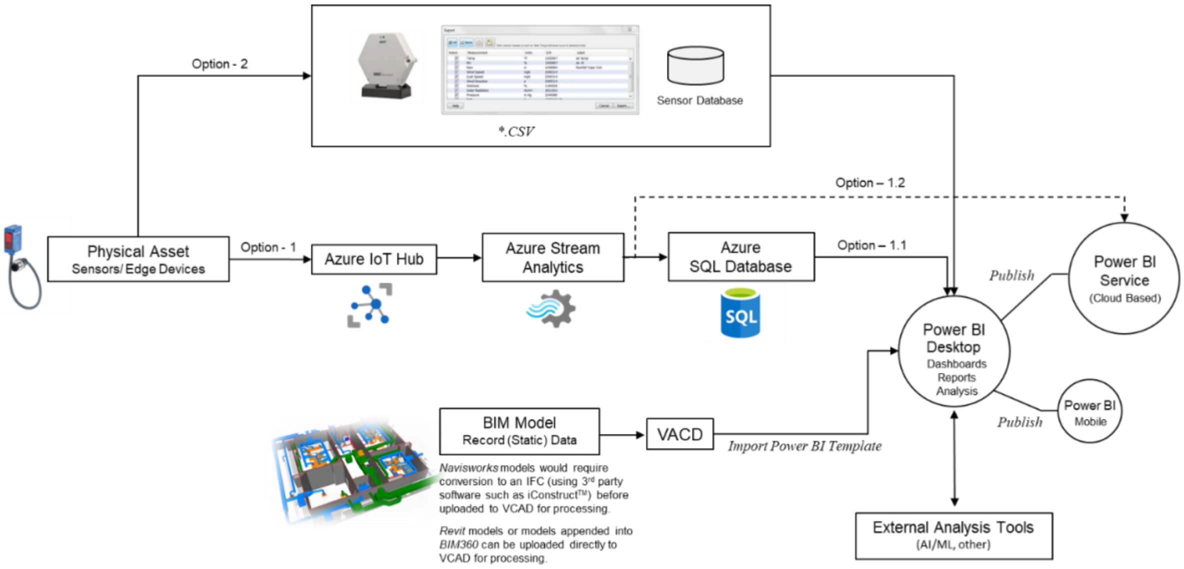
From the storage location, the captured data is imported to a central platform (5) that can integrate with the digital asset representation (BIM model). Graphical and non-graphical data are linked and visualized through a user interface comprised of several dashboards designed to allow the user to view critical information in real-time. Through the central platform, data can be linked (7) to external analysis tools to perform various analyses and calculations and generate meaningful and actionable information that can be relayed back to the physical twin (8). This allows the Digital Twin to provide its physical counterpart with a variety of prognostics and diagnostic opportunities for optimization.

5. TOOL-BASED SYSTEM ARCHITECTURE FOR ASSET DIGITAL TWINS

The functionality-based architecture described in Fig.4 can take various configurations based on the technology tools used. In this section, we discuss two tool-based system architectures and explore available technology (hardware and software) that can be leveraged to create a Digital Twin. The system architecture options proposed and defined vary based on their initial costs, license options, level of coding and programming experience needed, and influenced by the technical literature we identified and reviewed (Cityenith, n.d., SIEMENS, n.d., Autodesk, n.d., Autodesk, 2020, Autodesk University, n.d., Bray, 2021, Broz, 2019, Microsoft, n.d.-a, Microsoft, n.d.-b, Blogic, n.d., Takyar, n.d., Coriani, 2020).

5.1 Alternative-1 System Architecture:

Fig. 5 shows the first alternative tool-based system architecture that utilizes either Microsoft Power BI Desktop (Option-1.1 and Option-2) or its cloud-based Power BI Service (Option-1.2)as the Digital Twin Central Platform to combine and visualize data. Data connectivity is achieved using Microsoft Azure services including IoT Hub Cloud Gateway, Azure Stream Analytics for data stream processing, and Azure SQL (Structured Query Language) for data storage (Coriani, 2020, Microsoft, n.d.-a, Microsoft, n.d.-b).



Navisworks models would require conversion to an IFC (using 3rd party software such as iConstruct™) before uploaded to VCAD for processing.

Revit models or models appended into *BIM360* can be uploaded directly to VCAD for processing.

FIG. 5: Tool-based system architecture (Alternative-1: Power BI and Microsoft Azure Services).

Azure IoT Hub is a cloud-based service that allows connecting edge devices to a cloud-hosted solution back end without writing code. Once edge devices and sensors are connected to the Azure IoT Hub, users can use Azure Stream Analytics to build an end-to-end serverless streaming pipeline. Azure Stream Analytics allows for real-time analytics and complex event-processing of a high volume of fast streaming data from multiple sources simultaneously. Data transmitted from sensors connected edge devices can also be stored as a SQL database using Azure SQL Database. This is a cloud-based database service provided by Microsoft. Using this functionality, data from sensors connected to Azure Hub can be either streamed in near real-time into services like Power BI Service or streamed and stored as SQL Database tables which can then be imported into the Power BI desktop.

Power BI is a business intelligence collection of software services, apps, and connectors that can be utilized to turn data from different sources into coherent, visually immersive, and interactive insights (Microsoft, n.d.-b). Using Power BI, users can easily connect to data irrespective of format or location (stored locally or on the cloud), visualize and analyze the data and create different dashboards. Power BI is available in three basic versions:

1. Power BI Desktop is a free Windows desktop application that allows users to connect to data from multiple sources. Users can create data models to visualize and analyze the data that can be later shared as reports with others (Microsoft, n.d.-b).
2. Power BI Service is a cloud-based SaaS (Software as a Service), also referred to as Power BI Online. It provides the same functionality as the desktop version but with the ability to link to limited data sources (Microsoft, n.d.-b). It also allows users to connect to data streaming services to stream live data as a line

chart or information tile. Users can create dashboards that display tiles that can be selected to open reports to explore data further (Microsoft, n.d.-b).

3. Power BI Mobile provides the ability to display Power BI reports and dashboards on Windows, iOS, and Android mobile devices. Reports and dashboards created in Power BI Desktop and Service can be published, shared, and accessed as mobile reports.

VCAD, a 3rd party tool, is utilized to convert BIM model files to a Power BI file to be utilized in the integration process. VCAD internally utilizes the Forge services to generate the graphical interface within Power BI to visualize and manipulate the graphical representation of the BIM model. It also incorporates various Forge-based navigation tools provided to the user to manipulate the model within Power BI. In this alternative, the users have no interaction with the Forge services.

VCAD does not necessarily have to use Forge services to transfer Digital Asset Representation (BIM model) to a Power BI file. VCAD allows the option to choose the Open Viewer to get similar results. The use of either Open Viewer or Forge can be selected by the user prior to processing the conversion of the model file. Using the Open Viewer has functionality limitations over Forge For the case study discussed, the authors have chosen to use the Forge Services when using VCAD to convert the BIM to a Power BI file.

Other options exist, including iConstruct’s Genus platform (iConstruct, n.d.) and Pro-Revit Toolkit, a plugin to Autodesk Revit by Pro Tools (MG, n.d.). The Power BI template generated through VCAD can also be used to create reports and dashboards that can be shared and distributed.

Real-time data from sensors installed in the physical twin can be exported out of the sensors and imported into Power BI using two (2) options:

1. As a local CSV (comma separated values) file (Fig. 5/Option 2) that is imported as a static query: Using proprietary data loggers, data receivers, and software available for plug-and-play sensors, users can export csv files for the data generated by sensors either for a particular time frame or complete data. An example of this can be sensors, data loggers, data receivers, and software manufactured by ONSET to create an IoT network (ONSET, n.d.). Using this system architecture, users can capture and store sensor data into a local system/computer and export the data as .csv to be used by analytics software like Power BI. This alternative bypass the need for Cloud Gateway, and Stream Processing nodes shown in Fig. 4. Applications are limited to localized data collected for a limited-size location (e.g., a specific room or area).
2. Through connecting the sensors to Azure IoT Hub service (Cloud Gateway node) (Fig. 5/Option 1) and saving it to a SQL database format using Azure SQL (Storage node) via Azure Stream Analytics (Stream Processing node). This SQL database can be imported into Power BI and updated at regular intervals automatically. Microsoft Azure servicesprovides users with services and capabilities to connect with edge devices and transmit data to the cloud. Users can connect, maintain, and extract data through multiple devices.

Any sensor data imported into Power BI can be mapped and linked to graphical model elements, facilitating data viewing along with other data embedded and imported with the BIM.

Using Power BI Service as the Central Platform, data from sensors installed in the physical twin can be directly streamed into Power BI Service using Azure IoT Hub through Azure Stream Analytics without the need for storing the data in Azure SQL. This data will be updated automatically. Power BI Service provides users with three types of real-time datasets (Microsoft, n.d.-b):

1. Push dataset: Power BI service creates a new dataset to store the data being pushed into the Power BI services. Since a dataset is being created and continues to store data, it can be used to create reports and perform analysis.
2. Streaming dataset: In a streaming dataset, data is being pushed into the Power BI services but with a major difference from push data. Streaming data is only stored temporarily and can be used only to display visuals like a line chart. The data is stored for one hour.
3. PubNub streaming dataset: Power BI services can use PubNub SDK (Software Development Kit) to read existing PubNub data streams. No data is stored in Power BI services.

Our digital twin prototype described in detail in section 7 has utilized alternative 1 for the implementation.

5.2 Alternative 2 Systems Architecture:

Fig. 6 describes a second alternative tool-based system architecture proposed for implementing a digital twin. This alternative utilizes a dedicated web application used to develop the Central Platform. To transfer the Digital Asset Representation (BIM model) to the web application, Forge API is used to write code to first convert the Digital Asset Representation into a format suitable for web application. Programming is further needed to develop the interface for the Digital Twin, link sensor data and create various dashboards to display the data. This solution provides more flexibility for creating customized interfaces for graphics and data but requires a large amount of high-level programming. (Autodesk University, n.d.). The API provides capabilities to upload a BIM in various file formats (over 60 types) and to combine model graphics and data with IoT data. Using Forge API, end users can tailor their applications to their required functionality and desired features. To achieve data connectivity, this alternative can also utilize Microsoft Azure services. Various other options for data connectivity to transfer telemetry data may include AWS (Amazon Web Services) or the Google Cloud platform.

