Automation in Construction 135 (2022) 104117

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

BIM-based decision support for building condition assessment

Hamidreza Alavi [a,*], Rafaela Bortolini [a,b], Nuria Forcada [a]

[a] Department of Project and Construction Engineering (DPCE), Group of Construction Research and Innovation (GRIC), Universitat Politècnica de Catalunya (UPC), Colom, 11, Ed. TR5, 08222 Terrassa, Barcelona, Spain

[b] School of Architecture and Urbanism, Universidade Federal de Pelotas, Pelotas, Brazil

ARTICLE INFO

Keywords:
Building information modelling
Data model
Unified modelling language
Bayesian networks
Building condition
Facility management
Decision support
Visualization

ABSTRACT

Building condition assessment requires the integration of various types of data such as building characteristics, the properties of elements/systems and maintenance records. Previous research has focused on identifying these data and developing a building condition risk assessment model based on Bayesian networks (BN). However, due to interoperability issues, the process of transferring the data is performed manually, which requires considerable time and effort. To address this issue, this paper presents a data model to integrate the building condition risk assessment model into BIM. The proposed data model is implemented in existing software as a case study and tested and evaluated on three scenarios. Addressing interoperability will leverage the BIM tool as a data repository to automate the data transfer process and improve its consistency and reliability. It will also enable BIM to be a more effective tool for building condition and causality analysis visualization.

1. Introduction

In the life cycle of a project, the operation and maintenance (O&M) phases are as important as the planning and construction of the project itself. Compared with other phases, the highest costs occur during the O&M phase [1], which shows the importance of Facility Management (FM) activities. In the broad context of FM, building maintenance is generally recognized as the main activity, since more than 65% of the total cost of FM comes from facility maintenance management [2]. There are some challenges in current FM practices that have required a paradigm shift in the sector in recent years. Clients are demanding strategies for predicting events instead of responding to problems [3]. This shift marks the transition from corrective or planned strategies to preventive and predictive strategies. The failure of building elements can be predicted by preventive maintenance through an analysis of condition data and historical maintenance records. This increases their efficiency, reliability and safety [4].

Buildings tend to deteriorate unless they are properly maintained. The lack of a preventive maintenance plan and the building's natural aging accelerates the degradation of existing buildings [5,6]. The application of maintenance actions is imperative to prevent defects and failure of building elements and to extent the service life of the materials [7]. A condition assessment system is used primarily to facilitate the ranking of all the elements of the asset according to the amount of repair that is needed, which is detected during an inspection, and to produce consistent, relevant, useful information [8].

A Bayesian network (BN) method can be used to simulate cause-effect relationships of uncertain factors that impact building conditions. The BN is a probabilistic graphical model that offers a framework for reasoning about partial beliefs in uncertain situations [9]. It is regarded as a strong technique for modelling risks, based on uncertain data [10–12]. In a reasoning process, the BN can represent complicated linkages among building elements and systems, and qualitatively and quantitatively characterize variable dependencies. In addition, it can model a building's condition as a probabilistic process, contrary to deterministic models [13]. Bortolini and Forcada [13] developed a probabilistic model based on a BN that covers several interconnected elements for assisting decision-making on building maintenance and retrofitting measures to improve building conditions. Although the model can handle uncertainty and make predictions, the data that is required is dispersed among platforms. What is worse, the data is transferred manually, which is a laborious, inefficient process [14–16].

Building Information Modelling (BIM) can be a unit of the overall system architecture to solve the issues of information reliability for maintenance operations [15] and help decision-makers to address building maintenance concerns. BIM is "an approach to design, construction, and facilities management, in which a digital representation of the building process is used to facilitate the exchange and

* Corresponding author.
E-mail address: seyed.hamidreza.alavi@upc.edu (H. Alavi).

https://doi.org/10.1016/j.autcon.2021.104117
Received 6 July 2021; Received in revised form 28 November 2021; Accepted 23 December 2021
Available online 31 December 2021

interoperability of information in digital format" [1]. BIM, integrated with a Decision Support System (DSS), may constitute a powerful methodology to support the selection of strategic management activities [17,18]. Nevertheless, the greatest obstacle of this integration is the lack of interoperability in the O&M context [19,20].

To tackle this issue, this research presents a data model to enable interoperability between BIM models and the building condition risk assessment model based on BN. The research provides the system architecture to implement the data model in a case study. The integration of BIM with BN models facilitates data transfer and reduces the time and effort that the FM team spends on manual input. It also allows BIM tools to visualize the building elements/systems in an integrated, interactive way for decision-makers. Moreover, it helps the FM team to optimize building operation strategies and supports decision-making on FM activities (e.g., predictive maintenance) to improve building performance. The first step of this paper was to identify the required data for the BN model. Then, BIM and BN models were integrated, based on the proposed data model to assess building condition and visualize the current condition of the building elements and systems, established within Revit Software and employing a color scale. Finally, a case study was used to test and validate the proposed data model on three scenarios.

2. Literature review

2.1. Decision support system for O&M

A DSS can be used to make decisions in an early design development stage and during the O&M phase. The former helps designers to identify multiple technical and commercial options that are compliant with predetermined specifications and the latter help facility managers to optimize building operations techniques [21]. To support decision-making on building condition assessment, Matos et al. [22] prioritized maintenance actions, using Key Performance Indicators (KPI) and a support tool. During the O&M phase, existing studies utilized probabilistic models to make decisions on improving building condition. Frederik et al. [23] created a probabilistic model that learns from user feedback and adapts to the users' specific preferences over time to analyze building conditions. Yang et al. [24] developed a probabilistic model based on a comprehensive survey of air handling unit (AHU) fault detection and diagnosis methods. Lee et al. [11] developed a Bayesian method for probabilistic occupant thermal preference categorization and prediction in office buildings, to provide predictions for personalized thermal preference profiles. Bortolini and Forcada [13] developed a model for assessing the condition of a building using a Bayesian network (BN) method. Despite the fact that these researchers have made a significant contribution, none of them automatized the data transfer process or integrated BIM into their probabilistic models, which would facilitate data transfer due to the interoperability issues [25].

2.2. Information standards for O&M

BuildingSMART, the worldwide industry body, has developed a standard data format, the Industry Foundation Classes (IFC). The IFC data model is intended to describe architectural, building and construction industry data and has been mostly used as the data exchange schema between BIM and other systems such as Computerized Maintenance Management Systems (CMMS) and electrical instrumentation control (EIC) [3,26–29].

The Construction Operations Building Information Exchange (COBie), a subset of IFC data, is an international standard for exchanging data from the design phase to the O&M phase using a formal spreadsheet. The version of COBie for the FM handover Model View Definition (MVD), [30] is the MVD delivered in a file format that can be viewed and edited in Microsoft Office Excel [31]. However, it allows for the storage of a large volume of different kinds of data, which results in overloading [32]. Accordingly, COBie needs to be customized for facility information as a means to building operation [33]. Becerik-Gerber et al. [34] showed that each FM activity is data-intensive and requires specific data requirements. Kim et al. [35] focused on identifying specific data for FM maintenance activity and proposed a data management approach to integrate IFC objects, COBie data, and maintenance work information from the FM system database.

2.3. BIM interoperability for O&M

Efforts to address BIM interoperability for O&M have been made by many researchers. Ahmed Gouda et al. developed a framework by employing semantic web technology to store maintenance information and BIM data using COBie [36]. Cheng et al. [37] determined FM information requirements referring to the Information Delivery Manual (IDM) and developed an integrated data-driven system based on BIM and IoT technologies for predictive maintenance of building facilities using COBie and the IFC extension. To enhance decision-making in FM, Chen et al. [2] proposed a system for automated maintenance work order scheduling, based on BIM and FM software using COBie and the IFC extension. Marmo et al. developed a framework to address the interoperability issue by mapping the IFC into a relational database for maintenance and performance management [29]. Other researchers developed applications on BIM by integrating various systems to execute maintainability analysis [38–41], indoor localization [42], fire emergency simulation and analysis [43–45], fault detection and diagnosis [46,47], sustainability assessment [48,49], and energy simulation and forecast [50–52].

The variety of standards and technologies available (i.e., building automation protocols such as BACnet, Modbus, ZigBee and C-Bus) is one of the BIM–O&M interoperability problems [19]. Hence, many researchers have focused on system-based approaches to address the specific interoperability issue between BIM and software systems, standards or protocols in the O&M phase [51,53–55]. The system-based approaches propose a systematic architecture for data integration [56]. Such approaches make full use of open libraries, components and commercial software tools, and implement data integration architecture [56]. Kang and Hong [56] proposed system architecture to effectively integrate BIM into geographic information system (GIS)-based FM software. Such approaches make full use of open libraries, components and commercial software tools, and implement data integration architecture [56]. Motawa et al. [57] developed system architecture to collect data and knowledge about building maintenance activities while and after they are performed. Lee and Cheng et al. [44,58] presented a system architecture to integrate BIM with Barcodes and Radio-Frequency Identification (RFID) tags to enable timely data access. Quinn et al. [59] proposed system architecture to extract data from a Building Automation System (BAS) and incorporate it in BIM using a linked data structure. However, research on the integration of building condition assessment and BIM is scarce. Only Ani et al. [60] integrated information from a survey on a water ponding defect on a flat roof to the BIM model to identify the flat roof condition.

With respect to visualization, Tashakkori et al. [61] integrated BIM-based 3D indoor navigation functions with the proposed emergency management systems. Moreover, Wang et al. [43] applied the same approach to find the escape route to support fire safety management of buildings. Oti et al. [62] utilized color scheme visualization in BIM to visualize data related to the energy management systems, to reflect time-dependent energy consumption information. Regarding maintenance activities, some researchers utilized BIM 3D visualizations to locate building components and support troubleshooting in proposed maintenance systems [43,63]. In conclusion, although many studies address BIM interoperability, none of them focus on interoperability between BIM and building condition assessment.

3. BN model for building condition assessment

With the aim of assessing the entire condition of a building, Bortolini and Forcada [13] developed a BN model for building condition assessment. This model was created using cause-and-effect relationships between uncertain elements that impact building conditions. The condition of building elements and systems was categorized as high, medium or low. For example, the term "high condition" refers to a piece of equipment that is in high working order and can be used to its maximum potential for its intended function. The BN model to assess building conditions is presented in Fig. 1. Hierarchical levels could be visualized in the model that include all the general civil and architectural elements, as well as MEP (mechanical, electrical, and plumbing) systems.

The development of the model required several cycles of analysis, implementation and verification. Once the variables that have the most impact on building condition had been identified, several methods were used in these steps. These included obtaining quantitative (real data from existing buildings, statistical analysis and literature reviews) and qualitative evidence (a survey with domain experts). To legitimate the inference of cause-effect relationships between nodes, a database on 1974 building defects and 5373 maintenance requests from 40 buildings was used. Finally, to check and improve the model structure, experts in the field of building performance and facility management were interviewed. The model was then refined after rounds of questions with feedback and consensus between experts. For this purpose, nine experts were interviewed in the field of building pathology and facility management. All interviewees had over ten years of experience in facility management, consultation and maintenance activities. The detailed process of the model development can be consulted in [13].

The BN model was divided into building elements and systems. The building elements were classified as: 1) structure, 2) façade, 3) roofing, 4) flooring, 5) interior partitions and 6) doors/windows. The building systems were also defined as follows: 1) electrical systems, 2) plumbing systems, 3) HVAC systems, 4) elevator and 5) fire systems.

