Journal of Building Engineering 65 (2023) 105730

Contents lists available at ScienceDirect

Journal of Building Engineering

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

BIM-based method to inform operation and maintenance phases through a simplified procedure

Massimiliano Condotta, Chiara Scanagatta *

Università Iuav di Venezia, Santa Croce 191, 30135, Venezia, Italy

A R T I C L E I N F O

Keywords:
BIM
CMMS
Facility management
IFC
Interoperability
Operation and maintenance

A B S T R A C T

The use of BIM in the architecture, engineering, construction and operation (AECO) industry is becoming increasingly widespread. Nevertheless, interoperability with Computerized Maintenance Management Systems (CMMS) remains a significant challenge in the domain. Furthermore, available applications require high expertise to correctly implement the data export, don't allow incremental database updates and are overall non-cost-effective, moreover in the case of small and medium-sized structures. At the same time, given the need to invest in a more sustainable built environment, it is necessary to consider the whole lifecycle of constructions, including the operation and maintenance phase, while designing. In the framework of this scenario, the paper presents a research work which defines a method to implement interoperability between BIM and CMMS software by using the IFC open standard, to allow an easier management of buildings, from design to decommissioning. This method, devised to be as feasible as possible, combines an operational procedure and a practical tool. The work was supported by the software house Prometeo Srl and the Facility Management company GEMMO SpA. The method was developed based on the Feltre Hospital (Italy) and results, that were technically validated through an ad hoc BIM model, can be used by other practitioners to further implement the method in other research.

1. Introduction

BIM systems – defined as 'shared digital representation of a built asset to facilitate design, construction and operation processes to form a reliable basis for decisions' [1] - are becoming increasingly popular, both as an approach and as a tool, to be applied and utilized in the design phases of the building process. The use of BIM systems allows stakeholders to reach an effective and direct interaction between several disciplines and specialist skills, in particular within architectural design, structural design, and plant design. This allows for an integrated design approach where different fields contribute to defining all the design aspects organically - from the compositional ones to the technological and engineering ones.

The BIM project, *informed* by various design components (e.g., structural model, HVAC systems, architectural model), is increasingly approaching the concept of 'digital twin' towards a complete virtual representation of a real building which comprises its spatial aspects and the physical objects within [2]. In such a scenario it becomes logical to take advantage of BIM in the construction phase to create a single digital project which works as a hub for all the information necessary for the erection of a building. For the same reason, the design and construction management phases are increasingly taking advantage of the potential offered by BIM tools [3–6].

In recent years, however, in an environmentally conscious design perspective the concept of sustainable design is paying increasing

* Corresponding author.
E-mail address: cscanagatta@iuav.it (C. Scanagatta).

https://doi.org/10.1016/j.jobe.2022.105730
Received 4 August 2022; Received in revised form 5 December 2022; Accepted 10 December 2022
Available online 12 December 2022

attention to the building's life cycle [7–9]. The project must therefore evaluate both the usage phase (considered as the period during which energy is used and maintenance is needed, and therefore it is a time of renewal of materials and components) and the decommissioning phase, during which, besides energy consumption, there is 'waste' of materials. Therefore, designing a sustainable building can no longer disregard the planning of its entire life cycle.

In this new scenario, where the design stage and the operation and maintenance phase are no longer separate but are interconnected, BIM systems can play a relevant role. A project developed in BIM can be used in the construction phase and can be updated in real time with any variation with respect to the original project, despite it currently being a big challenge. Subsequently, the up-to-date BIM model (As-Built model), which is supposed to be validated by professionals who participated in the work, can be used in the operation and maintenance phase. This can also include the decommissioning of the building: the information held and updated in the digital model during the various maintenance activities can supply all the necessary inputs for the pre-demolition audit [10].

Nevertheless, this process of using the BIM model also in the operation and maintenance phase presents difficulties when implemented in a real environment [11,12]. The exchange of information between the design, construction, and operation and maintenance phases is not always easy to manage, especially when different competencies and specialized software are involved [13–15]. Thus, it is essential to think in terms of interoperability between applications, where interoperability is to be read as 'the ability to manage and communicate electronic product and project data between collaborating firms and within individual companies' design, construction, maintenance, and business process systems' (NIST, 2004) and not only in term of software.

The operation and maintenance phase nowadays is characterized, in terms of digital tools, by the use of CMMS (Computerized Maintenance Management System) software. Their role is to create and maintain an up-to-date database of information for the buildings' maintenance operations. To date, the information contained within the CMMS software is then used by the maintenance staff to conduct their work more effectively and efficiently while allowing them to make informed decisions about their tasks. A further function of CMMS software is to facilitate the verification of the compliance of installed plants with both internal technological update requests and current regulations, enabling timely intervention in the event of modifications or updates to standards. Some of the main uses of CMMS software are: creation of a log of the tasks already conducted, as well as active ones and those planned for routine maintenance; tracking of the jobs completed and assigned to each maintenance man, to optimize the management of available resources; supplies storage management.

Unfortunately, as Shalabi and Turkan [16] state 'current Facility Management systems are lacking interoperability capabilities and are operated by different teams resulting in poor data coordination and management'. Thus, it is necessary to find ways of implementing a structured and functional tool that allows such missing interoperability.

Considering what has been said above, the difficulties of including the operation and maintenance phase within a single and interoperable process, inherent to the entire life cycle of the building and based on BIM models, have a double origin. Firstly, a procedural one, that is linked to the existing practice of considering the operation and maintenance phase isolated and independent from the design phase. Secondly, a technical one, due to the difficulty of using the information that can be found in the BIM model within the CMMS software since they use different standards in terms of metadata and of logic of definition of the building structure. Until these obstacles are overcome, they will function as a hinder for the spread of procedures based on interoperability and integrated use of BIM systems within operational and maintenance processes.