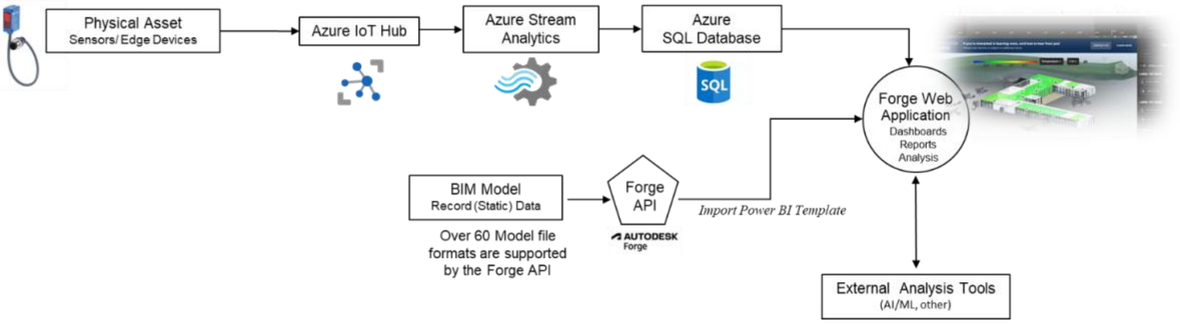


FIG. 6: Tool-based system architecture (Alternative-2: Forge API and Microsoft Azure Services). Adapted from Coriani (2020)

External analysis tools can be used to perform user-specific analysis on the imported data such as trigger emergency notifications based on sensor readings exceeding set points.,. As an alternative, the web application can also be coded to analyze telemetry data, develop insights, and generate actionable information.

5.3 Other Options:

Alternate software applications that can be used as central platforms come with various levels of functionalities and include Ansys, Twinzo, Invicara, Tandem, and EcoDomus. For example, the recently released Autodesk Tandem cloud platform (Autodesk, 2020) is a cloud-based Digital Twin platform that enables users to create Digital Twins for building handovers to owners and operators. Since Tandem is the latest release from Autodesk, the platform currently can only create Digital Twins from Revit and IFC (Industry Foundation Class) files. The Digital Twin generated using Tandem is currently descriptive and focused on generating Digital Twins with data embedded in the BIM. Autodesk plans on introducing capabilities to link real-time operational data from sensors to the Tandem platform in future versions. Predictive and autonomous features are also planned as a long-term goal.

Recent versions of EcoDomus allow users to create virtual replicas of facilities and assets with integration to BIM/IoT/SCADA (Supervisory Control and Data Acquisition)/ CMMS/ GIS (Geographic Information System) systems (SIEMENS, n.d.). Cityzenith (Cityenith, n.d.) is another platform that allows users to create a Digital Twin of the urban built environment. Cityzenith analyzes data from buildings, transportation, and people to optimize the functioning of an urban environment to reduce emissions, ensure public safety, and improve productivity. Digital Twins developed by Citizenith can also be used to develop large infrastructural projects in the urban environment.

5.4 Summary of Proposed Tool-Based System Architectures

Table 3 summarizes our analysis of the different platforms and tool-based system architecture discussed above to create Digital Twins. The columns are divided into three categories, “Alternative-1 System Architecture”, “Alternative-2 System Architecture”, and “Other Option – E.g., Autodesk Tandem” discussed above.

The rows of Table 3 contain factors that define characteristics that are specific to each tool-based system architecture. These factors were synthesized based on the authors’ review of the literature (Microsoft, n.d.-b, Microsoft, n.d.-a) and on differences identified between the tool-based system architecture alternatives discussed above and their requirements to develop a Digital Twin.

1. Central Platform Requirements: Different tool-based system architecture requires users to have varying levels of familiarity with coding languages to create the central platform and transfer static and live data to the platform. Languages used vary from SQL, Java, or C#. Depending on the user's familiarity with these languages, they would like to consider one tool-based system architecture over the other to implement a Digital Twin.
2. Requires Subscription to Cloud account: Subscriptions to cloud accounts for services like Power BI Service, VCAD, BIM 360, Tandem, Autodesk Forge, and Azure can add additional costs and should be considered in the implementation budget.
3. Data analysis: Apart from Tandem, data can be analyzed in all the system architectures discussed. The difference lies in the level and depth of data analysis. Power BI has more ready-to-use tools and features available to analyze data. Using web applications may provide more flexibility but requires extensive programming.
4. Data visualization: Based on how the Digital Twin is hosted, the tools available for data visualization can change. For example, Power BI (Desktop and Service) provides users with multiple tools to visualize and filter data, while if a user decides to create a web application, these tools will have to be developed by the users and would be limited by either the users coding knowledge or by the API used.
5. 3D model file format: The 3D model file format supported by the system architecture is critical in deciding which alternative to choose to create a Digital Twin. For example, until recently, VCAD did not support Navisworks file formats and conversion of Navisworks files to IFC was required before processing the file using VCAD.
6. Notification Automation: Using Azure IoT Central, users can automate the emergency notification system to a responsible person. If the sensor reading crosses a predefined threshold or setpoint, an email notification is automatically sent to the responsible person.
7. Access and Sharing of Digital Twins: It is critical to share developed Digital Twins among different stakeholders of the project. The mode of sharing needs to be considered by the users depending on the existing technology infrastructure of the project.
8. IoT/Edge Device data format: The user may also want to consider the format in which the devices collect the data. IoT/Edge devices can collect data in CSV, Excel, and SQL format, to name a few. Based on the data format, the user might prefer one tool-based system architecture over the other.
9. Level of Security for building owners: Some construction projects have additional security requirements related to data transmission. It is important to understand how the data will be transmitted and stored in different system architectures in these projects.
10. Speed of data transfer: IoT/Edge devices record data in real-time. However, transferring the data to a Digital Twin may create a time lag between when the data was produced and when the data was represented in the Digital Twin. In some scenarios like measuring the temperature of office space, this lag is insignificant, while in scenarios of measuring the temperature of IT rooms, this lag is critical and needs to be minimal.
11. Ability to store/save data: The data generated through IoT/Edge devices not only help in understanding the current performance of a facility but can also be stored and analyzed to predict future behaviors. Depending on the tool-based system architecture adopted, users will have the option to store the data for future analysis.

TABLE 3: Summary of the different platforms discussed to create Digital Twins using alternative system architectures.

No.	**Factors**	**Digital Twin for A/E/C/FM – Tool-Based System Architecture**				
		Alternative-1 System Architecture			**Alternative-2 System Architecture**	**Other Options - E.g., Autodesk Tandem**
		Option-1.1	**Option-1.2**	**Option-2**		
1.	Central Platform Requirements	Low level of programing required as Power BI Desktop is used as central platform and sensor data is transferred using Azure Services that require users to understand SQL.	Low level of programing required as Power BI Service is used as central platform and sensor data is transferred using Azure Services that require users to understand SQL.	No programming requirement as Power BI Service is used as central platform and sensor data is transferred using proprietary data loggers.	High programming requirement and users need to code the entire central platform along with its data analytics capability and how data is transferred.	Not Applicable
2.	Requires Subscription to Cloud Account	VCAD and Azure Subscription	VCAD, Power BI Service, and Azure Subscription	VCAD subscription	Azure and Autodesk Forge	Tandem free up to 1000 tagged assets
3.	Data Analysis	Data Analysis using Power BI tools	Data Analysis using Power BI Service tools (Check the difference between power bi and power bi service)	Data Analysis using Power BI tools	Data can be analyzed through Azure IoT central and through the web development phase	Currently does not have data analysis capabilities
4.	Data Visualization	Data visualization is possible using Power BI tools (Microsoft, n.d.-b)	Data visualization is possible using Power BI Service tools. Limited visualization is available for streamed data.	Data visualization is possible using Power BI tools (Microsoft, n.d.-b)	High data Visualization in the model using Forge and through Web Development	Construction data can be added and modified. Models can be navigated through filters for embedded properties
5.	3D Model File Format	ifc, .rvt, .nwd, .nwc, .obj, .stl and .dwg when using VCAD service from their website. All formats supported by BIM 360 when using VCAD as a BIM 360 plugin	ifc, .rvt, .nwd, .nwc, .obj, .stl and .dwg when using VCAD service from their website. All formats supported by BIM 360 when using VCAD as a BIM 360 plugin	ifc, .rvt, .nwd, .nwc, .obj, .stl and .dwg when using VCAD service from their website. All formats supported by BIM 360 when using VCAD as a BIM 360 plugin	Forge viewer supports 60 different file formats	Currently, only Revit files are acceptable
6.	Notification Automation	Not Available	Not Available	Not Available	Possible through Azure IoT central (Microsoft, n.d.-a)	Currently Not Available
7.	Access and sharing of Digital Twin	Cloud and locally	Cloud	Shared through Power BI service to anyone with a Power BI license (Microsoft, n.d.-b). Power BI files can also be shared within the organization	Cloud	Cloud

No.	Factors	Digital Twin for A/E/C/FM – Tool-Based System Architecture				
		Alternative-1 System Architecture			Alternative-2 System Architecture	Other Options - E.g., Autodesk Tandem
		Option-1.1	Option-1.2	Option-2		
8.	IoT/Edge Device Data Format	Sources Supported by Power BI (Microsoft, n.d.-b)	Sources Supported by Power BI Services. For streaming needs, data through stream analytics	Sources Supported by Power BI (Microsoft, n.d.-b)	All Format	Currently Not Available
10.	Level of Security for the building owner	Data and 3D building models need to be shared with third-party service providers. Microsoft Azure provides advanced security features at varying price points (Microsoft, n.d.-a).	Data and 3D building models need to be shared with third-party service providers. Microsoft Azure provides advanced security features at varying price points (Microsoft, n.d.-a).	Data can be stored locally and imported into Power BI, but 3D model data needs to be shared with VCAD (Third Party)	3D models can be hosted locally. IoT data is streamed through Azure (Third Party). Microsoft Azure provides advanced security features at varying price points (Microsoft, n.d.-a).	The model needs to be uploaded to Autodesk Tandem
11.	Speed of IoT data transfer	Latency in updating Digital Twin data	Data can be streamed in real-time or can be stored (Push Power BI Service) (Microsoft, n.d.-b)	Latency in updating Digital Twin data	Data can be streamed in real-time	Currently Not Available
12.	Ability to store/save IoT data	Data can be stored in Power BI	Push data is stored in Power BI Service (Microsoft, n.d.-b)	Data can be stored in Power BI	Data can be stored	Currently Not Available

6. CASE STUDY OVERVIEW

A case-study approach is used to implement the Digital Twin. The case study involves monitoring air quality in an OR suite. The OR selected is one of three OR suites that are in a 325,000 sqft 4-story hospital in Colorado, with 50 beds expandable to 150 beds. The facility provides various services including intensive care, emergency care, surgery, radiology, advanced cardiology, a birth center, and an in-house pharmacy. Fig. 7 shows the mechanical system for the OR Suite.