Variables that impact the performance of building elements and systems were classified as: design and construction errors; policy for building operation and maintenance; defects in building elements/systems; environmental agents; and building properties including age, type of elements, and whether or not preventive maintenance actions are planned. Weather conditions, the surrounding environment, the danger of natural catastrophes and geological conditions are examples of environmental agents.

In the BN model, variables that impact the condition of building elements and systems were represented as nodes. Depending on the data type, they were defined as discrete (labeled, Boolean, discrete real or ranked) or continuous [64]. Some nodes were defined as ranked and had various states such as 'High', 'Medium', and 'Low'. Others were specified as Boolean, with binary states like 'Yes' and 'No'. For whatever element or system condition, the model can be queried by inserting evidence in the BN model and setting its state (i.e., low condition). Then, the BN calculates the probability function of the parent nodes by conducting backward propagation, and estimates the most likely causes (e.g., age of the equipment, lack of preventive maintenance and design errors).

In this study, the TNormal distribution was utilized to determine the probability distribution. When the mean (μ) and variance (σ^2) are determined, TNormal is a suitable distribution since it allows for the creation of many distribution forms [64]. Unlike the regular Normal distribution, TNormal has finite endpoints that range from 0 to 1 in equal intervals. The variance parameter reflects the influence of parent nodes' uncertainties. In the simplest case, the parameter mean is determined as a weighted mean of the parent nodes with the following expression:

$$Wmean = \frac{\Sigma i = 1 \ldots n\ wiXi}{n} \tag{1}$$

where wi ≥ 0 are weights, and n is the number of parent nodes.

The BN structure was constructed by identifying the causal relation between the variables based on the data available and expert judgment. A panel of experts provided feedback on the causal relations constructed by data, which helped to identify key variables or processes that were overlooked and fix potential errors of the model. Conditional probability tables for the variables can be consulted in [13]. AgenaRisk was chosen to construct the BN model for building condition assessment, due to its

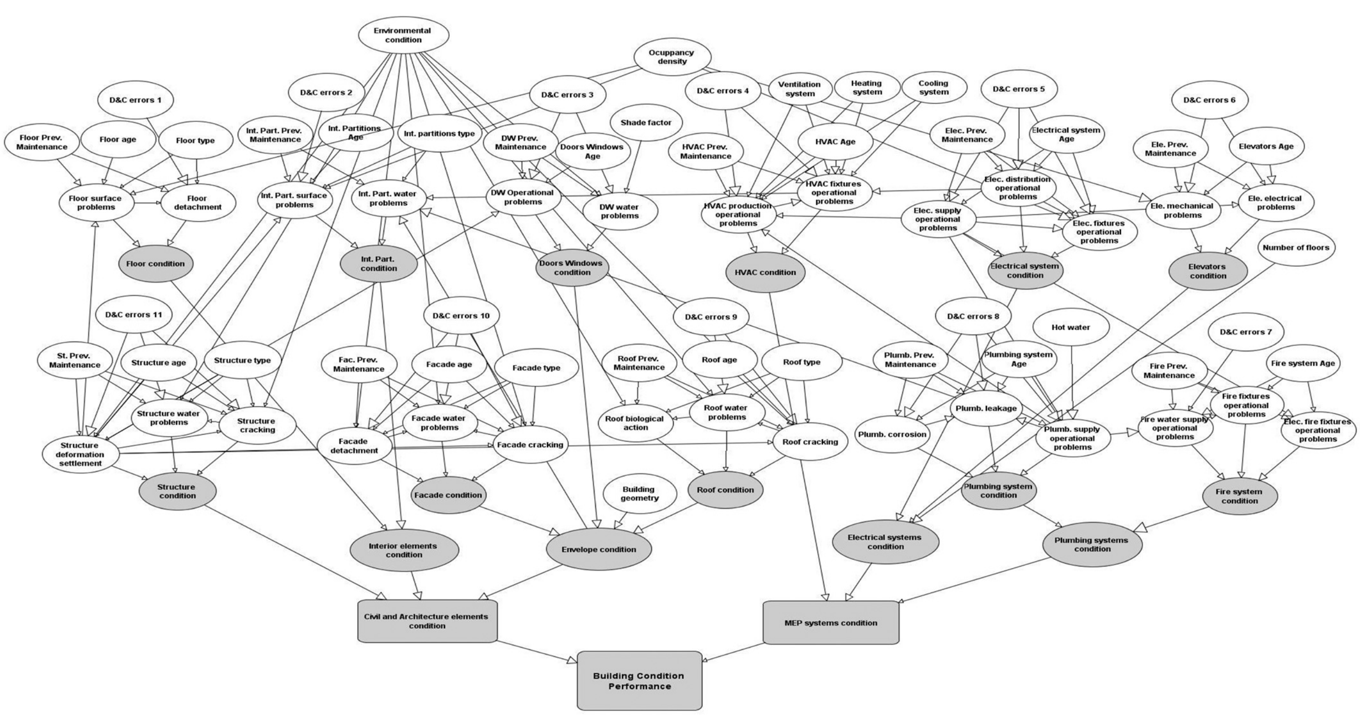


Fig. 1. BN model for assessing a building's condition assessment [13].

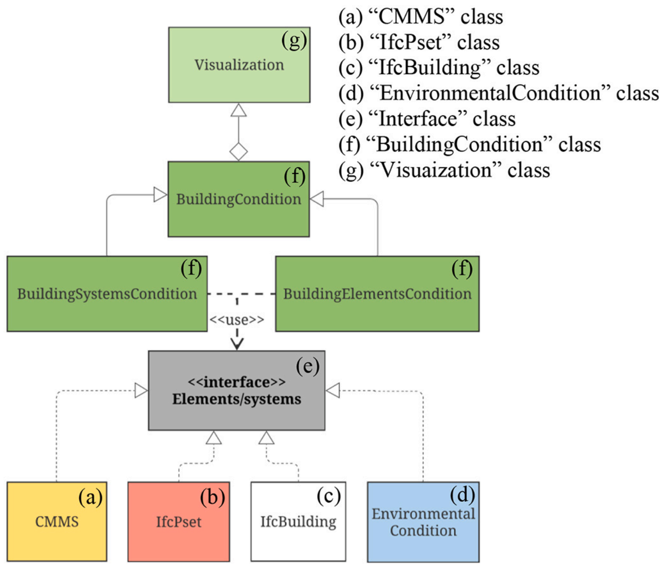


Fig. 2. Conceptual design of the UML diagram for the proposed data model.

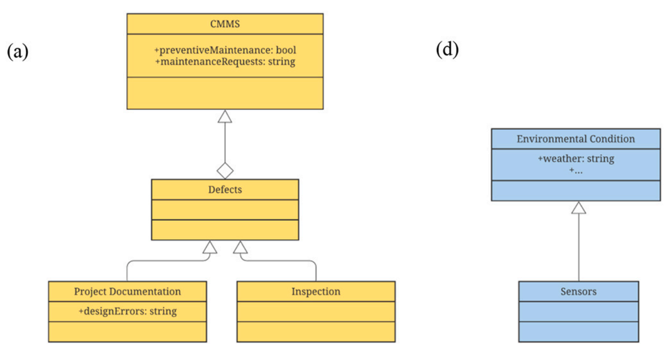


Fig. 3. UML diagram of (a) "CMMS" class and (d) "EnvironmentalCondition" class.

power, versatility and user-friendly interface [65]. It can visualize the sensitivity analysis for the BN model to represent the importance of causal factors.

4. Methodological approach

A data model was designed to integrate risk condition assessment into BIM. Then, the data model was implemented into existing BIM tools and finally it was validated in three scenarios. The data model consisted of seven thematic classes, namely: "BuildingCondition", "CMMS", "EnvironmentalCondition", "IfcBuilding", "IfcPset", "Interface" and "Visualization". A Unified Modelling Language (UML) class diagram, which is a worldwide industry standard [66], was employed to present the data model. A class diagram in the UML is a type of static structure diagram that describes the structure of a system by showing its classes, attributes and behavior (e.g., operations). Fig. 2 highlights the conceptual design of the proposed data model for BIM and BN model integration. In Fig. 2, the "interface" class for building elements/systems merged all data sources and transformed data into the appropriate format by creating new attributes to support compatibility of a BIM model and a BN model. To create new attributes, algorithms for various data types such as Number, Boolean and String were created. These attributes were then required by the "BuildingCondition" class, using an interface to assess a building's condition.

To enhance the readability of the UML diagrams, classes were depicted in different colors, considering different data sources. The "CMMS" class (in yellow, [a]) includes maintenance requests and preventive maintenance records, which play an important role in identifying defects in building elements/systems. The "EnvironmentalCondition" class contains a sensor to obtain the weather conditions (in blue, [d]). Fig. 3 shows the UML diagram for "CMMS" and "EnvironmentalCondition" classes.

The "IfcBuilding" class (in white, [c]) is considered a major data exchange schema standard for BIM [67]. The IFC Property Set known as "IfcPset" is a class (in red, [b]) that contains required data on building condition assessments. These data are assigned to an IFC model object and their class names are preceded by the prefix IfcPset.

The "BuildingCondition" classes (in green, [f]) were divided into building system condition and building element condition for ease of reading, as shown in Fig. 4 and Fig. 5. Due to the complexity of the model and limitations of space, the attributes are not illustrated in the class diagrams (see the Appendix A for the complete data model).

The "BuildingCondition" classes that are based on causality analysis use "interface" class (in grey, [e]) to assess a building's condition. This requires the acquisition of data from various sources such as "CMMS", "IfcBuilding", "IfcPset" and "EnvironmentalCondition" classes, followed by the transformation of these data into an appropriate format.

UML diagrams for building system condition and building element condition differ according to their characteristics. For instance, the "IfcBuilding" class for building element condition is comprised of IFC for