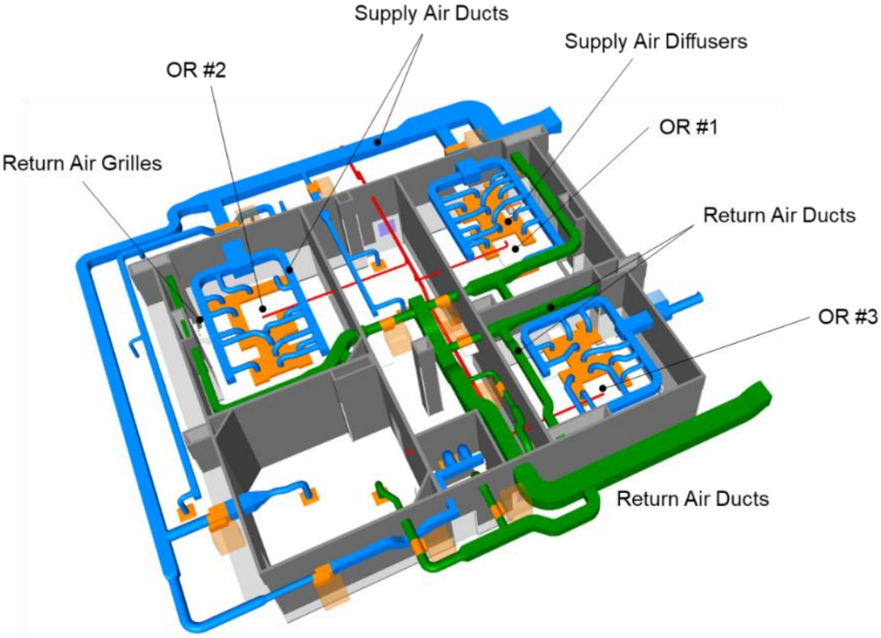


FIG. 7: Navisworks 3D model of the mechanical system for an operating room suite.

The supply air (SA) ducts, shown in blue, are connected to diffusers located directly above the operating table. Return air (RA) ducts, shown in green, are connected to grilles located in the corner and closer to the floor. The location of the return air (RA) grilles is selected based on code requirements for air circulation design patterns. The location of the grilles forces air to move away from the center of the room, where the operating table is to pull away contaminated air and eliminate chances of patient infection.

The air quality of an operating room needs to be monitored and maintained. This is critical to benefit both the patient and the operating room staff. There are several primary objectives in achieving acceptable air quality in an operating room: (1) to maintain a sterile air environment and infection control for the benefit of the patient and staff, (2) to maintain control of anesthetic gases for the benefit of the operating room staff, which includes measuring the exposure levels, detecting any leaks from the gas supply system, and ensuring the effective operation of the waste gas scavenging system, (3) to control oxygen levels as a fire protection measure (Sheriff, 2020) through detection of leaks from weak points in the anesthetic equipment supplying oxygen, and (4) to ensure a comfortable work environment for the operating staff by ensuring adequate room temperature and humidity levels.

The paper focused on measuring five air quality parameters that address some of the air quality objectives listed above. This includes temperature, humidity, Nitrous oxide (N_2O - commonly known as laughing gas) concentrations, particulate contamination, and airflow in supply and returns ducts. N_2O is used as an anesthetic agent in medical, dental, and veterinary operatories (National Institute for Occupational Safety Health, 1994). N_2O levels in an OR need to be maintained below recommended exposure limit to prevent a decrease in hearing/vision ability, manual dexterity, and mental performance of operating room staff. The concentration of airborne particulate in an OR is maintained using air filtration systems. To decrease the chances of patients contracting infections, the concentration of airborne particulate must be monitored and controlled.

In the United States, standard temperatures for operating rooms are kept between 70-75°F with humidity between 50- 60% (Ellis, 1963). Additionally, the INAIL (National Institute for Insurance against Accidents at Work) and ISO (International Organization for Standards)14644-1 (Romano et al., 2020) put the threshold limit for airborne particles in an ISO Class 5 operating room at 3520 particle/meter cube for particles with a diameter greater than 0.5 micrometers (Romano et al., 2020). To reduce the rate of infection in an OR, 20 Air Changes per hour are recommended (Khankari, 2018). Additionally, N_2O concentration should remain below 25 ppm as a time-weighted average during the period of anesthetic administration to prevent the decrease in mental performance, audiovisual ability, and manual dexterity of the operators (National Institute for Occupational Safety Health, 1994).

7. DIGITAL TWIN TO MONITOR AIR QUALITY FOR AN OPERATING ROOM

Fig. 8 describes the system architecture chosen for implementing the proof-of-concept Digital Twin. The system components and arrangements follow alternative #1 option #1 tool-based system architecture described in Fig. 5. The Digital Twin will receive information from two sources: a 3D Navisworks model of the OR suite, and sensors connected to a Raspberry Pi 4 computer to collect real-time data to monitor five air quality parameters in the operating room including temperature, humidity, particulate contamination, airflow, and N_2O concentration. The system architecture selection was based on the authors' familiarity and experience with the Microsoft Power BI platform for developing Digital Twins, robustness, and functionality of platforms and tools used (e.g. Azure Services); compatibility of data transfer across platforms; strong working relationship and support from software developers, including VCAD (Blogic, n.d.) and iConstruct Pro (iConstruct, n.d.); and the ability to secure operating licenses.

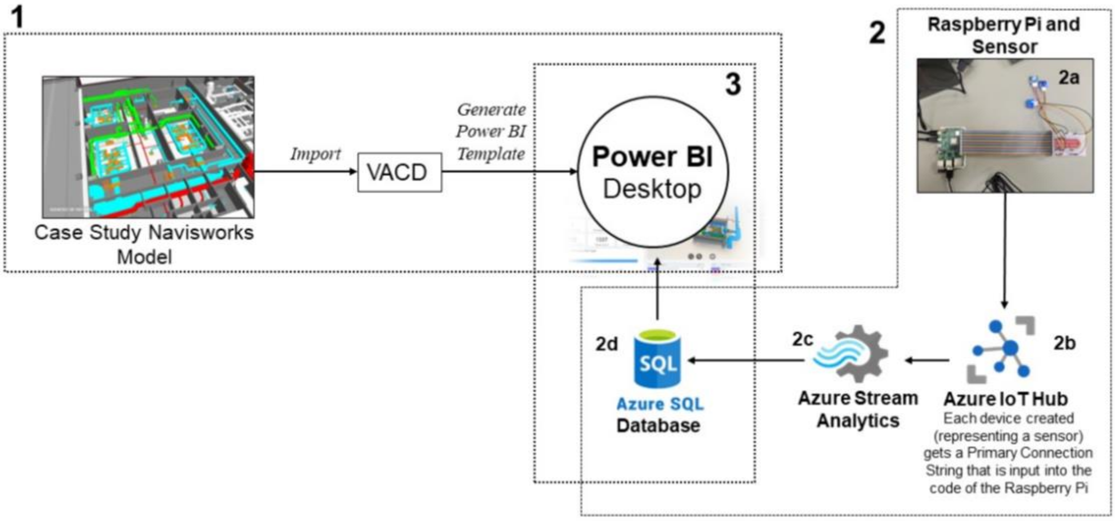

1
Case Study Navisworks Model
Import
VACD
Generate Power BI Template
3
Power BI Desktop
2
Raspberry Pi and Sensor
2a
2b
Azure IoT Hub
Each device created (representing a sensor) gets a Primary Connection String that is input into the code of the Raspberry Pi
2c
Azure Stream Analytics
2d
SQL
Azure SQL Database

FIG. 8: System architecture for proof-of-concept Digital Twin.

As shown in Fig. 9, three sensors are proposed to measure four of the five parameters: temperature, humidity, particulate contamination, and N_2O concentrations. The authors have not installed these sensors in an OR. Rather, the sensors are connected to the Raspberry Pi 4 computer located in the authors' research lab and are used to capture air quality parameter data from the lab space simulating actual sensors that could be installed in the OR. The configuration of sensors connected to the Raspberry Pi 4 creates an edge device capable of connecting to the internet and cloud services (Darling, 2021). Fig. 9 also shows the proposed location of the sensors if they are to be installed in the OR. The temperature, humidity, and N_2O sensors will be located on the wall or at other designated locations. Sensors to measure particulate contamination are located within the return ducts. Temperature and humidity readings will be measured every 15 minutes while readings for N_2O, airflow, and particulate contamination will be measured every 15 minutes when the OR is operational. The authors used sensor DHT11 to measure temperature and humidity, sensor MQ-135 to measure N_2O, and sensor PMSA003I to measure particulate contamination. Due to the unavailability of a sensor to measure airflow, mockup data was created using the literature reviewed. Table 4 shows sample data for each of the five air quality parameters used in the implementation.

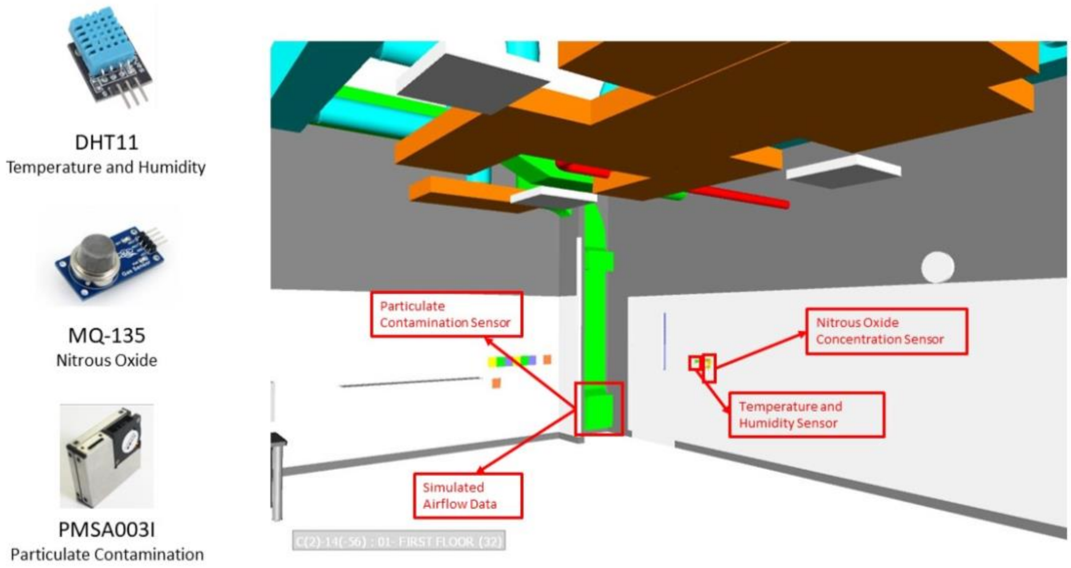

DHT11
Temperature and Humidity
MQ-135
Nitrous Oxide
PMSA003I
Particulate Contamination
Particulate Contamination Sensor
Nitrous Oxide Concentration Sensor
Temperature and Humidity Sensor
Simulated Airflow Data
C(2)-14(-56) : 01 - FIRST FLOOR (32)

FIG. 9: Sensor locations in OR

TABLE 4: Sample of parameters data for temperature, humidity, N2O, particulate contamination, and airflow.

Date	Time	Temperature (Celsius)	Humidity (%)	Nitrous Oxide (ppm)	Particulate Contamination (pp/m^3)	Airflow (CFM)
		Sensor DHT 11		Sensor MQ-135	Sensor PMSA003I	Sample Data
3/4/2022	12:39:13 PM	22	53	8	3245	1215
3/4/2022	1:09:13 PM	22	53	9	3357	1245
3/4/2022	1:24:13 PM	22	53	25	3004	1245
3/4/2022	1:39:14 PM	22	53	5	3007	1268
3/4/2022	1:54:14 PM	22	53	11	3500	1240
3/4/2022	2:09:14 PM	22	53	15	3112	1245
3/4/2022	2:24:14 PM	22	53	17	3160	1262
3/4/2022	2:39:15 PM	22	53	16	3164	1280
4/26/2022	6:14:20 AM	22	53	11	3321	1250
4/26/2022	6:44:20 AM	22	54	13	3236	1270
4/26/2022	6:59:20 AM	22	55	27	3148	1273
4/26/2022	7:14:21 AM	22	55	7	3421	1288
4/26/2022	7:29:21 AM	22	56	20	3301	1267
4/26/2022	7:44:21 AM	22	56	20	3350	1253
4/26/2022	7:59:21 AM	22	56	11	3440	1244
4/26/2022	8:14:22 AM	22	56	15	3058	1211
4/26/2022	8:29:22 AM	22	56	24	3114	1234
4/26/2022	8:44:22 AM	22	56	2	3115	1200
4/26/2022	8:59:23 AM	22	56	26	3236	1199

Details to create the proof-of-concept Digital Twin are as follows:

1. Import the Revit model into Power BI Desktop using VCAD

The architectural, mechanical, and interior Revit models of the healthcare facility were appended to Navisworks and the model was sectioned to isolate the OR suite in the model view. Using iConstruct Pro (iConstruct, n.d.), the sectioned model was exported as a .nwd file and converted into a Power BI template file using the VCAD platform provided by Blogic.srl (Blogic, n.d.) that was imported into the Microsoft Power BI desktop platform. A VCAD_Asset table is automatically generated with an 'objectId' column containing unique values corresponding to the graphical elements in the imported model. The unique ids corresponding to graphical elements representing the sensors in the model will be used to map the elements to their sensor data.

2. Transmit and save sensor data to Azure SQL Database

Temperature, humidity, N_2O levels, and particulate contamination data captured by sensors connected to the Raspberry Pi 4 computer are exported to Azure IoT Hub, streamed using Azure Stream Analytics, and then saved in the Azure SQL database. The Raspberry Pi acts as an edge computing device and is configured to collect, store, transform and transmit data. The Raspberry Pi 4 is comparable to an entry-level x86 PC system and is capable of connecting to the internet via ethernet cable or Wi-Fi without requiring additional hardware (Raspberry Pi, n.d.). Raspberry Pi's official operating system (OS) Debian is used for the implementation. The Raspberry Pi 4 is chosen for its availability and the abundance of online support documentation.

The three sensors used for temperature/humidity (Sensor: DHT11), N_2O concentration (Sensor: MQ-135), and particulate contamination (Sensor: PMSA003I) have different working principles to collect data, which is often not generated in a human-readable format. The sensors need to be connected to a computer/processor that can interpret the data collected by these sensors and convert it into a human-readable format. For example, the MQ-135 gas sensor measures the concentration of gases based on the changes to the conductivity of its sensitive material. This sensor uses SnO_2 as the sensitive material that has low conductivity in clean air and its conductivity rises as the concentration of the gas it is monitoring rises in the air. Connecting the sensor to the Raspberry Pi 4 computer allows converting these changes in the conductivity to corresponding gas concentrations that can be understood.

The increased popularity of edge computing devices connected to sensors and Digital Twin has led to the advancement and availability of cloud computing and storage technology with online documentation support with systems architecture to collect, translate, and store sensor data. Microsoft Azure Services (Azure IoT Hub, Azure Stream Analytics, and Azure SQL Database) are deployed in this study to transmit sensor data collected through the Raspberry Pi4 to the Azure SQL database.

To test the workflow required to establish the connection between the sensors, the edge device (Raspberry Pi 4), and Azure Services to transmit sensor data, the authors first tested the process using the Raspberry Pi Simulator available online at (https://azure-samples.github.io/raspberry-pi-web-simulator/). Microsoft has created an online Raspberry Pi Simulator to help users build and test their custom Raspberry Pi solutions. Currently, the Raspberry Pi simulator can only simulate 'temperature' and 'humidity' data.

Once the workflow to transmit and store the sensor data in SQL Database was tested, data from the actual sensors connected to the Raspberry Pi 4 computer is connected to the Azure IoT Hub and transmitted to the Azure SQL Database using the Azure Stream analytics. Using pre-installed Python IDE (Integrated Development Environment) in the Debian OS (Operating System) of the Raspberry Pi 4, a Python code (https://github.com/sunfounder) was added to capture data from the three sensors, translate the data into a human-understandable format, and transmit the data to Azure IoT Hub and Azure SQL Database. The Python code used to perform these tasks comprises two parts. The first part of the code is focused on collecting sensor data. As explained above, each sensor has a different working principle. This part of the code had different components for each of the sensors used. Each component will receive output from one of the sensors connected to a specific port in the Raspberry Pi 4 and convert the output into a human-understandable format. The code also had to specify the port number to which each sensor is connected to the Raspberry Pi 4 and the measurement unit of the data generated. For example, the open-source code used to extract information from the DHT11 sensor (for temperature and humidity) outputs data in degrees Celsius. The Python code used was modified to convert the temperature readings to degrees Fahrenheit.

The second part of the Python code is focused on transferring the data to Azure Services (Pankovs, 2021). A Python package is provided by Microsoft to assist with transmitting sensor data as messages to an asynchronous Azure IoT Hub client. This package is called "azure.iot.device.aio". This code has two distinct parts. One part focuses on establishing a link with a device created within an Azure IoT Hub using a "Primary Connection String", while the other part focuses on sending the sensor data to the Azure IoT Hub. To execute this part of the Python code, the python library "asyncio" and "azure.iot.device" had to be imported.

Microsoft Azure also provides services to store data from IoT sensors in a SQL database. Data in the SQL database are imported into Power BI Desktop and linked to the operating room model graphics. This allowed for visualization and manipulation of both the 3D model and near real-time sensor data.

While using the different Azure services, the authors recommend that end-users should provide meaningful names for different variables, including IoT Hubs created and data transfer jobs as well as SQL Servers, databases, and tables defined. Through this paper, the authors have proposed and used a naming convention that provides a meaningful, consistent way to name the different Azure services used for the implementation.

Once logged into the Azure portal, users need to first create an Azure IoT Hub account. A Hub will act as a central location for all the sensors and data with common functionalities. Users can create one or more hubs depending on how they want to organize and group sensors across their building or campus. As depicted in Fig. 10, data-transmitting devices can be grouped and assigned to one or more Azure IoT hubs. For example, sensors can be grouped by building, a specific room, floor, or zone in a building, or for a group of buildings in the entire campus.

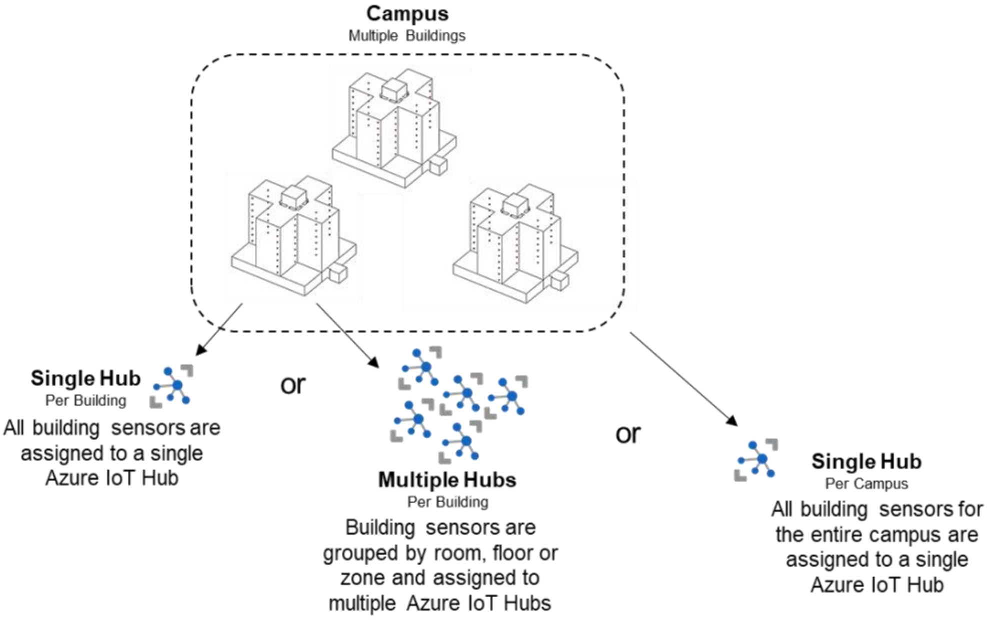

Campus
Multiple Buildings
Single Hub
Per Building
All building sensors are assigned to a single Azure IoT Hub
or
Multiple Hubs
Per Building
Building sensors are grouped by room, floor or zone and assigned to multiple Azure IoT Hubs
or
Single Hub
Per Campus
All building sensors for the entire campus are assigned to a single Azure IoT Hub

FIG. 10: Various proposed options to group and assign IoT/Edge devices and data to one or more Azure IoT Hubs.