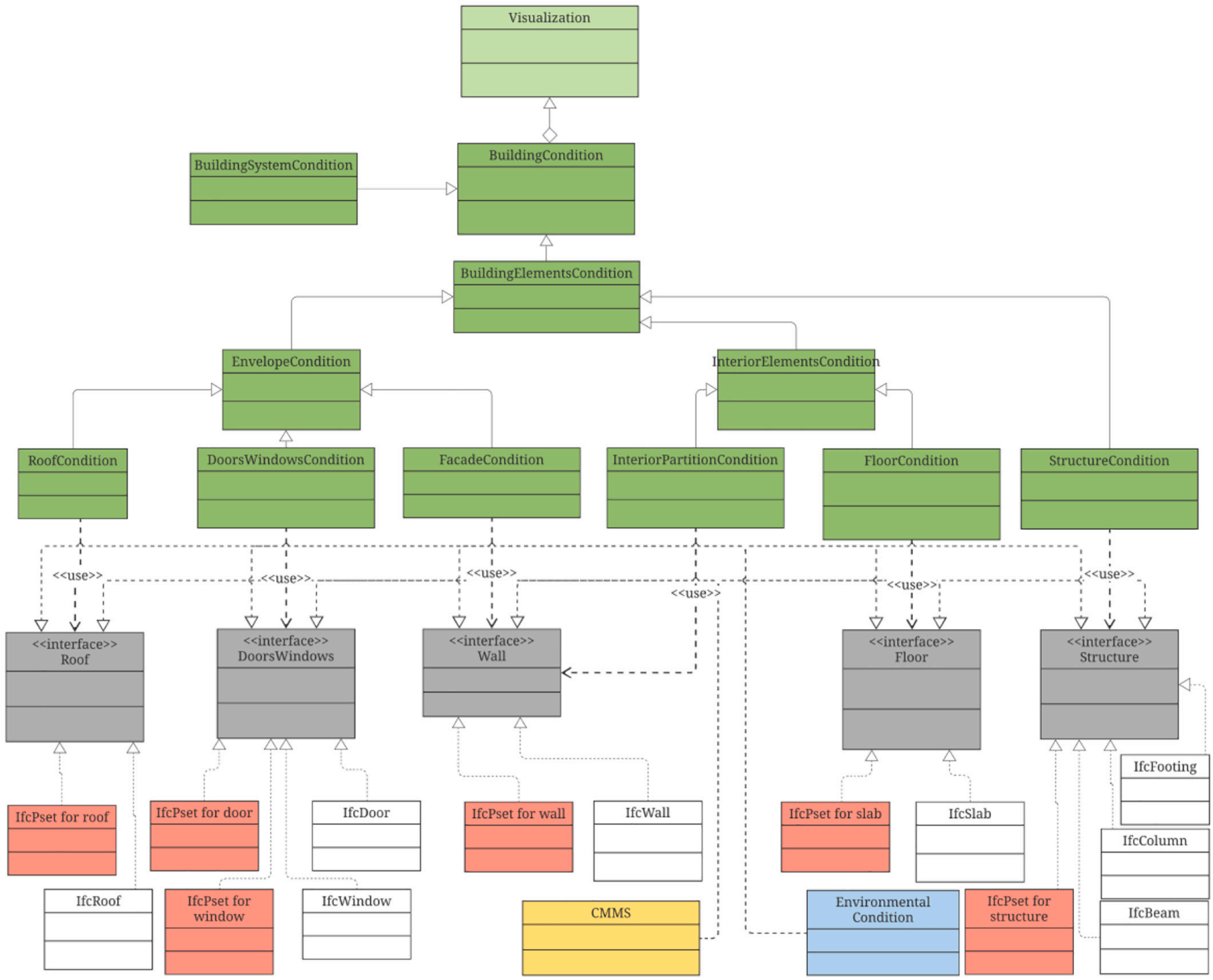


Fig. 4. UML diagram of the data model on the building element condition.

elements such as IfcDoor, IfcWindow, Ifcwall and IfcRoof, while for building system condition it consists of various IFC with respect to systems (e.g., IfcChiller, IfcDamper and IfcBoiler).

Finally, among all the thematic classes, the "Visualization" class (light green, [g]) represents a link through which the results of the building condition assessments can be imported into any possible data visualization tool.

5. Case study implementation

The data model was implemented in Autodesk Revit, which is one of the most popular BIM tools in the AEC sector. The system architecture of implementing the data model into Autodesk Revit (i.e., a BIM tool) to facilitate the assessment of building conditions consisted of three main steps, illustrated in Fig. 6. (1) The parameters for the Revit model were created as IfcPset, based on the required data for building condition assessment. (2) The Revit model was integrated with the BN model to evaluate building condition using Dynamo, a visual programming extension for Autodesk Revit, and Python programming language. (3) The BN results of the building condition assessment were exported to local storage and visualized in Revit in a way that the FM team can easily understand the data.

5.1. Parameter creation

To allow BIM models to contain the required data on building condition assessments, a Dynamo script was used to create parameters for data that could not be obtained from the BIM model, such as the age of each building element and system. All variables of the BN model were considered parameters in Dynamo. Fig. 7 shows the process of creating parameters to host relevant data in BIM using a Dynamo script.

In this study, the parameter name of the required data was exported from the BN model as an .XML file converted into Microsoft Excel (.xls), an intermediate format, before mapping it to the BIM model. Then, authors manually defined the parameter types and families to assign the required data into their relevant families in BIM. There were different kinds of parameter types, including Numbers, Strings and Yes/No Boolean. For example, the "HVAC age" parameter requires a numeric value since it contains the age of equipment. Therefore, its type was considered as "Numbers" and its corresponding family was defined as "Mechanical Equipment" in the BIM model. Next, the .xls file containing the parameter's name, type (i.e., data type) and family for each item of data, was imported into the Dynamo through a *Data.ImportExcel* node. Eventually, all the parameters were created in the BIM model based on the required data from the BN model to host relevant data using a *ParameterCreateSharedParameter* node in Dynamo.

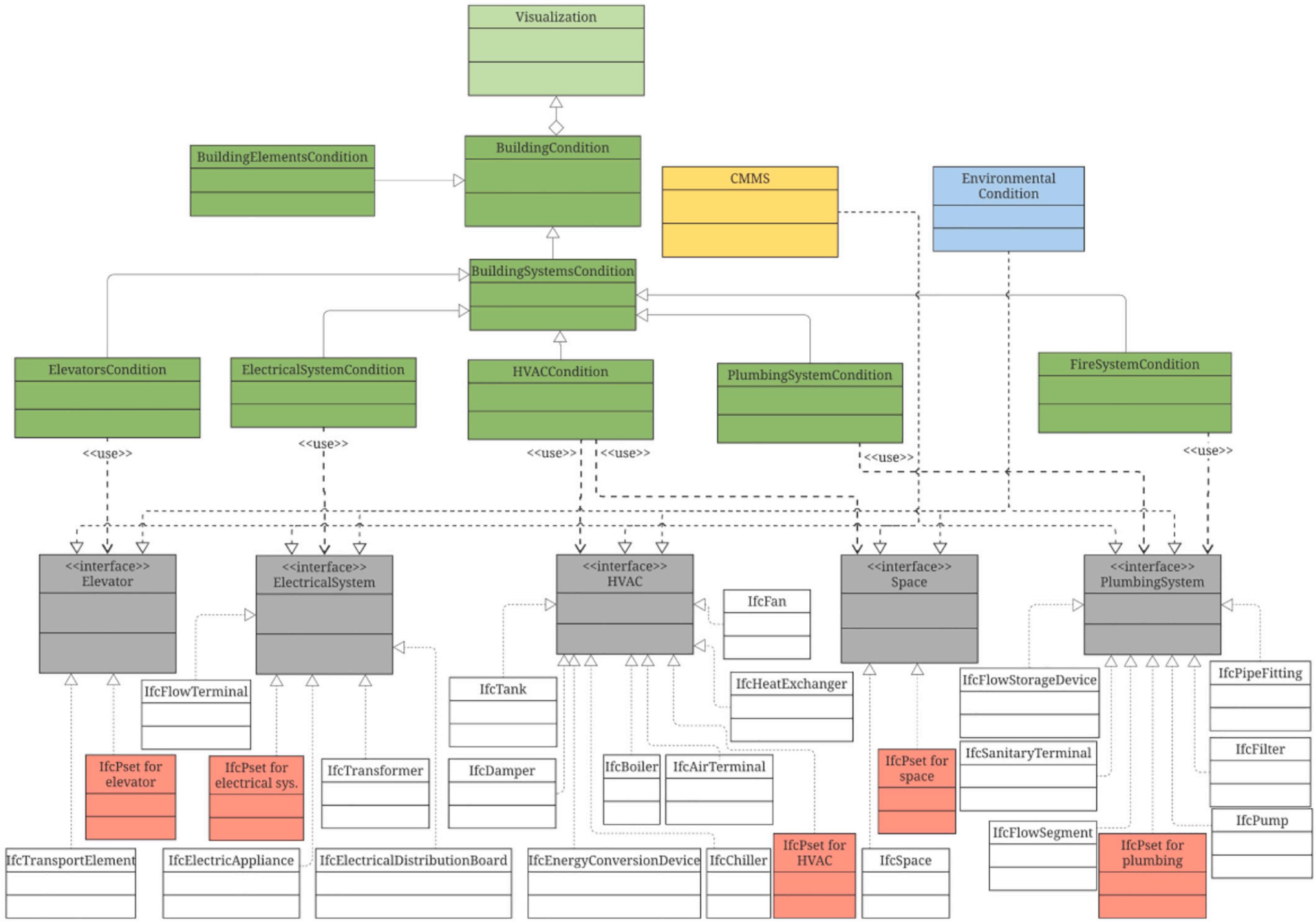


Fig. 5. UML diagram of the data model on the building system condition.

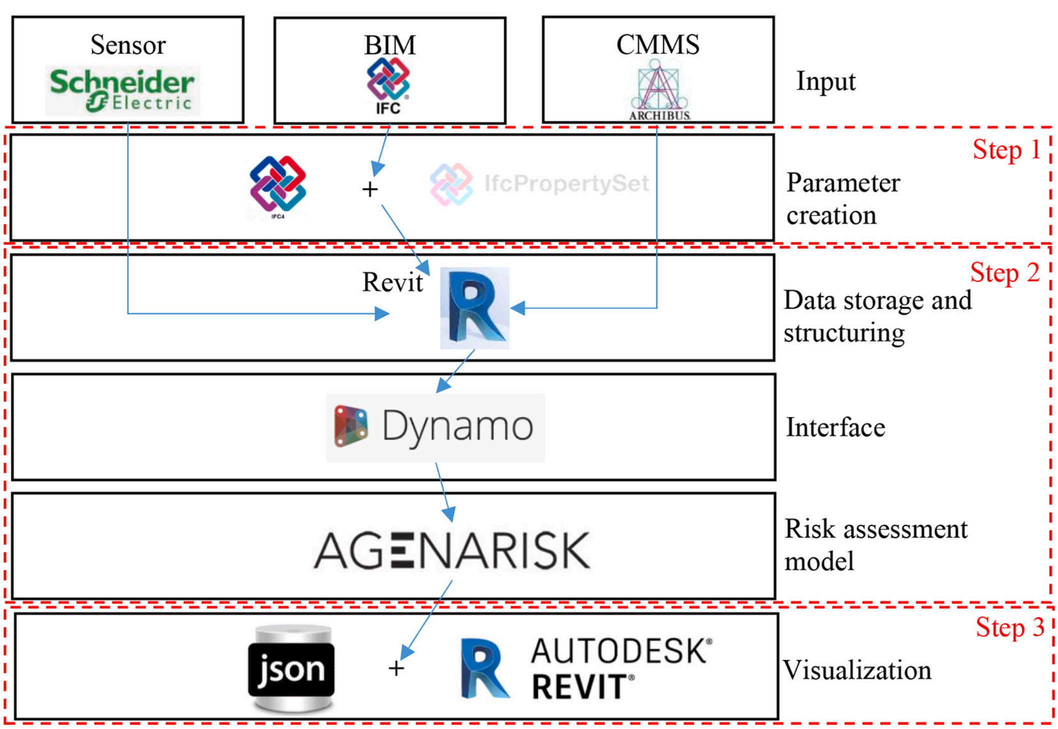


Fig. 6. System architecture for the integration.

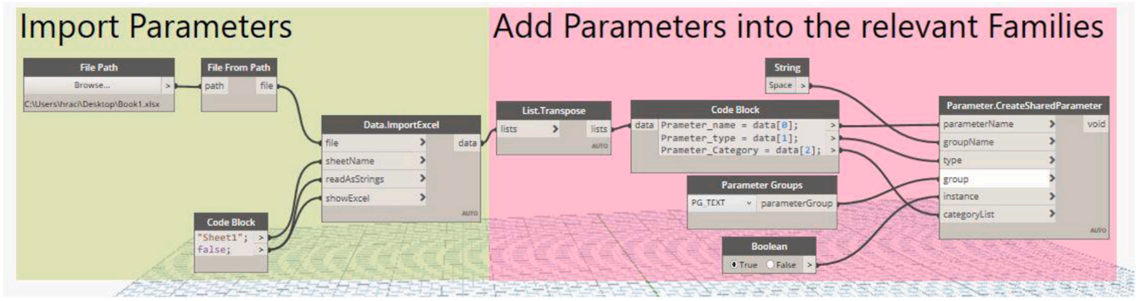


Fig. 7. Dynamo scripts to create parameters.