The Azure IoT Hub service also allows users to define one or more ‘Resource groups.’ A resource group is a collection of Azure resources that share a common lifecycle, permissions, and policies. A resource group can also include Azure resources that the user wants to manage as a group or are part of a common workflow (Microsoft, n.d.-a). Fig. 11 shows two resource groups. Resource group 1 includes Azure Services for both buildings A and building B if the owner wants to manage these services together as a group. Resource group 2 includes Azure Services for only building C. In this case, the owner wants to manage these services separately.

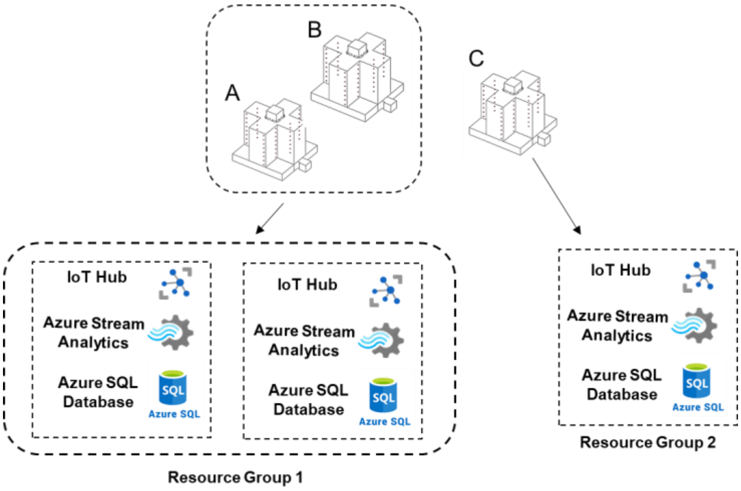


FIG. 11: Proposed use of Azure resource group

When naming IoT Hubs, the authors propose the following convention: "HubID-*field1/.../fieldx"*. One or more fields can be used. The number of fields used is user-dependent and may denote RoomNo, FloorNo, ZoneNo, BldgNo, and so on. For example: "Hub001-OR001-ABCHospital". For resource group naming, the convention proposed is: "ResourceGroup-*PermissionCategory/Class".* The category or class will be specific to the organization and may use its classification including 1,2,3… or A, B, C…. For e.g., "ResourceGroup-*CatA"*. Once created, the IoT Hub will be deployed and ready for connecting to IoT/Edge devices.

To link the configured edge device (the 3 sensors and the Raspberry Pi 4) to the established IoT Hub, each physical device is connected using a unique primary connection string to a corresponding virtual IoT device within the IoT Hub. We propose the following descriptive naming convention for each virtual edge device: "DeviceID-SensorType-RoomNumber-AssetID." This category-based naming would allow for tracking each device. The 'RoomNumber' field would be defined if the device is in a specific room or space (e.g., a temperature sensor located in an office or a mechanical room). The 'AssetID' field would be defined if the device is associated with specific equipment (e.g., a sensor measuring the output capacity of a chilled water pump). In the case study a device with a unique ID: SenRPi-001-DHT11-OR001 is connected to the established IoT Hub: Hub001-OR001-ABCHospital. The device "SenRPi-001-DHT11-OR001" corresponds to the temperature and humidity sensor DHT11. Once a device ID is created in the Azure IoT Hub, it will auto-generate a 'Primary Connection String,' This string is added to the sensor code to establish the connection between the physical device and the virtual IoT device within the Azure IoT Hub.

Sensor data linked to the IoT Hub needs to be stored in Azure SQL databases. Users can create one or more SQL servers for each facility they are monitoring. These servers can share the same resource group as the IoT Hub to have common permissions and policies. Server names can only contain lowercase letters, numbers, and '-'. The naming convention proposed for the SQL servers would be: "ServerID-OrganizationID." Example: "ser001-abchospital". While creating the SQL server, users need to create an admin login id and password to access the server in the future. Once the server is created, users can create multiple databases within the server. Each database can correspond to data generated by an individual sensor or a group of sensors, such as a single sensor or multiple sensors in a specific room, sensors providing a specific type of data (e.g., temperature), sensors connected to a specific building system, or sensors in a facility. The naming convention proposed for databases is: "DatabaseID-ServerID-*field1/.../fieldx"*. The number of fields can vary and may include RoonNo, FloorNo, ZoneNo, BldgNo, and so on to denote data collected from different sensors belonging to a specific location or building. E.g.: Db001-Ser010-OR001-ABCHospital.

Azure also allows users to add Azure Defender for SQL to provide their SQL database with additional security. Once the databases and servers are created and deployed, users can start sending the data from the device in the IoT Hub to the SQL database.

Before storing the data in the SQL database, a server firewall must be configured and a client IP must be added to the server to allow Azure services and resources to access the created server. Once the firewall is configured, the flow of data also needed to be created from IoT Hub to the SQL database. This is achieved by using Azure Stream Analytics to create jobs and move the data into the SQL database tables. The job naming convention proposed is: "JobID-HubID-DatabaseID". Example: "Job001-Hub001-Db001". The data generated from edge devices can be stored in a SQL table. Each table corresponds to a unique Stream Analytics Job created. The table naming convention proposed is: "TableID-HubID". Example: "Tbl001-Hub001". Data imported into these tables can be transformed and modified using SQL. Users can select which column and row to import based on their preference. Azure also allows users to test the data format before starting to stream and add the data into the SQL table. Before starting the stream job analytics, users can also choose the row count per batch inserted into the SQL table. Once the stream analytics job has started, the data can be continuously added to the SQL table, which can later be linked to Power BI.

Table 5 provides a summary of the proposed naming conventions used for the Azure services workflow needed to transmit the data from the IoT Hub to the Azure SQL database.

TABLE 5: Summary of proposed naming conventions

	Item	Naming Convention	Options	Example
1	Azure IoT Hub	“HubID-field1/…/fieldx”	Room number Floor number Zone number Bldg name/number/id	Hub001-OR001-ABCHospital
2	Resource Group	“ResourceGroup-PermissionCategory/Class”	1,2,3… A,B,C…	ResourceGroup-CatA
3	Device ID	“SensorID-SensorType-RoomNumber-AssetID”	N/A	SenRPi001-DHT11-OR001
4	SQL Server	“ServerID-OraganizationName” (only lower case alphabet, numbers, and “-“ are allowed in server naming)	N/A	ser001-abchospital
5	SQL Database	“DatabaseID-ServerID-field1/…/fieldx”	Room number Floor number Zone number Bldg name/number/id	“Db001-Ser001-OR001-ABCHospital”
6	Stream Analytics Job	“JobID-HubID-DatabaseID”	N/A	“Job001-Hub001-Db001”
7	SQL Table	“TableID-HubID”	N/A	“Tbl001-Hub001”

3. Importing Azure SQL database to Power BI Desktop and linking to model graphic elements

Sensor data stored in the Azure SQL database tables are imported into Power BI Desktop with a live link. The link allows for automatic updating of the data in Power BI at regular user-defined intervals as data from the edge device is refreshed and updated in the SQL database. This allows for a real-time transfer of data from the source (the edge device) to Power BI for visualization and manipulation through a set of dashboards. To allow for a live link, the Azure SQL table needs to be imported as a 'DirectQuery' into Power BI. Imported data is then, cleaned, and transformed using Power Query Editor tools. This data scrubbing process is necessary to allow for presenting the data efficiently using various visualization tools and to link the data to respective graphical elements representing sensors in the 3D model.

To attach sensor data to corresponding sensor graphical elements in the model, a new column is added in the sensor data table populated with the ‘objectId’ of the model element to which the sensor data belongs. Using this column, the sensor data tables can be connected to the VCAD_Asset table in Power BI, which includes a similar ‘objectId’ column (generated in 1). The relationship between these two tables will have cardinality "Many to One" and Cross filter direction as "Both".

Once the data tables and the model table were linked, Power BI reports and dashboards were generated to visualize air quality parameter data establishing a proof-of-concept Digital Twin for the OR. The following section discusses the various dashboards created to visualize real-time air quality parameter data and link them to their spatial location in the model.

8. DESIGN OF DIGITAL TWIN DASHBOARDS FOR VISUALIZING SENSOR DATA

The operating room 3D model and real-time sensor data are visualized in Power BI using various dashboards created to monitor the five air quality parameters in relation to their corresponding sensor spatial location in the model. The dashboards are created using various filters, navigation, and search options in Power BI to illustrate how graphics and data can be integrated and visualized for monitoring purposes. Fig. 12 displays the home page dashboard showing the full 3D model of the operating room and several visuals designed to show real-time data for the five air quality parameters.

The dashboard is divided into 3 areas. Area 1 mainly displays the latest date and time when the air quality sensor readings are captured. The 3D model is displayed in area 2. The model is interactive, and users can manipulate the model using navigational tools provided by the 3D model viewer. Area 3 displays the real-time data readings for the five air quality parameters. The Power BI gauge visualization tool is used to create the visuals. At the bottom of area 3, navigational buttons are added for each parameter and linked to the 3D model in the view. Selecting a bottom for any sensor will zoom in and highlight the sensor's graphical element in the model. Fig. 13 shows an example of how the buttons are linked to the model highlighting the location of the N_2O sensor in the 3D model when its navigational button is selected.

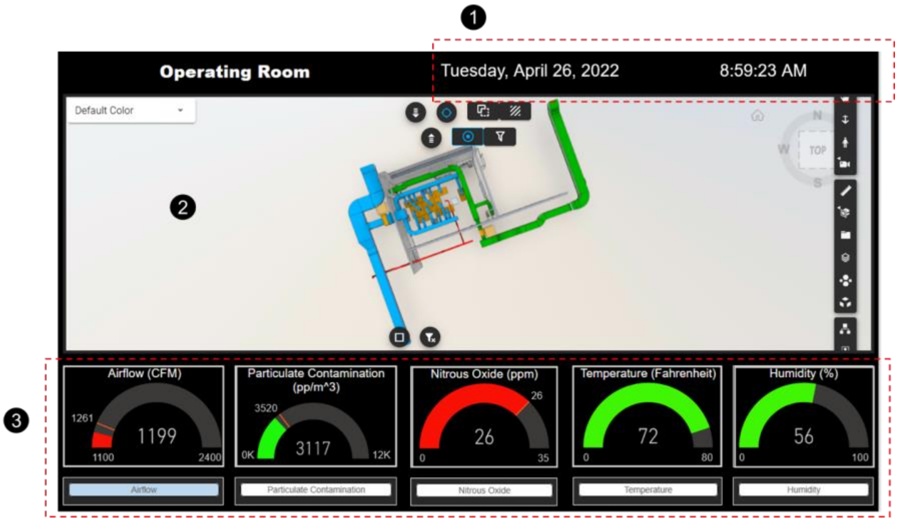


FIG. 12: Developed dashboard's home page

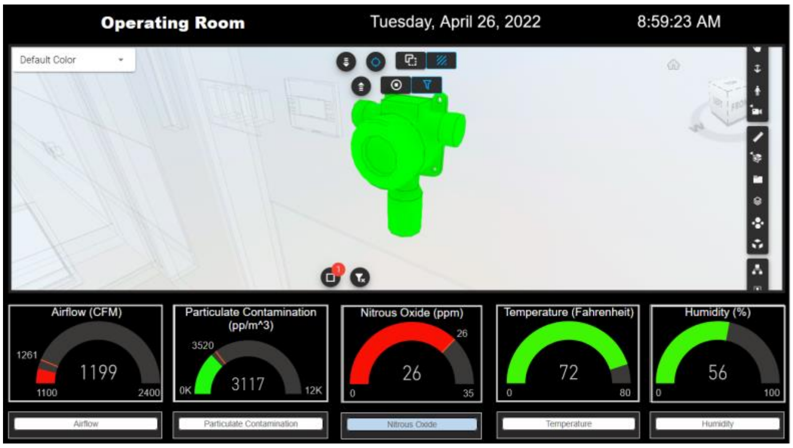
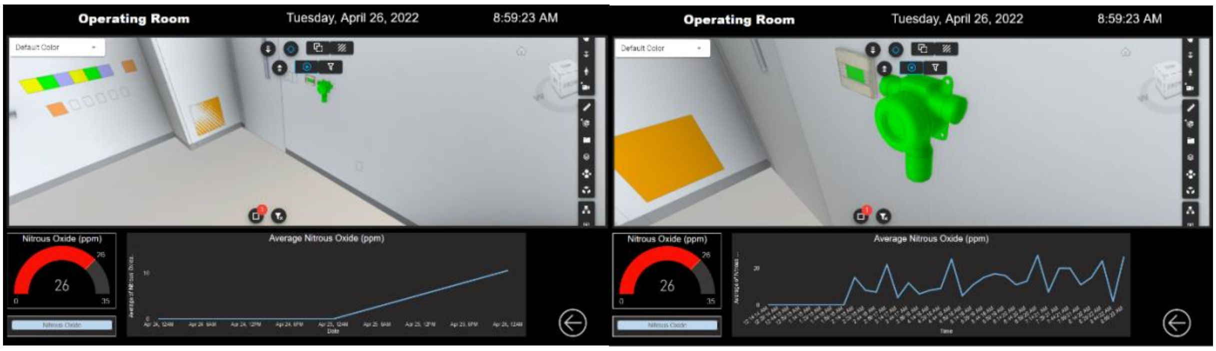


FIG. 13: Highlighting the location of the Nitrous Oxide (N2O) sensor in the model

To see details of air quality parameter values over a period, users can click on the respective gauge visual. This will navigate the user to a detailed dashboard that can display parameter variations over time using a line chart tool within a day or across several days. Fig. 14 shows two dashboards for the N_2O air quality parameter showing the average of the N_2O concentration for the last three days (Fig. 14(a)) or variation of the N_2O concentration every 15-minute interval on a particular day (Fig. 14(b)).



(a)

(b)

FIG. 14: (a) Variation of N_2O concentration for the last three days (b) Variation of N_2O concentration on April 24th every 15-minute interval

Users can navigate back to the dashboard home page by clicking the back button at the bottom right corner of the screen.

9. VALIDATION OF DEVELOPED DIGITAL TWIN PROTOTYPE

The prototype Digital Twin was presented to a highly experienced BIM manager of a healthcare facility. Comments and feedback on the implementation and the dashboard design were captured and recommendations for future work were discussed. During the recorded meeting, the BIM manager was first briefed on the overall implementation and provided a walkthrough demonstration of the Digital Twin dashboard interface. His comments regarding how the prototype can assist in monitoring OR air quality and improve response to facility maintenance requests were captured. During the meeting, the BIM manager also suggested several future modifications that included considerations for modeling multiple room types with the navigational ability to identify and separate data from sensors in different locations in the hospital. Table 6 provides a summary of the comments and feedback captured during the meeting.

TABLE 6: Summary of the comments and recommendations made by the healthcare BIM manager

Comments	Recommendations
1. The ability to locate the sensors corresponding to a particular reading in the 3D model is helpful because the facility manager would like to know the location of the sensor while addressing the facility maintenance request. (“Visually identifying the airflow location in the 3D model was a good feature because you (the user) always want to know the location of the sensor that is detecting airflow.”)	1. For the future expansion of the Digital Twin when including multiple rooms add navigational and identifiable features that display which room the sensor is reading airflow from. ("In the future, this is only one room but if there are multiple rooms you should be able to identify airflow sensors by room number so that we know which room or sensor we are looking at.”)
2. The developed Digital Twin can assist in a quick response to issues like a drop-in airflow during the surgery. ("Just on the off-chance that the air handler on the roof had a catastrophic failure, this would be good if you have multiple rooms served by the AHU and you would start detecting this across all these rooms that the air handler is serving. What happens if doctors are in the middle of surgery and the air handler fails? The Digital Twin will detect the issue before a doctor notices what was going on”) (“I don’t know if our OR control rooms have tied into the air handler like if it fails an alarm will go off. I don’t think that happens, it may go off to the facilities people but they are usually at the back of the building that is even if they are monitoring it, so this is a good job (feature)”).	2. Change the title of the location navigational button. ("the airflow location button says “Airflow” maybe it can say “Airflow Location” or “Click here for Airflow Location” because right now it looks like a label rather than a button”.)

10. DISCUSSION

The case study focused on the use of Power BI, VCAD, and Microsoft Azure services to create a Digital Twin of an Operating Room. The tools are chosen and the tool-based system architecture defined for the implementation considered the 3 components proposed for the creation of a Digital Twin as outlined by (Takyar, n.d.) and discussed in Section 3: Design, Operation, and Augmentation.

1. Design: The OR model was provided to the authors by the BIM manager of the healthcare facility. The model included architectural, interior, and mechanical systems present in the OR and authored using Autodesk Revit. This satisfied the “Design” component for the creation of a Digital Twin.
2. Operation: The goal and function of the proposed Digital Twin were clearly outlined at the start of the research work and intended to visualize real-time air quality data collected using several sensors connected to a Raspberry Pi 4 computer. This would allow monitoring of air quality in critical spaces of a hospital such as operating rooms. A Navisworks model was imported into the Power BI platform using VCAD. Telemetry data from the sensors were transferred using the Azure IoT services and stored in an Azure SQL database. Data were subsequently imported into Power BI and linked to the OR model. Power BI tools were used to develop various dashboards to view the data in relation to the BIM.
3. Augmentation: The software platforms used are comprehensive, flexible, and support the development of a Digital Twin. VCAD uses Autodesk Forge API to create Power BI template files and supports various model file types, including Revit, Navisworks, and IFC. Integration of the Raspberry PI 4 and sensor data with Azure services, including Azure IoT Hub, Azure Stream Analytics, and Azure SQL database was

> lengthy and complex but provided a robust implementation of the twin and promises greater flexibility for future expansion and scalability. Another aspect of augmentation can be in the form of standardization of information that needs to be captured and analyzed by the Digital Twins to meet the needs of owners and facility managers. This standardization can take the form of ISO standards like ISO 19650, which details a framework to manage information exchange, recording, versioning, and organization using a BIM. Since we are still at the onset of the Digital Twin implementation our goal for this paper was to test several tool-based system architectures to create a prototype. Once, more understanding regarding the development of an effective and secure Digital Twin is established in the A/E/C/FM industry, efforts can be made towards standardizing required information (static and dynamic information) for a Digital Twin based on the owner's and facility manager's goals. Future research efforts can focus on standardizing the Digital Twin information exchange, recording, and analysis including but not limited to analyzing current state of ISO 19650 and how it can be augmented to better support the adoption and implementation of Digital Twins.

Using the tool-based system architecture proposed, as well as potential others, users can build a scalable Digital Twin solution. For example, the authors used the free tier of Azure IoT Hub, which had a limited messaging allowance. Users looking to build a Digital Twin solution for a large project can use a standard or basic tier of Azure IoT Hub that can handle 400 thousand messages a day at level 1, to 300 million messages a day at level 3 (Microsoft, n.d.-a). In terms of data that can be imported into Power BI, the authors used a Power BI free license that had a data capacity of 1GB; this capacity can be increased to 10 GB (Giga Bytes) when using a Pro license allowing importing of larger models. Power BI also provides users with premium licenses that further increase the data storage capacity and processing power of Power BI (Microsoft, n.d.-b).

Microsoft Azure also provides multiple security features to its users to safeguard their data at various price ranges (Microsoft, n.d.-a). Some of these security features include:

1. Microsoft Sentinel: This provides users with a single solution for attack detection, threat visibility, proactive hunting, and threat response.
2. Microsoft Defender for Cloud: This provides users with a single dashboard that displays alerts and recommendations to act upon to assist with security operations.
3. Azure Advisor: This provides users with a personalized cloud consultant that can assist users with safely deploying their Azure solutions while providing recommendations to improve performance, security, and reliability.