Table 1
Parameters in the BN model for building condition assessments.

Type	Parameters		States in the BN model
Boolean	Façade prev. maintenance	Elevators prev. maintenance	Yes, No
	Roof prev. maintenance	Structure prev. maintenance	
	Doors/windows prev. maintenance	Floor prev. maintenance	
	Cooling	Interior partitions prev. maintenance	
	Heating	Plumbing–hot water	
	HVAC prev. maintenance	Plumbing prev. maintenance	
	Electrical prev. maintenance	Fire system prev. maintenance	
String	Façade type		Concrete panels/ masonry, metal panels, glazed, others
	Roof type		Flat concrete, flat metal panels, glazed, others
	Ventilation		Forced, natural
	Structure type		Concrete, masonry, steel, others
	Floor type		Continuous, discontinuous, others
	Interior partition type		Masonry walls, light partition walls, others
Number	Façade age	Elevator's age	<10, 10 to 20, >20
	Roof age	Electrical age	
	Doors/windows age		
	Structure age	Interior partitions age	<10, 10 to 30, >30
	Floor age		
	Plumbing age	Fire system age	<3, 3 to 10, >10
	HVAC age		

Table 2
Examples of Python code blocks for transforming data.

Nodes	Type	States in BN	Locate in BIM	Python code block in Dynamo

Floor preventive maintenance	Boolean	Yes / No	Spaces	Yes_Count = list. count(data, 'Yes')
				No_Count = list. count(data, 'No')
				if Yes_Count > No_Count:
				result = "Yes"
				else:
				result = "No"

Floor age	Numbers	<10
		10 to 30
		>30
			Building elements	average = sum (data)/len(data)
				if average < 10:
				result = "< 10"
				elif average > 20:
				result = "> 20"
				else:
				result = "10 to 20"

Ventilation	String	Forced/ Natural	Spaces	if data == Forced result = "Forced" else: result = "Natural"

5.2. BIM-BN data integration

To transfer data between BIM and BN models bidirectionally, firstly the required data was extracted from the BIM model using Dynamo and Python scripts, by creating a dataset in a JavaScript Object Notation (.Json) format, which is a lightweight format for storing and transferring data. The dataset containing all the required data was then imported into the BN tool, AgenaRisk, which utilized the data as 'evidence'. Then, the FM team could run the BN model straightaway to acquire the results of analyzing the condition of a building. Secondly, the assessment results of a building's condition were extracted from the AgenaRisk tool into a Json format and imported into the BIM model using Dynamo and Python to visualize the results in a 3D model.

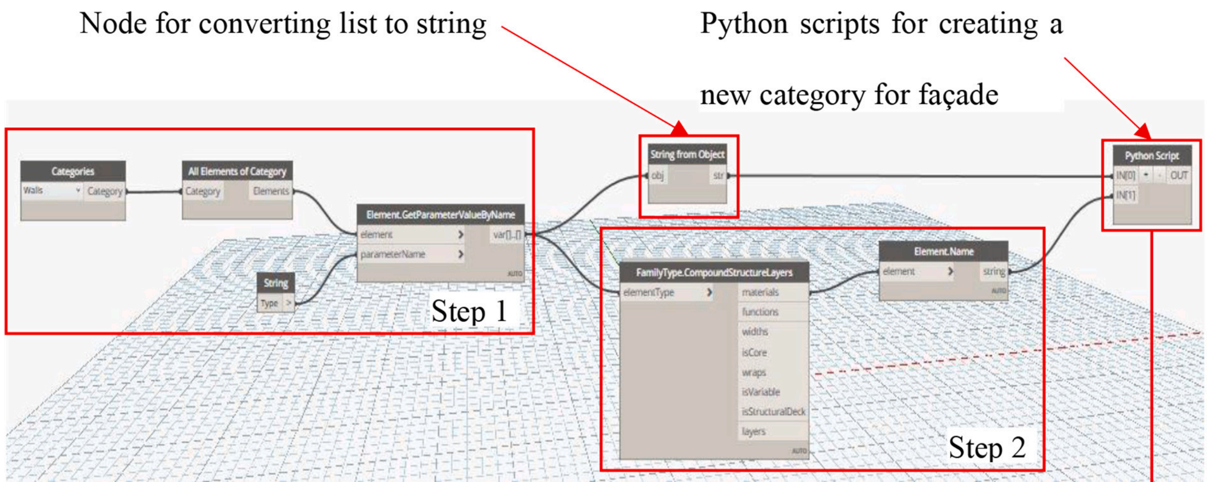
5.2.1. BIM data transfer processes

The options of whether or not to have preventive maintenance, cooling, heating and different kinds of data such as building properties (e.g., age, type of elements) can be obtained from the BIM model. However, before extracting these data from the BIM model, they must be transformed into an appropriate format so as to be compatible with the BN model. To achieve this, a bunch of Python scripts was designed in Dynamo to transform data from the BIM into the appropriate format. Dynamo is a scalable way to extract data from centralized spreadsheets and update common parameters with a range of data types including Boolean, Strings and Numbers. Table 1 shows all the parameters that need to be transformed to be utilized in the BN model.

For those data expressed in numbers (e.g., roof age, floor age), a Python code block was used to calculate the average age of all elements in BIM since the BN model evaluated the condition of entire buildings rather than a single element. For example, when one floor of a building is renovated, the "floor age" is determined by the Python code block calculating the average age of all floors in a building and transforming the results into an appropriate format for the BN model which is "<10" if the average age is less than 10 years, "10 to 30" if the average age is between 10 and 30 years, and ">30" if the average age is greater than 30 years.

For data expressed in a Boolean form (e.g., "Yes" having or "No" not having preventive maintenance, cooling or heating), a Python code block was designed to enumerate all the "Yes" and "No" for each Boolean to determine which one was repeated more than the other. For instance, for data on Having or not having heating in a room, all the rooms were considered in a Python code block and all "Yes" and "No" were enumerated to find out whether the building had heating or not. The most repeated answer was considered the result for the question of whether or not there was heating in the building. If the number of "Yes" and "No" were equal, the result would be considered "No".

A similar approach to Boolean and numbers can be used for strings




```
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
import sys
pyt_path=r'C:\Program Files (x86)\IronPython 2.7\DLLs'
pyt_path=r'C:\Program Files (x86)\IronPython 2.7\Lib'
sys.path.append(pyt_path)
# The inputs to this node will be stored as a list in the IN variables.
# Import re module to use regular expression
import re
# Declare the list contains subject code
ext = IN[0]
mat = IN[1]
extlength = len(ext)
matlength = len(mat)
newlist = []
sim = []
Nlist = []
regex = re.compile("Exterior")
filtered = [i for i in ext if regex.match(i)]
fiterlength = len(filtered)

for m in range(0,extlength):
    for k in range(0,fiterlength):
        if ext[m] == filtered[k]:
            sim.append(m)
for j in sim:
    if j not in Nlist:
        Nlist.append(j)
for adad in Nlist:
    newlist.append(mat[adad])

# Assign your output to the OUT variable.
OUT = newlist
```

Fig. 8. Dynamo scripts to create a new category in BIM for façade, as an example.

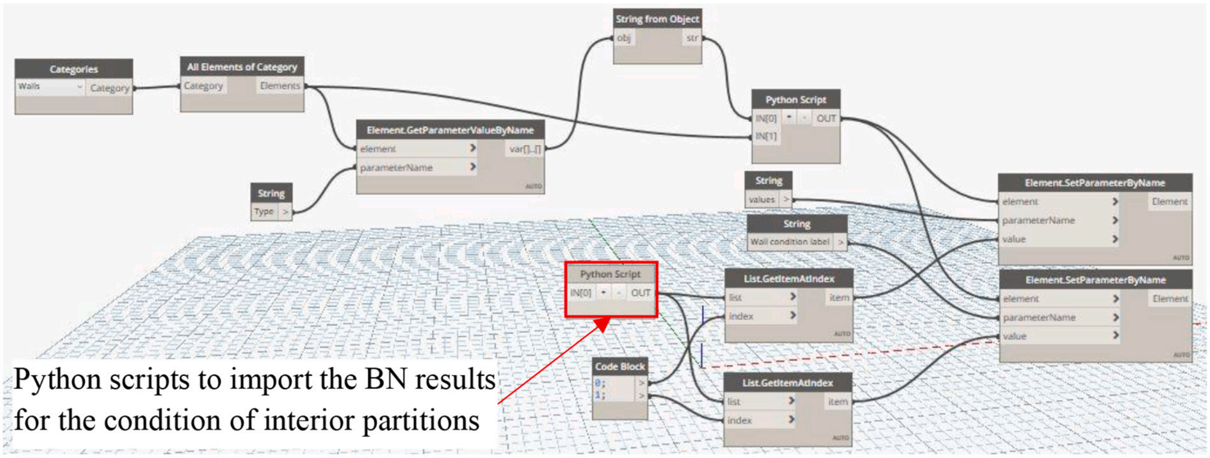
(e.g., ventilation). For example, the ventilation type in the BN model for buildings was either forced or natural. A Python code block recognized whether the building had forced ventilation or not. If not, the type of ventilation was considered natural. If there were more than two options (e.g., façade type), the "if…elif…else" statement could be used (i.e., the same as floor age). Table 2 shows an example of Python code blocks for floor age as numbers, floor preventive maintenance as Boolean, and ventilation as strings.

5.2.2. Mapping BN results into BIM

In accordance with the BN model, the BN results assessed conditions of entire buildings, comprising various groups of elements. For example, all windows and floors (i.e., different elements) in the building had to be taken into account to evaluate the condition of the window and floor respectively. Therefore, various categories were designed using the "Categories" node in Dynamo to match the results of the building condition assessment with the corresponding groups of elements in the BIM model.

Even though categories in BIM provide various groups of elements, some categories on building condition assessments based on the BN model still cannot be represented. Hence, Dynamo and Python scripts were used to create a new category for BIM to be compatible with the BN results. For instance, the BN model assessed the condition of either façade or interior partitions individually, both of which have the same category in the BIM model called "wall category". In this example, regular expressions, a sequence of characters that specifies a search pattern in a Python code block were designed to distinguish the wall category between the interior partitions and façade, to create new categories in BIM for both of them. Regular expressions utilize text to conduct pattern matching and "search-and-replace" operations. Fig. 8 illustrates an example of creating a new category in BIM for façade.

Three steps were imperative to create a new category for façade as an example. Firstly, a list of all wall elements for the building was created in the first step. Then, this list was converted to string using the "*String from Object*" node as a regular expression supports strings. A regular expression was used to find the "exterior walls" among the list by string-searching algorithms. Secondly, the material of all walls was obtained using *FamilyType.CompoundStructureLayers* and *Element.Name* nodes in Dynamo, creating a list of material (Step 2). Thirdly, the list of all wall elements was connected to the Python code block as input#0, and the list of material was connected as input#1. Next, a Python code block queried input#0 to filter a list by exterior walls (i.e., façade). Then, it




```
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
import sys
pyt_path=r'C:\Program Files (x86)\IronPython 2.7\DLLs'
pyt_path=r'C:\Program Files (x86)\IronPython 2.7\Lib'
sys.path.append(pyt_path)
# The inputs to this node will be stored as a list in the IN variables.
import json
lis_value = []
lis_label = []
with open('C:\Hamid\PhD Hamidreza\BN model\Export results for visualization\All visualization\Scenario 6 Safety Int. Part. condition.json', 'r+') as file:
    data = json.load(file)

for res in data['dataSets']:
    for resv in res['results']:
        for val in resv['resultValues']:
            lis_value.append(val['value'])
            lis_label.append(val['label'])
            for i in range(0,len(lis_value)):
                for j in range(0,len(lis_label)):
                        if lis_value[i] == max(lis_value):
                                x = lis_label[i]
                        else:
                                continue

# Assign your output to the OUT variable.
OUT = max(lis_value), x
```

Fig. 9. Dynamo scripts to map the BN results for interior partitions as an example.