Users are not limited to these resources and can use other resources to create a Digital Twin. There is a multitude of software and middleware platforms focused on providing Digital Twin and IoT services; users can select from any of the system architectures and solutions described based on their needs. Users can use their preferred software platform and services to develop each component of the Digital Twin. For example, to extract and stream real-time data from sensors, users can use Amazon Web Services (AWS) provided by Amazon in lieu of Microsoft Azure Services. Another example would be to leverage a more powerful Forge API to convert the BIM into Power BI files instead of Blogic s.r.l.VCAD. iConstruct is another software platform that allows users the ability to import and view their BIM models in Power BI through their Genus platform (iConstruct, n.d.). Exporting 2D or 3D Revit models along with BIM data and geometry to Power BI files is also possible with Pro-Revit Toolkit, a plugin to Autodesk Revit by Pro Tools (MG, n.d.).

Software platforms currently used to convert BIM into formats suitable for the creation of Digital Twins need also to be more customized and refined. Early testing of software to convert BIM into Power BI files showed that model elements created as nested families and assemblies in Revit did not translate accurately into the Digital Twin format. The nested families imported into the Digital Twin could not be selected as a whole, though data for both the nested families and their individual components was present in the Digital Twin. This could create a navigational issue if the users need to view the properties for the nested family as a whole. Similarly, assemblies created in Revit were translated as their individual components in the Digital Twin, hence, losing any property that was associated with the assembly. Godager et al. (2022) conducted a study that explored the key concepts discussed in ISO 19650 related to different information requirements, different information level needs, and common data environment to strengthen Enterprise BIM. Their paper highlighted that the Project Information Model acts as input for developing an Asset Information Model. Therefore, models delivered at project development stage need to be created with the understanding of how they can be utilized during operation and maintenance stage. This mindset during model creation and delivery can assist in using software platforms and data structure to provide better interoperability. It also allows to avoid loss of critical information when utilizing the model for the creation of Digital Twins.

As more and more research are conducted on the benefits and applications of Digital Twins in the A/E/C/FM industry, more standardized practices and documentation are required. For a more effective implementation of Digital Twins, the A/E/C/FM industry will have to understand what results they want to achieve from a Digital Twin implementation and what resources are available.

While discussing the case study in this paper, the authors proposed a standard naming convention for Azure Services, used to facilitate a better understanding and identification of Azure services involved with the created Digital Twin. Further research needs to be conducted to develop standard procedures to support the streamlined development, implementation, and operation of Digital Twins. For example, sensor data are linked to model elements using a unique object ID in Power BI. These IDs are manually added to the sensor data table. This would become laborious work for developing a Digital Twin with hundreds (building) or thousands (campus) of sensors. This process can be enhanced through automation if a standard naming convention for sensor IDs is developed and defined during both the model creation and the setup of the sensor hardware. This would allow us to automatically map each sensor to its corresponding modeled element object ID directly. This example shows the importance and need for future research efforts to develop a standard for Digital Twin creation, information exchange, storage, and analysis like ISO 19650.

The case study discussed in this paper showcase the utilization of the proposed system architecture to create a low-level Digital Twin. Integrating BIM geometry and data from multiple sources to develop this Digital Twin was not an automated process. Multiple software had to be manually integrated to exchange data. For example, Navisworks geometry had to be uploaded to VCAD to be processed and generate a Power BI template to be able to visualize the 3D model within Power BI. Another, aspect of preparing the Digital Twin for this case study is the cleaning and transforming of the data. Data cleaning, also known as data scrubbing, is the process to make data more useable and readable to the software focused on removing duplicates, fixing incorrect data format, and removing or replacing missing values. Data transformation is the process of converting data from one format or structure into another. This will allow for developing the required visualizations more efficiently and creating the desired Power BI reports and dashboards. This process, currently mainly manual, is an important step towards effectively visualizing data and drawing insights.

11. CONCLUSION

Digital Twins are envisioned to integrate data from multiple sources and analyze this data to accurately represent the operation of the physical asset or space. One of the challenges faced by practitioners is the inability to evaluate and identify the appropriate technological capabilities of the Digital Twin. The authors have discussed different levels into which Digital Twins can be categorized based on the complexity of real-time data integration, the ability to perform analysis on the data, the ability to perform simulations to predict issues, and the ability to suggest operational improvements.

The paper defined two unique objectives: (1) propose a functionality-based and multiple tool-based system architectures that can support the creation of the Digital Twin in the A/E/C/FM industry, and (2) use a case study approach to create a Digital Twin using one of the proposed tool-based system architecture as a proof on concept.

Currently, there are no specific, structured approaches to developing a Digital Twin, and tools for the creation of a Digital Twin are diverse. Given the multiple definitions of a Digital Twin, the authors in this paper had to custom-create a specific functionality-based system architecture. Based on the literature review conducted and the availability of commercial tools, a functionality-based and several tool-based system architectures are proposed, to create Digital Twins which adds to the novelty of this paper. These system architectures can incorporate data from multiple sources and link that data to BIM. The multiple tool-based system architectures can be used to develop low-level Digital Twins that can be further improved and expanded by adding data analysis and prediction functionalities. Each suggested system architecture provides users with a workflow to create their own Digital Twin based on the available resources, expertise, and required functionality from the Digital Twins. Creating a higher-level Digital Twin requires significantly more resources and expertise. But the system architectures provided in this paper can act as a guiding foundational step toward introducing and kick-starting the development of a Digital Twin to support and enhance various facility management processes. As more organizations move forward with the adoption of Digital Twins, more resources and efforts need to be allocated toward research and implementation for a higher-level twin. Future research will explore increasing the data integration and analysis capability of the current twin and test its effectiveness in supporting tasks like energy efficiency and advanced air quality monitoring.

The case study of an OR Digital Twin presented in this paper act as a proof of concept and validation for the creation of a low-level Digital Twin using a proposed tool-based system architecture. The study introduced a functioning prototype Digital Twin of an OR created in Power BI Desktop using Azure Services and 3rd party software that included VCAD and iConstruct. The developed prototype can connect to multiple sensors connected to an edge device (sensors and Raspberry Pi 4 setup) through databases, such as Azure SQL Database, and visualizing sensor information along with 3D model elements by using Power BI's built-in data visualization and analysis tools. The case study shows how telemetry data from sensors used to measure air quality in OR can be accessed, processed, stored, and visualized in real-time using Power BI. Once the low-level Digital Twin of the OR was developed, it was presented and discussed with a healthcare BIM manager to understand if the prototype twin can provide vital, real-time insights into the performance of operating rooms in healthcare facilities.

The Digital Twin developed for the case study assumed that the different sensors used for measuring air quality are mounted on the OR wall and inside the return grill. It should also be noted that monitoring of N_2O and other anesthetic gases to ensure air quality can be achieved using other methods including “diffusion badges or small battery-powered air samplers attached to OR occupants and worn while performing their duties” (Sheriff, 2020). Additionally, “infrared spectrophotometers can be used to detect leaks in the gas supply system or to ensure the effective operation of the waste gas scavenging system” (Sheriff, 2020).

In addition to monitoring N_2O, it is important to monitor O_2 levels to prevent the possibility of fire. A leak in the OR equipment can lead to a rise in the O_2 concentration and greatly increase the risk of fire. OSHA put the level of O_2 in operating rooms between 19.5% and 23.5%. The current study has not included the measurement of N_2O or O_2 levels. Future implementation will expand on the current work to explore the integration of O_2 data and include additional parameters for additional gas measurement methods.

ACKNOWLEDGEMENT

The authors would like to thank Michael B. DuLaney for sharing his experiences and knowledge related to air quality requirements in operating rooms as well as his feedback on the prototype demonstration. The authors would also like to sincerely thank the following individuals for their support and for providing an academic license of their software that helped make this research possible: Francesco Necciari, BLogic (VCAD), and Rob Gadbaw, iConstruct. The views and findings expressed in this paper are those of the authors and do not reflect those of the individuals that provided support to this research.

REFERENCES

Abdelrahman M. M., et al. (2022). Personal thermal comfort models using digital twins: Preference prediction with BIM-extracted spatial–temporal proximity data from Build2Vec, *Building and Environment,* 207**,** 108532.

Abdelrahman M. M. and Miller C. (2022). Targeting occupant feedback using digital twins: Adaptive spatial–temporal thermal preference sampling to optimize personal comfort models, *Building and Environment,* 218**,** 109090.

Agrawal A., et al. (2022). Digital twin in practice: Emergent insights from an ethnographic-action research study, *arXiv preprint arXiv:2203.07030*.

Ahmed V., et al. (2018). Challenges and drivers for data mining in the AEC sector, *Engineering, Construction and Architectural Management*.

Akanmu A. A., et al. (2021). Towards next generation cyber-physical systems and digital twins for construction, *J. Inf. Technol. Constr.,* 26**,** 505-525.

Almatared M., et al. Digital Twin in the Architecture, Engineering, and Construction Industry: A Bibliometric Review. Construction Research Congress 2022, 2022. 670-678.

Alshammari K., et al. (2021). Cybersecurity for digital twins in the built environment: current research and future directions, *Journal of Information Technology in Construction*, 26**,** 159-173.

Arup 2019. Digital Twin Towards A Meaningful Framework. ARUP.

Autodesk 2020. Autodesk Tandem™ Brings Digital Twin To Building Information Modeling. Autodesk.

Autodesk. n.d. *Autodesk* [Online]. Available: https://www.autodesk.com/ [Accessed November, 23 2022].

Autodesk University. n.d. *Autodesk University* [Online]. Autodesk. Available: https://www.autodesk.com/autodesk-university/ [Accessed November, 23 2022].

Blogic. n.d. *VCAD Connecting BIM and BI* [Online]. Available: https://www.bimservices.it/ [Accessed November, 23 2022].

Boje C., et al. (2020). Towards a semantic Construction Digital Twin: Directions for future research, *Automation in Construction,* 114**,** 103179.

Booch G. 2005. *The unified modeling language user guide,* Pearson Education India.

Bray B. 2021. Autodesk Tandem Digital Twin Platform Launches for AEC. Available from: https://adsknews.autodesk.com/news/tandem-public-beta [Accessed Febuary, 22 2022].

Broo D. G., et al. (2022). Design and implementation of a smart infrastructure digital twin, *Automation in Construction,* 136**,** 104171.

Brown J. 2021. *The Future of Construction: 10 Technologies and Trends to Watch in 2021* [Online]. Construction Executive. Available: https://www.constructionexec.com/article/the-future-of-construction-10-technologies-and-trends-to-watch-in-2021 [Accessed November, 23 2022].

Broz P. 2019. *Working with 2D and 3D Scenes and Geometry in Forge Viewer* [Online]. Available: https://forge.autodesk.com/blog/working-2d-and-3d-scenes-and-geometry-forge-viewer [Accessed November, 23 2022].

Buildingsmart International. n.d. *Take BIM Processes to the next level with Digital Twins* [Online]. buildingSMART International. Available: https://www.buildingsmart.org/take-bim-processes-to-the-next-level-with-digital-twins/ [Accessed November, 23 2022].

Bujari A., et al. IPPODAMO: A Digital Twin Support for Smart Cities Facility Management. Proceedings of the Conference on Information Technology for Social Good, 2021. 49-54.

Chiachío M., et al. (2022). Structural digital twin framework: Formulation and technology integration, *Automation in Construction*, 140**,** 104333.

Cityenith. n.d. *Cityenith* [Online]. Available: https://cityzenith.com/ [Accessed November, 23 2022].

Coriani S. 2020. Build your full-PaaS IoT solution with Azure SQL Database. Available from: https://devblogs.microsoft.com/azure-sql/build-your-full-paas-iot-solution-with-azure-sql-database/ [Accessed October 12, 2022].

Darling G. 2021. *IoT vs. Edge Computing: What's the difference?* [Online]. IBM. Available: https://developer.ibm.com/articles/iot-vs-edge-computing/ [Accessed November, 23 2022].

Deria A., et al. Integrating AI in an Audio-Based Digital Twin for Autonomous Management of Roadway Construction. Construction Research Congress 2022, 2022. 530-540.

Ding L., et al. (2018). A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory, *Automation in construction,* 86**,** 118-124.

Durão L. F., et al. Digital twin requirements in the context of industry 4.0. IFIP international conference on product lifecycle management, 2018. Springer, 204-214.

Ellis F. (1963). The control of operating-suite temperatures, *Occupational and Environmental Medicine,* 20**,** 284-287.

Elsehrawy R., et al. (2021). A digital twin uses classification system for urban planning & city infrastructure management, *Journal of Information Technology in Construction*, 26**,** 832-862.

Ensafi M., et al. A Modeling Methodology Towards Digital Twin Development in Smart Factories for the Industry 4.0 Human Augmentation Experiments. Proc. of the Conference CIB W78, 2021. 11-15.

Godager B., et al. (2022). Towards an improved framework for enterprise BIM: the role of ISO 19650, *Journal of Information Technology in Construction (ITcon),* 27**,** 1075-1103.

Hunhevicz J. J., et al. (2022). Digital building twins and blockchain for performance-based (smart) contracts, *Automation in Construction,* 133**,** 103981.

Ibm. n.d. *What is a digital twin?* [Online]. IBM. Available: https://www.ibm.com/topics/what-is-a-digital-twin [Accessed November, 23 2022].

Iconstruct. n.d. *iConstruct* [Online]. Available: https://iconstruct.com/ [Accessed November, 23 2022].

Jiang Y., et al. (2022). Digital twin-enabled real-time synchronization for planning, scheduling, and execution in precast on-site assembly, *Automation in Construction,* 141**,** 104397.

Kaewunruen S., et al. (2020). Digital twin aided sustainability and vulnerability audit for subway stations, *Sustainability,* 12**,** 7873.

Kaewunruen S., et al. (2018). A digital-twin evaluation of net zero energy building for existing buildings, *Sustainability,* 11**,** 159.

Kamari M. and Ham Y. (2022). AI-based risk assessment for construction site disaster preparedness through deep learning-based digital twinning, *Automation in Construction,* 134**,** 104091.

Khajavi S. H., et al. (2019). Digital twin: vision, benefits, boundaries, and creation for buildings, *IEEE access,* 7**,** 147406-147419.

Khankari K. (2018). Hospital operating room ventilation systems, *ASHRAE Journal,* 60**,** 16-26.

Kim J. and Ham Y. Real-Time Participatory Sensing-Driven Computational Framework toward Digital Twin City Modeling. Construction Research Congress 2022, 2022. 281-289.

Kitchin R. (2014). The real-time city? Big data and smart urbanism, *GeoJournal,* 79**,** 1-14.

Kritzinger W., et al. (2018). Digital Twin in manufacturing: A categorical literature review and classification, *IFAC-PapersOnLine,* 51**,** 1016-1022.

Li M., et al. (2021). Digital twin-driven virtual sensor approach for safe construction operations of trailing suction hopper dredger, *Automation in Construction,* 132**,** 103961.

Lu Q., et al. From BIM towards digital twin: strategy and future development for smart asset management. International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing, 2019. Springer, 392-404.

Lu Q., et al. (2020). Digital twin-enabled anomaly detection for built asset monitoring in operation and maintenance, *Automation in Construction,* 118**,** 103277.

Ma G. and Wu M. (2019). A Big Data and FMEA-based construction quality risk evaluation model considering project schedule for Shanghai apartment projects, *International Journal of Quality & Reliability Management*.

Madakam S., et al. (2015). Internet of Things (IoT): A Literature Review, *Journal of Computer and Communications,* Vol.03No.05**,** 10.

Madubuike O. C., et al. (2022). A review of digital twin applications in construction, *Journal of Information Technology in Construction (ITcon),* 27**,** 145-172.

Menassa C. C. (2021). From BIM to digital twins: A systematic review of the evolution of intelligent building representations in the AEC-FM industry, *Journal of Information Technology in Construction (ITcon),* 26**,** 58-83.

Mg. n.d. *Pro Tools* [Online]. Available: https://info.mg-aec.com/pro-tools [Accessed November, 23 2022].

Microsoft. n.d.-a. *Azure documentation* [Online]. Available: https://learn.microsoft.com/en-us/azure/?product=popular [Accessed November 2023 2022].

Microsoft. n.d.-b. *Technical documentation* [Online]. Available: https://learn.microsoft.com/en-us/docs/ [Accessed November, 23 2022].

National Institute for Occupational Safety Health 1994. Controlling Exposures to Nitrous Oxide During Anesthetic Administration. US Department of Health and Human Services, Public Health Service, Centers ….

Onset. n.d. *Onset HOBO and InTemp Data Loggers* [Online]. Available: https://www.onsetcomp.com [Accessed November, 23 2022].

Pan Y. and Zhang L. (2021). A BIM-data mining integrated digital twin framework for advanced project management, *Automation in Construction,* 124**,** 103564.

Pankovs J. 2021. *Sending Telemetry to Azure IoT Hub* [Online]. Available: https://jev-pankov.com/2021/01/24/sending-telemetry-to-azure-iot-hub/ [Accessed November, 23 2022].

Peng Y., et al. (2020). Digital twin hospital buildings: an exemplary case study through continuous lifecycle integration, *Advances in Civil Engineering,* 2020.

Pregnolato M., et al. (2022). Towards Civil Engineering 4.0: Concept, workflow and application of Digital Twins for existing infrastructure, *Automation in Construction,* 141**,** 104421.

Raspberry Pi. n.d. *Raspberry Pi* [Online]. Available: https://www.raspberrypi.com/ [Accessed November, 23 2022].

Romano F., et al. (2020). Operating theatre ventilation systems and their performance in contamination control:"at rest" and "in operation" particle and microbial measurements made in an Italian large and multi-year inspection campaign, *International Journal of Environmental Research and Public Health,* 17**,** 7275.

Salem T. and Dragomir M. (2022). Options for and Challenges of Employing Digital Twins in Construction Management, *Applied Sciences,* 12**,** 2928.

Seghezzi E., et al. (2021). Towards an occupancy-oriented digital twin for facility management: test campaign and sensors assessment, *Applied Sciences,* 11**,** 3108.

Shariatfar M., et al. Digital Twin in Construction Safety and Its Implications for Automated Monitoring and Management. Construction Research Congress 2022, 2022. 591-600.

Sheriff R. 2020. *Air Quality in Operating Rooms and Surgical Suites* [Online]. Available: https://www.atlenv.com/air-quality-in-operating-rooms-and-surgical-suites [Accessed November, 23 2022].

Siemens. n.d. *Digital Twin for facilities and infrastructure* [Online]. Available: https://new.siemens.com/global/en/products/buildings/digital-building-lifecycle/ecodomus-software.html [Accessed November, 23 2022].

Sobhkhiz S. and El-Diraby T. Developing BIM-Based Linked Data Digital Twin Architecture to Address a Key Missing Factor: Occupants. Construction Research Congress 2022, n.d., 11-20.

Song Y. and Li Y. Digital Twin Aided Healthcare Facility Management: A Case Study of Shanghai Tongji Hospital. Construction Research Congress 2022, 2022. 1145-1155.

Stanford-Clark A., et al. 2019. *What are digital twins?* [Online]. Available: https://developer.ibm.com/articles/what-are-digital-twins/ [Accessed November, 23 2022].

Takyar A. n.d. *A complete knowledge guide on Digital Twins* [Online]. LeewayHertz. Available: https://www.leewayhertz.com/digital-twin/ [Accessed November, 23 2022].

Tariq R., et al. (2022). Digital twin models for optimization and global projection of building-integrated solar chimney, *Building and Environment,* 213**,** 108807.

Teizer J., et al. The Concept of Digital Twin for Construction Safety. Construction Research Congress 2022, 2022. 1156-1165.

Tohani M. and Metcalfe D. 2019. Smart Innovators: Digital Twins For Industrial Facilities. Verdantix.

Van Eck N. J. and Waltman L. (2011). Text mining and visualization using VOSviewer, *arXiv preprint arXiv:1109.2058.*

Walker G. and Strathie A. (2016). Big data and ergonomics methods: a new paradigm for tackling strategic transport safety risks, *Applied ergonomics,* 53**,** 298-311.

Xie X., et al. (2020). Digital twin enabled asset anomaly detection for building facility management, *IFAC-PapersOnLine,* 53**,** 380-385.

Yuan X. and Anumba C. J. (2020). Cyber-physical systems for temporary structures monitoring, *Cyber-physical systems in the built environment***,** 107-138.

Zaballos A., et al. (2020). A smart campus' digital twin for sustainable comfort monitoring, *Sustainability,* 12**,** 9196.

Zhang J., et al. (2022). Automatic relative humidity optimization in underground heritage sites through ventilation system based on digital twins, *Building and Environment,* 216**,** 108999.