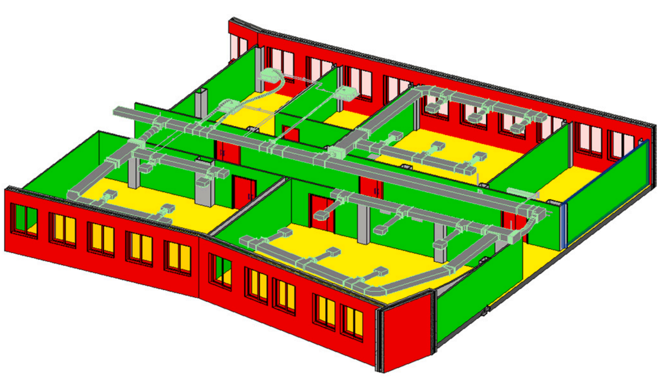


Fig. 10. The visualization of a building's condition in BIM.

queried input#1 to find materials that matched those from the exterior walls (input#0) and create a new list with the exterior walls and their corresponding material. Eventually, the category for façade was created to be consistent with the BN results of building condition assessments.

5.3. Data visualization

The results of the building condition assessments were extracted from the AgenaRisk tool into Json format and imported into Revit using a Python programming language in Dynamo to be matched with corresponding building elements. A Python code block queried the BN results to find the condition of elements categorized as high, medium or low. Then, the condition for each element was mapped to its corresponding elements in the BIM model using *GetItemAtIndex* and *SetParameterByName* nodes. Fig. 9 illustrates the process of mapping the BN results for interior partitions as an example.

Lastly, the BIM model visualizes the results with different colors to vary from 'High' to 'Low'. The tabulated data taken from Revit's schedule were visualized in a 3D format in the BIM model by applying view filters. For a given element, the relevant results of the building condition assessment were identified. Fig. 10 illustrates the BIM visualization of the building's condition as an example.

BIM visualization allows the FM team and owners to evaluate building and system elements based on the causality analysis using different color codes, where red represents a low performance condition, yellow a medium performance condition, and green a high-performance condition. The FM team would be able to filter the elements in the BIM model to view the color associated with their condition. It is also possible to compare building elements between different buildings.

6. Model evaluation

In this study, two types of evaluation were carried out as follows: (1) scientific data model validation and (2) software verification.

6.1. Scientific data model validation

The data model was validated by implementing it in three buildings (TR5, TR11 and TR14) of the Terrassa campus from the Universitat Politècnica de Catalunya (UPC), (Fig. 11). The campus includes 25 buildings with classrooms, offices, laboratories, dining rooms,

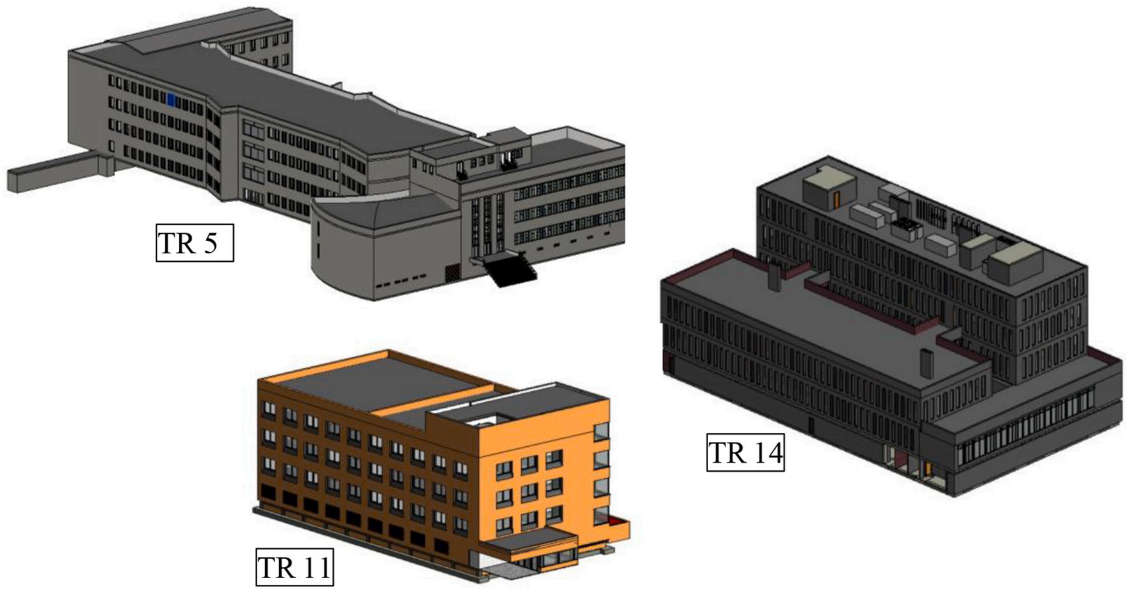


Fig. 11. Case study projects.

restrooms, common areas and study areas.

The consistency of the data model was validated by running the proposed system architecture in TR5, TR11 and TR14 and comparing the condition assessment results with those obtained from the existing manual method (e.g., AgenaRisk) in which the FM is required to perform the data transfer process manually. Furthermore, 18 scenarios per building were simulated and compared with the results obtained from the existing manual method.

The completeness of the data model was validated by [68]: a) checking the Json data formatting, b) checking either an empty or null value for each data item in the Json file.

The first step validates the Json data for correctness and provides a list of missing data in the validation report one after another, until all the required data is complete. In this study, a Json validator (e.g., http://jsonlint.com/) was used to validate Json data for formatting. If the data in the Json file is incorrect or incomplete, the validation will report a failure to assist in the debugging of Json data [69].

The second step ensures that all data have their value, demonstrating the data completeness. Once the Json data formatting had been evaluated, the value of the transferred data is checked to meet data completeness. If the value for data is missing, it should be presented either as empty ("") or a null value in the Json file. Therefore, the Json file is parsed using the json.load() method in Spyder [70]. The Python script is then used to check whether the value for each data is empty (null) or not.

The benefits of using the proposed system architecture (task efficiency analysis) were analyzed in terms of time reduction in comparison with the manual method. The advantages of the visualization in terms of intuitiveness were discussed.

All buildings were maintained by the same Facility Management company. Therefore, all have the same maintenance protocols. TR5 was constructed in 1960, it has five floors with 11,492 m^2; TR11 was built in 1997 and has 4 floors with a total area of 2779 m^2; and TR 14 was built in 2011 and is a six-story building with one parking lot and 7378 m^2.

Both TR5 and TR11 have a reinforced concrete structure, flat roofs and masonry façades, while TR14 has a metal panel façade. Regarding HVAC in TR5, most classrooms and offices have radiators, air-water systems and multi splits while TR14 is heated and cooled by fan coils, one chiller and two boilers. In TR11, there is no cooling system at all, and the ventilation is only natural, by opening windows.

To run the proposed system architecture, many parameters were created, such as the age of elements and whether or not they have preventive maintenance or ventilation. The Python code blocks were used to calculate the data required by the BN (square meters, average age, etc.). Other parameters were created to adapt the classification of the elements obtained from the BIM model to those required by the BN model. For example, the façade type was classified as "concrete panels/ masonry", "metal panels", "glazed" and "others".

As an example, to allow data integration and interoperability regarding the ventilation system, the algorithm for "HVAC interface" created new attributes to be compatible with the BN model considering the entire buildings. In TR5, for instance, since most rooms (e.g., offices, classrooms and corridors) have an air-water system, the new "Forced" attribute was created while in TR11 the new attribute was "Natural". For TR14, it was also considered as "Forced" on account of having air handling units (AHU) and fan coils in all rooms.

With respect to flooring, the algorithm for floor "interface" created new attributes for buildings. In TR14, the floor is "discontinuous" as it was constructed in various phases. For other buildings (TR5, TR11), the attribute is "continuous".

The BN results (condition of the building elements and systems) were visualized in the BIM model (Fig. 12) by the proposed system architecture and compared with those obtained using the AgenaRisk in which data are introduced manually. The system architecture showed the same results as the existing manual method but in a user-friendly way, allowing the FM team to quickly identify problems in buildings. Besides, 54 different scenarios for all three buildings (i.e., 18 scenarios for each building) were simulated in the proposed system architecture and the existing manual method. After running all these scenarios, the results in both methods were the same and thus confirmed the data consistency.

Regarding the task efficiency analysis, the same approach as Kang and Hong [56] was used in this study. To achieve this, two tasks were classified as follows. (1) BIM Data Transfer Process (BDTP), which transfers data from the BIM into the BN model. (2) Mapping BN results into BIM (MBB), which imports the BN assessment results into the BIM model to visualize a building's condition. Then, each task was timed and compared with the others listed in Table 3.

The time for performing the "BDTP" task, which is known as the most time-consuming, decreased nearly 100% in all buildings when the proposed system architecture was used. This shows the importance of automation of data transfer. In general, using the existing manual method, it took 39.5 h for TR5, 31.7 h for TR11 and 24.8 h for TR14 to

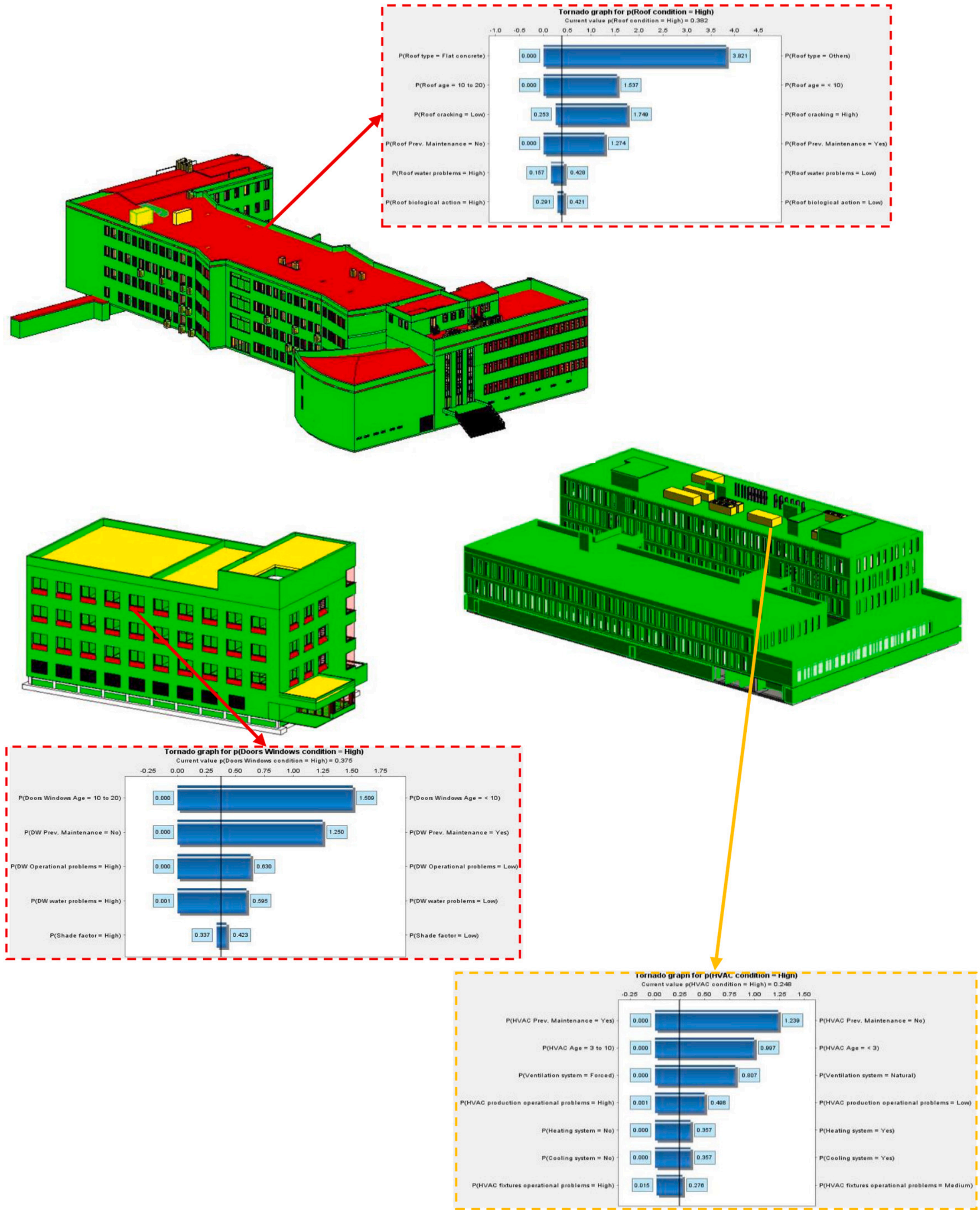


Fig. 12. The visualization of building conditions and causality analysis results.

Table 3
Task efficiency analysis.

Building	Task	Time to perform each task (minutes [hours])	
		AgenaRisk (manual)	Proposed system architecture (automated)
TR5	BDTP	1845 (30.7)	44 (0.7)
	MBB	530 (8.8)	15 (0.25)
TR11	BDTP	1390 (23.2)	40 (0.7)
	MBB	510 (8.5)	13 (0.2)
TR14	BDTP	1000 (16.6)	37 (0.6)
	MBB	495 (8.2)	10 (0.2)

perform “BDTP” and “MBB” tasks. When the system architecture was used, the same task took 0.95 h, 0.9 h and 0.8 h for each building respectively to check that all algorithms were running correctly.

Regarding data completeness, the results from a Json validator showed that the Json data (containing BIM data) formatting was correct and there was no incorrect Json syntax or missing data. Once the Json data formatting was found to be correct, the value of the transferred data was checked to meet the data completeness criterion. To achieve this, the Python script was used for all 54 scenarios. The results demonstrated that there were 1728 data items, all of which had a specific value, explaining that data from the BIM model were transferred to the BN model completely without losing data.

With regard to the transfer of the assessment results of a building's condition to the BIM model, the same approach was applied. Firstly, the Json data formatting was checked for the Json file containing the results of the building condition assessments, which were correct. Secondly, the Python script revealed that none of the values were empty (null). Besides, if a value is missing, the Dynamo will report an error while it runs for the visualization. Hence, it was concluded that the process of data transfer from the BN model to the BIM model was also performed properly and showed no data loss.

The visualization of the condition of the elements and systems from the campus buildings facilitated prioritization of investments in buildings. In the case of the three campus buildings, the roof from TR5 was found to require renovation and thus prioritization. Furthermore, the BN results allowed an evaluation of the causal factors of the condition of the elements, using sensitivity analysis. The length of the bars is a measure of the influence of parameters on the building condition assessment. Therefore, the FM team can evaluate the most probable cause of those building elements/systems associated with poor condition to implement corrective actions and plan future preventive measures.

From a sensitivity analysis of the TR5 roof, cracks in the tiles due to age and a lack of preventive actions were found to be the main causes of this poor condition. Substituting tiles, painting them with a waterproof coating to avoid efflorescence and sealing them were found to be appropriate corrective actions for the poor roof condition, while periodic inspections of roof tiles (cracked or chipped tiles) and replacement when necessary were implemented as preventive maintenance actions.

6.2. *Software verification*

Based on the case study results, the technical efforts were evaluated using the six criteria developed by Tang et al. [71]. (1) Degree of automation: the process of integrating BIM and BN is semi-automated. The first step of parameter creation, assigning the type of parameters and their families, was done manually, while the rest of the process was automated. (2) Required input & output assumptions: the required inputs are stored in the BIM model. These can be semantic information that is necessary for the BN model to run and assess building condition. (3) Computational complexity: the computational complexity of the model is low and the process can be executed on standard performance computers. The model is highly adaptable and requires only some modification if the size of the data or the number of element types increases. (4) Extensibility to new environments: the data model can be applied to different types of environments and is not specific to one or a certain class of spaces. (5) Learning capabilities: although the BN model can learn from data to improve its performance when it develops, there is no learning element in it. (6) Uncertainty modelling: one of the advantages of integrating the BIM and BN model is that the BN model can deal with uncertainty. Due to the wide range of elements that may impact a building's condition, there are varying degrees of uncertainty. As a result, evaluating the performance of a building's condition requires the examination of many factors in the presence of uncertainty.

7. Conclusions

The data model allows interoperability between the BIM and BN model to evaluate building elements and systems. The proposed system architecture automatizes the data workflows to increase the use efficiency of the BN model, reducing the time and effort that the FM team spends on manual input. Enabling interoperability between BIM and the BN model allows transformation of the data into an appropriate format automatically to run the BN model. Automating data transfer enables the FM team to take advantage of the BN model in favor. Thus, the FM team could use the proposed system architecture to prioritize the work order to improve maintenance activities and support decision-making, extend the lifespan of building elements or systems and increase building durability. Besides, it enables the FM team to address the challenges of information reliability, interoperability, usability and minimization of labor time. The data model can be applied to any building typology and is very relevant because its application allows the assessment of building conditions in a semi-automated way.

The method of visualization in this approach focuses on the condition of building elements and systems, which is demonstrated on a color scale where red indicates urgency in building elements and systems intervention, yellow indicates deteriorating performance condition, and green indicates satisfactory condition of the building elements and systems. This visualization makes it possible to detect the condition of current building elements and systems more intuitively, and potentially makes it easier to deal with the problem. This will result in a considerable improvement in building performance. Overall, the workflow for the FM team to use the system architecture is:

- Run the system architecture in all buildings managed by the FM company
- Visualize the condition of the building elements/systems for those buildings
- Check sensitivity analysis to determine the most probable causes for the building elements/systems with low-performance condition
- Make corrective action plans
- Propose preventive maintenance plans

There are some limitations: (1) the condition risk assessment model based on BN was developed to evaluate and prioritize building renovations managed by big facility management companies. The approach of this condition risk assessment considers different elements (façade, interior partitions, etc.) and systems (plumbing, HVAC, etc.) in each building as entities. Therefore, the data model to integrate the BIM and the BN was based on these assumptions. Further analysis can adapt this condition risk assessment model to other functionalities. (2) The methodology is semi-automated. Hence, an end-user is recommended between each step of extraction to ensure that the exported files are stored in the right location with the correct names. For instance, the Python scripts used in the BIM model can only read the exported file (e.g., BN results), which is matched with its name and location.

Future work could extend the model further using industry 4.0 technologies, sensor-based systems, AI, IoT, BIM, and other technologies together to create a fully integrated and automated solution. It could also move further still towards dark factory environments where a building is controlled by robots without any human intervention.

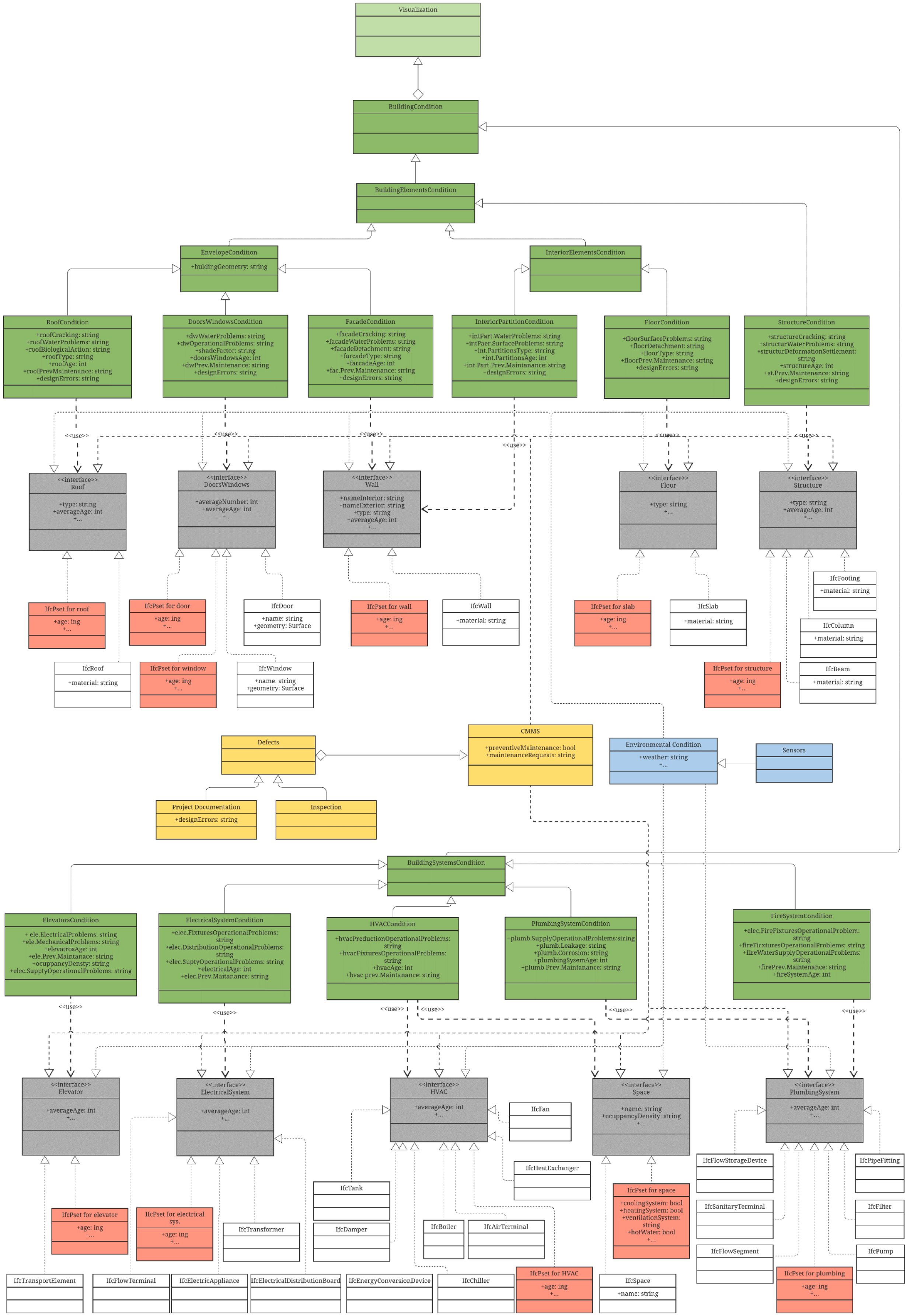
Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) from Generalitat de Catalunya under Grant 2019 FI_B00064.

Appendix A. Appendix



Visualization
BuildingCondition
BuildingElementsCondition
EnvelopeCondition
+buldingGeometry: string
InteriorElementsCondition
RoofCondition
+roofCracking: string
+roofWaterProblems: string
+roofBiologicalAction: string
+roofType: string
+roofAge: int
+roofPrevMaintenance: string
+designErrors: string
DoorsWindowsCondition
+dwWaterProblems: string
+dwOperationalProblems: string
+shadeFactor: string
+doorsWindowsAge: int
+dwPrev.Maintenance: string
+designErrors: string
FacadeCondition
+facadeCracking: string
+facadeWaterProblems: string
+facadeDetachment: string
+farcadeType: string
+farcadeAge: int
+fac.Prev.Maintenance: string
+designErrors: string
InteriorPartitionCondition
+intPart.WaterProblems: string
+intPaer.SurfaceProblems: string
+int.PartitionsType: string
+int.PartitionsAge: int
+int.Part.Prev.Maintanance: string
+designErrors: string
FloorCondition
+floorSurfaceProblems: string
+floorDetachment: string
+floorType: string
+floorPrev.Maintenance: string
+designErrors: string
StructureCondition
+structureCracking: string
+structurWaterProblems: string
+structurDeformationSettlement: string
+structureAge: int
+st.Prev.Maintenance: string
+designErrors: string
<<use>>
<<interface>> Roof
+type: string
+averageAge: int
+...
<<interface>> DoorsWindows
+averageNumber: int
+averageAge: int
+...
<<interface>> Wall
+nameInterior: string
+nameExterior: string
+type: string
+averageAge: int
+...
<<interface>> Floor
+type: string
+...
<<interface>> Structure
+type: string
+averageAge: int
+...
IfcPset for roof
+age: ing
+...
IfcPset for door
+age: ing
+...
IfcDoor
+name: string
+geometry: Surface
IfcPset for wall
+age: ing
+...
IfcWall
+material: string
IfcPset for slab
+age: ing
+...
IfcSlab
+material: string
IfcFooting
+material: string
IfcColumn
+material: string
IfcBeam
+material: string
IfcRoof
+material: string
IfcPset for window
+age: ing
+...
IfcWindow
+name: string
+geometry: Surface
IfcPset for structure
+age: ing
+...
Defects
CMMS
+preventiveMaintenance: bool
+maintenanceRequests: string
Environmental Condition
+weather: string
+...
Sensors
Project Documentation
+designErrors: string
Inspection
BuildingSystemsCondition
ElevatorsCondition
+ ele.ElectricalProblems: string
+ele.MechanicalProblems: string
+elevatrosAge: int
+ele.Prev.Maintanance: string
+ocuppancyDensity: string
+elec.SupptyOperationalProblems: string
ElectricalSystemCondition
+elec.FixturesOperationalProblems: string
+elec.DistributionOperationalProblems: string
+elec.SuptyOperationalProblems: string
+electricalAge: int
+elec.Prev.Maitanance: string
HVACCondition
+hvacPreductionOperationalProblems: string
+hvacFixturesOperationalProblems: string
+hvacAge: int
+hvac prev.Maintanance: string
PlumbingSystemCondition
+plumb.SupplyOperationalProblems:string
+plumb.Leakage: string
+plumb.Corrosion: string
+plumbingSysemAge: int
+plumb.Prev.Maintenance: string
FireSystemCondition
+elec.FireFixturesOperationalProblem: string
+fireFicxturesOperationalProblems: string
+fireWaterSupplyOperationalProblems: string
+firePrev.Maintenance: string
+fireSystemAge: int
<<interface>> Elevator
+averageAge: int
+...
<<interface>> ElectricalSystem
+averageAge: int
+...
<<interface>> HVAC
+averageAge: int
+...
IfcFan
IfcHeatExchanger
<<interface>> Space
+name: string
+ocuppancyDensity: string
+...
<<interface>> PlumbingSystem
+averageAge: int
+...
IfcFlowStorageDevice
IfcPipeFitting
IfcSanitaryTerminal
IfcFilter
IfcFlowSegment
IfcPump
IfcPset for elevator
+age: ing
+...
IfcPset for electrical sys.
+age: ing
+...
IfcTransformer
IfcTank
IfcDamper
IfcBoiler
IfcAirTerminal
IfcPset for space
+coolingSystem: bool
+heatingSystem: bool
+ventilationSystem: string
+hotWater: bool
+...
IfcTransportElement
IfcFlowTerminal
IfcElectricAppliance
IfcElectricalDistributionBoard
IfcEnergyConversionDevice
IfcChiller
IfcPset for HVAC
+age: ing
+...
IfcSpace
+name: string
IfcPset for plumbing
+age: ing
+...

References

[1] R. Sacks, C. Eastman, G. Lee, P. Teicholz, BIM Handbook, John Wiley & Sons, Inc, Hoboken, New Jersey, 2018, https://doi.org/10.1002/9781119287568 (ISBN: 9781119287568).
[2] W. Chen, K. Chen, J.C.P. Cheng, Q. Wang, V.J.L. Gan, BIM-based framework for automatic scheduling of facility maintenance work orders, Autom. Constr. 91 (2018) 15–30, https://doi.org/10.1016/j.autcon.2018.03.007.
[3] J.K.W. Wong, J. Ge, S.X. He, Digitisation in facilities management: a literature review and future research directions, Autom. Constr. 92 (2018) 312–326, https://doi.org/10.1016/j.autcon.2018.04.006.
[4] H.B. Gunay, W. Shen, C. Yang, Text-mining building maintenance work orders for component fault frequency, Build. Res. Inf. 47 (2019) 518–533, https://doi.org/10.1080/09613218.2018.1459004.
[5] N.A. Garyaev, F. Ayoub, Towards building information modelling for diagnosis, assessment and rehabilitation automation for existing buildings, in: Journal of Physics: Conference Series, Institute of Physics Publishing, 2020, p. 12121, https://doi.org/10.1088/1742-6596/1425/1/012121.
[6] R. Bortolini, N. Forcada, Analysis of building maintenance requests using a text mining approach: building services evaluation, Build. Res. Inf. 48 (2020) 207–217, https://doi.org/10.1080/09613218.2019.1609291.
[7] I. Flores-Colen, J. De Brito, A systematic approach for maintenance budgeting of buildings facades based on predictive and preventive strategies, Constr. Build. Mater. 24 (2010) 1718–1729, https://doi.org/10.1016/j.conbuildmat.2010.02.017.
[8] S. Yacob, A.S. Ali, A.Y.C. Peng, Building condition assessment: Lesson learnt from pilot projects, in: MATEC Web of Conferences, EDP Sciences, 2016, p. 00072, https://doi.org/10.1051/matecconf/20166600072.
[9] L.G. Neuberg, in: Judea Pearl (Ed.), Causality: Models, Reasoning, and Inference 19, Cambridge University Press, 2003, pp. 675–685, https://doi.org/10.1017/s0266466603004109, 2000, Econometric Theory.
[10] L.D. Nguyen, D.Q. Tran, M.P. Chandrawinata, Predicting safety risk of working at heights using Bayesian networks, J. Constr. Eng. Manag. 142 (2016) 04016041, https://doi.org/10.1061/(ASCE)CO.1943-7862.0001154.
[11] S. Lee, I. Bilionis, P. Karava, A. Tzempelikos, A Bayesian approach for probabilistic classification and inference of occupant thermal preferences in office buildings, Build. Environ. 118 (2017) 323–343, https://doi.org/10.1016/j.buildenv.2017.03.009.
[12] J. Langevin, J. Wen, P.L. Gurian, Modeling thermal comfort holistically: Bayesian estimation of thermal sensation, acceptability, and preference distributions for office building occupants, Build. Environ. 69 (2013) 206–226, https://doi.org/10.1016/j.buildenv.2013.07.017.
[13] R. Bortolini, N. Forcada, A probabilistic performance evaluation for buildings and constructed assets, Build. Res. Inf. 0 (2019) 1–18, https://doi.org/10.1080/09613218.2019.1704208.
[14] C.J. Roberts, D.J. Edwards, M.R. Hosseini, M. Mateo-Garcia, D.G. Owusu-Manu, Post-occupancy evaluation: a review of literature, Eng. Constr. Archit. Manag. 26 (2019) 2084–2106, https://doi.org/10.1108/ECAM-09-2018-0390.
[15] H.B. Cavka, S. Staub-French, E.A. Poirier, Developing owner information requirements for BIM-enabled project delivery and asset management, Autom. Constr. 83 (2017) 169–183, https://doi.org/10.1016/j.autcon.2017.08.006.
[16] H. Alavi, N. Forcada, R. Bortolini, D.J. Edwards, Enhancing occupants' comfort through BIM-based probabilistic approach, Autom. Constr. 123 (2021), 103528, https://doi.org/10.1016/j.autcon.2020.103528.
[17] A. Carbonari, A. Corneli, G.M. Di Giuda, L. Ridolfi, V. Villa, A decision support system for multi-criteria assessment of large building stocks, J. Civ. Eng. Manag. 25 (2019) 477–494, https://doi.org/10.3846/jcem.2019.9872.
[18] A. Carbonari, A. Giretti, A. Corneli, V. Villa, G. Di Giuda, Decision support tool for multi-criteria analyses of the quality of large building stock, in: ISARC 2017 - Proceedings of the 34th International Symposium on Automation and Robotics in Construction, International Association for Automation and Robotics in Construction I.A.A.R.C, 2017, pp. 22–29, https://doi.org/10.22260/isarc2017/0003.
[19] X. Gao, P. Pishdad-Bozorgi, BIM-enabled facilities operation and maintenance: a review, Adv. Eng. Inform. 39 (2019) 227–247, https://doi.org/10.1016/j.aei.2019.01.005.
[20] S.H. Alavi, N. Forcada, BIM LOD for facility management tasks, in: Proceedings of the 2019 European Conference for Computing in Construction, University College Dublin, 2019, pp. 154–163, https://doi.org/10.35490/ec3.2019.187.
[21] A. Corneli, S. Meschini, V. Villa, G.M. Di Giuda, A. Carbonari, A decision support system for the multicriteria analysis of existing stock, in: Procedia Engineering, Elsevier Ltd, 2017, pp. 682–689, https://doi.org/10.1016/j.proeng.2017.08.058.
[22] R. Matos, F. Rodrigues, H. Rodrigues, A. Costa, Building condition assessment supported by building information modelling, J. Build. Eng. 38 (2021), 102186, https://doi.org/10.1016/j.jobe.2021.102186.
[23] F. Auffenberg, S. Snow, S. Stein, A. Rogers, A comfort-based approach to smart heating and air conditioning, ACM Trans. Intell. Syst. Technol. 9 (2017) 1–20, https://doi.org/10.1145/3057730.
[24] Y. Zhao, J. Wen, F. Xiao, X. Yang, S. Wang, Diagnostic Bayesian networks for diagnosing air handling units faults – part I: faults in dampers, fans, filters and sensors, Appl. Therm. Eng. 111 (2017) 1272–1286, https://doi.org/10.1016/j.applthermaleng.2015.09.121.
[25] R. Jang, W. Collinge, Improving BIM asset and facilities management processes: a Mechanical and Electrical (M&E) contractor perspective, J. Build. Eng. 32 (2020) 101540, https://doi.org/10.1016/j.jobe.2020.101540.
[26] B. Dong, Z. O'Neill, Z. Li, A BIM-enabled information infrastructure for building energy fault detection and diagnostics, Autom. Constr. 44 (2014) 197–211, https://doi.org/10.1016/j.autcon.2014.04.007.
[27] Ö. Göçer, Y. Hua, K. Göçer, A BIM-GIS integrated pre-retrofit model for building data mapping, Build. Simul. 9 (2016) 513–527, https://doi.org/10.1007/s12273-016-0293-4.
[28] J. Zhou, P.E.D. Love, J. Matthews, B. Carey, C.P. Sing, Object-oriented model for life cycle management of electrical instrumentation control projects, Autom. Constr. 49 (2015) 142–151, https://doi.org/10.1016/j.autcon.2014.10.008.
[29] R. Marmo, F. Polverino, M. Nicolella, A. Tibaut, Building performance and maintenance information model based on IFC schema, Autom. Constr. 118 (2020), 103275, https://doi.org/10.1016/j.autcon.2020.103275.
[30] BuildingSMART Team, Model View Definitions (MVD) - buildingSMART International. https://www.buildingsmartusa.org/standards/bsi-standards/model-view-definitions-mvd/, 2020 (accessed November 22, 2021).
[31] E. William East, N. Nisbet, T. Liebich, Facility management handover model view, J. Comput. Civ. Eng. 27 (2013) 61–67, https://doi.org/10.1061/(asce)cp.1943-5487.0000196.
[32] W. Thabet, J. Lucas, S. Johnston, A case study for improving BIM-FM handover for a large educational institution, in: Construction Research Congress 2016: Old and New Construction Technologies Converge in Historic San Juan - Proceedings of the 2016 Construction Research Congress, CRC 2016, 2016, pp. 2177–2186, https://doi.org/10.1061/9780784479827.217.
[33] P. Dias, S. Ergan, The need for representing facility information with customized LOD for specific FM tasks, in: Construction Research Congress 2016: Old and New Construction Technologies Converge in Historic San Juan - Proceedings of the 2016 Construction Research Congress, CRC 2016, 2016, pp. 2563–2572, https://doi.org/10.1061/9780784479827.255.
[34] B. Becerik-Gerber, F. Jazizadeh, N. Li, G. Calis, Application areas and data requirements for BIM-enabled facilities management, J. Constr. Eng. Manag. 138 (2012) 431–442, https://doi.org/10.1061/(ASCE)CO.1943-7862.0000433.
[35] K. Kim, H. Kim, W. Kim, C. Kim, J. Kim, J. Yu, Integration of ifc objects and facility management work information using Semantic Web, Autom. Constr. 87 (2018) 173–187, https://doi.org/10.1016/j.autcon.2017.12.019.
[36] A. Gouda Mohamed, M.R. Abdallah, M. Marzouk, BIM and semantic web-based maintenance information for existing buildings, Autom. Constr. 116 (2020), 103209, https://doi.org/10.1016/j.autcon.2020.103209.
[37] J.C.P. Cheng, W. Chen, K. Chen, Q. Wang, Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms, Autom. Constr. 112 (2020), 103087, https://doi.org/10.1016/j.autcon.2020.103087.
[38] W. Shen, Q. Hao, Y. Xue, A loosely coupled system integration approach for decision support in facility management and maintenance, Autom. Constr. 25 (2012) 41–48, https://doi.org/10.1016/j.autcon.2012.04.003.
[39] A. Motamedi, A. Hammad, Y. Asen, Knowledge-assisted BIM-based visual analytics for failure root cause detection in facilities management, Autom. Constr. 43 (2014) 73–83, https://doi.org/10.1016/j.autcon.2014.03.012.
[40] A. Golabchi, M. Akula, V. Kamat, Automated building information modeling for fault detection and diagnostics in commercial HVAC systems, Facilities 34 (2016) 233–246, https://doi.org/10.1108/F-06-2014-0050.
[41] F. Shalabi, Y. Turkan, IFC BIM-based facility management approach to optimize data collection for corrective maintenance, J. Perform. Constr. Facil. 31 (2017) 04016081, https://doi.org/10.1061/(asce)cf.1943-5509.0000941.
[42] A. Papapostolou, H. Chaouchi, RFID-assisted indoor localization and the impact of interference on its performance, J. Netw. Comput. Appl. 34 (2011) 902–913, https://doi.org/10.1016/j.jnca.2010.04.009.
[43] S.H. Wang, W.C. Wang, K.C. Wang, S.Y. Shih, Applying building information modeling to support fire safety management, Autom. Constr. 59 (2015) 158–167, https://doi.org/10.1016/j.autcon.2015.02.001.
[44] M.Y. Cheng, K.C. Chiu, Y.M. Hsieh, I.T. Yang, J.S. Chou, Y.W. Wu, BIM integrated smart monitoring technique for building fire prevention and disaster relief, Autom. Constr. 84 (2017) 14–30, https://doi.org/10.1016/j.autcon.2017.08.027.
[45] Y.J. Chen, Y.S. Lai, Y.H. Lin, BIM-based augmented reality inspection and maintenance of fire safety equipment, Autom. Constr. 110 (2020), 103041, https://doi.org/10.1016/j.autcon.2019.103041.
[46] G. Zimmermann, Y. Lu, G. Lo, Automatic HVAC fault detection and diagnosis system generation based on heat flow models, in: HVAC and R Research, 2012, pp. 112–125, https://doi.org/10.1080/10789669.2011.610427.
[47] X. Yang, S. Ergan, Leveraging BIM to provide automated support for efficient troubleshooting of HVAC-related problems, J. Comput. Civ. Eng. 30 (2016) 04015023, https://doi.org/10.1061/(asce)cp.1943-5487.0000492.
[48] H. Wang, Y. Pan, X. Luo, Integration of BIM and GIS in sustainable built environment: a review and bibliometric analysis, Autom. Constr. 103 (2019) 41–52, https://doi.org/10.1016/j.autcon.2019.03.005.
[49] J.J. McArthur, A building information management (BIM) framework and supporting case study for existing building operations, maintenance and sustainability, Proc. Eng. 118 (2015) 1104–1111, https://doi.org/10.1016/j.proeng.2015.08.450.
[50] T. Gerrish, K. Ruikar, M. Cook, M. Johnson, M. Phillip, C. Lowry, BIM application to building energy performance visualisation and management challenges and potential, Energ. Build. 144 (2017) 218–228, https://doi.org/10.1016/j.enbuild.2017.03.032.
[51] A. Galiano-Garrigós, A. García-Figueroa, C. Rizo-Maestre, Á. González-Avilés, Evaluation of BIM energy performance and CO2 emissions assessment tools: a case study in warm weather, Build. Res. Inf. 47 (2019) 787–812, https://doi.org/10.1080/09613218.2019.1620093.

[52] T. Gerrish, K. Ruikar, M. Cook, M. Johnson, M. Phillip, Using BIM capabilities to improve existing building energy modelling practices, Eng. Constr. Archit. Manag. 24 (2017) 190–208, https://doi.org/10.1108/ECAM-11-2015-0181.

[53] S. Matarneh, M. Danso-Amoako, S. Al-Bizri, M. Gaterell, R. Matarneh, BIM-based facilities information: streamlining the information exchange process, J. Eng. Des. Technol. 17 (2019) 1304–1322, https://doi.org/10.1108/JEDT-02-2019-0048.

[54] G.B. Ozturk, Interoperability in building information modeling for AECO/FM industry, Autom. Constr. 113 (2020), 103122, https://doi.org/10.1016/j.autcon.2020.103122.

[55] R. Volk, J. Stengel, F. Schultmann, Building Information Modeling (BIM) for existing buildings - Literature review and future needs, Autom. Constr. 38 (2014) 109–127, https://doi.org/10.1016/j.autcon.2013.10.023.

[56] T.W. Kang, C.H. Hong, A study on software architecture for effective BIM/GIS-based facility management data integration, Autom. Constr. 54 (2015) 25–38, https://doi.org/10.1016/j.autcon.2015.03.019.

[57] I. Motawa, A. Almarshad, A knowledge-based BIM system for building maintenance, Autom. Constr. 29 (2013) 173–182, https://doi.org/10.1016/j.autcon.2012.09.008.

[58] J. Lee, Y. Jeong, Y.S. Oh, J.C. Lee, N. Ahn, J. Lee, S.H. Yoon, An integrated approach to intelligent urban facilities management for real-time emergency response, Autom. Constr. 30 (2013) 256–264, https://doi.org/10.1016/j.autcon.2012.11.008.

[59] C. Quinn, A.Z. Shabestari, T. Misic, S. Gilani, M. Litoiu, J.J. McArthur, Building automation system - BIM integration using a linked data structure, Autom. Constr. 118 (2020), 103257, https://doi.org/10.1016/j.autcon.2020.103257.

[60] A.I.C. Ani, S. Johar, N.M. Tawil, M.Z.A. Razak, N. Hamzah, Building information modeling (BIM)-based building condition assessment: a survey of water ponding defect on a flat roof, J. Teknol. 75 (2015) 25–31, https://doi.org/10.11113/jt.v75.5222.

[61] H. Tashakkori, A. Rajabifard, M. Kalantari, A new 3D indoor/outdoor spatial model for indoor emergency response facilitation, Build. Environ. 89 (2015) 170–182, https://doi.org/10.1016/j.buildenv.2015.02.036.

[62] A.H. Oti, E. Kurul, F. Cheung, J.H.M. Tah, A framework for the utilization of building management system data in building information models for building design and operation, Autom. Constr. 72 (2016) 195–210, https://doi.org/10.1016/J.AUTCON.2016.08.043.

[63] H. Alavi, N. Forcada, S.-L. Fan, W. San, BIM-based augmented reality for facility maintenance management, in: Proceedings of the 2021 European Conference for Computing in Construction, University College Dublin, 2021, pp. 431–438, https://doi.org/10.35490/EC3.2021.180.

[64] N. Fenton, M. Neil, Risk Assessment and Decision Analysis with Bayesian Networks, Chapman and Hall/CRC, Boca Raton, Florida: CRC Press, 2018, https://doi.org/10.1201/b21982 (ISBN: 9781315269405).

[65] E. Pérez-Miñana, Improving ecosystem services modelling: insights from a Bayesian network tools review, Environ. Model. Softw. 85 (2016) 184–201, https://doi.org/10.1016/j.envsoft.2016.07.007.

[66] T. Weilkiens, Systems Engineering with SysML/UML, Elsevier, 2007, https://doi.org/10.1016/B978-0-12-374274-2.X0001-6 (ISBN: 9780123742742).

[67] BuildingSMART, Industry Foundation Classes (IFC) - buildingSMART International, Building Smart. https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/, 2020 (accessed September 14, 2021).

[68] K. Afsari, C.M. Eastman, D. Castro-Lacouture, JavaScript Object Notation (JSON) data serialization for IFC schema in web-based BIM data exchange, Autom. Constr. 77 (2017) 24–51, https://doi.org/10.1016/j.autcon.2017.01.011.

[69] M. Wickham, Introduction to JSON, in: Practical Android, Apress, Berkeley, CA, 2018, pp. 1–15, https://doi.org/10.1007/978-1-4842-3333-7_1 (ISBN: 978-1-4842-3332-0).

[70] Spyder, SPYDER IDE, Spyder Project. https://www.spyder-ide.org/, 2018 (accessed November 19, 2021).

[71] P. Tang, D. Huber, B. Akinci, R. Lipman, A. Lytle, Automatic reconstruction of as-built building information models from laser-scanned point clouds: a review of related techniques, Autom. Constr. 19 (2010) 829–843, https://doi.org/10.1016/j.autcon.2010.06.007.