The work presented in this paper raises from the request of Facility Managers to have a simple, but at the same time, effective method to use BIM data to populate the CMMS database, allowing access to such useful information for the operation and maintenance phase. Existing standard procedures, on the contrary, require too much effort compared to the investment when applied to modest-sized structures, resulting in a non-cost-effective process. The hereby presented work was developed from specific requests and a real case: the Feltre Hospital in Italy. Validation was undertaken using a BIM model specially designed to test some system characteristics.

Building on the experience of the two companies that participated in the project – the software house Prometeo Srl and the Facility Management company GEMMO SpA, it emerged that the data and linked information of the BIM model that are nowadays used in most operation and maintenance activities are limited in number. In fact, most of the *sites*' maintenance information derived from BIM are mainly aimed at the definition of the building spatial environments (e.g., rooms, floors) according to a hierarchical order. Classification and mapping of all the information related to equipment, systems, and technical elements of the building are then usually delegated to field survey operations. Even if a BIM model is used to also obtain equipment data, the activity of field survey, in jargon called 'census', is not a phase that can be renounced: relying only on data from the project model is not a viable way to have absolute certainty of the situation. In fact, almost always there are differences, especially at the level of plants and technical elements, between the project documents and the completed building. Although the practice of As Built has decreased this divergence, field verification is always necessary as Angers [17] and Adams [18] already stated.

It is however true that, in a process based on the automatic import of buildings' data (spatial and technical), the census could be configured as a 'verification' of the data coming from the BIM. Nowadays the census involves two activities: the first one being the import of the data into BIM, and the second one involves their verification; with a more coordinated and maintenance-oriented system that presents a complete BIM model and selects and exports only desired data, it would be necessary to carry out only the second of these activities. It is, therefore, necessary to define a simplified procedure that meets the needs of users, so that BIM models can become a real resource and opportunity. This is not the situation today. BIM presents a complex set of information that is difficult to manage and is not strictly related to Facility Management. As Carnero and Novés [19] pointed out, these difficulties occur mainly in the management of small and medium-sized structures with limited maintenance budgets and, as the involved company confirmed, this statement is still true today. In these cases, overly complex BIM and CMMS interface software systems discourage their application and therefore 'traditional' approaches are used without integration of the two phases - construction and maintenance - with the functionalities of the different systems.

In this context, the research purpose was therefore to develop an 'integrated method', to improve the effectiveness and usability of BIM standards in operational and maintenance phases, based on:

- A simplified 'operational procedure' that improves interoperability between BIM and CMMS software by also exploiting a 'taxonomy' that defines both the spatial aspects of a building (i.e., building site, building, floor, room) and the objects and elements that need maintenance and that are comprised in it (e.g., electrical plug, lamp, window);
- A practical tool based on an application programming interface (API) to automatically import needed BIM information into CMMS systems.

2. State of the art and challenges

2.1. Standards and interoperability

In analyzing the state of the art, the first issue addressed was linked to the use of different BIM software in the design phase and how interoperability between them is managed. This analysis aimed at understanding which metadata standards are used and how these are organized, to find the most suitable one to transfer information directly to CMMS software databases [20–23].

About this topic Sampaio and Gomes [24], in their analysis of interoperability between different BIM software, studied how native transfer is possible with programs from the same developer, while the other possible transfer option between BIM software is the use of the IFC open data schema, that today is a consolidated standard. Furthermore, they state that there is a level of inadequacy when combining models from different sectors - e.g., from structural calculations to architectural modeling. In any case, IFC is the only common language for BIM software but, since there are different IFC releases, there can be a compatibility issue based on the type of IFC used [25]. One example was given by Trzeciak and Borrmann [26]. They analyzed how AllPlan does not support Revit imports of 'IFC 4', while 'IFC 2 × 3' imports can be used. On the other hand, AllPlan supports both 'IFC 4' and 'IFC 2 × 3' as exporting formats, and Revit recognizes both when importing. Nevertheless, such differences are related to the model in the BIM environment, and not to the IFC data structure. Since the incongruity lies within the 3D model, it means that it is possible to use IFC as the base for full interoperability within the complete lifecycle of buildings. Thus, from the literature emerges that IFC is the right type of metadata to be used for interoperability [27,28].

In spite of this, 'not all the relevant standard-related aspects are always respected as they should be [and given that] this results in heterogeneous IFC solutions for similar situations which significantly hinders interoperability' [29]. Being aware of this possible interoperability issue it can be deduced that it is necessary to find a correspondence of data between IFC and CMMS databases based on minimum requirements.

2.2. BIM to CMMS: COBie

The most common format for data transfer from BIM to CMMS nowadays is the COBie standard. This standard was developed by East [30] to improve how information is captured during the design and construction phases to inform operation and maintenance, and to allow the elimination of the paper documentation needed by Facility Managers following the completion of a project. Since COBie works within the framework of BIM systems, for its development the Model View Definition (MVD) approach - which formally defines a subset of the IFC entities and attributes that are needed to satisfy one or many exchange requirements in BIM systems - was considered [31]. COBie was further developed to be built upon the IFC open data schema [32] and nowadays it is a standard MVD and a subset of the IFC schema.

Nevertheless, despite being a well-established MVD standard for BIM and CMMS interoperability, in order for COBie to function at its best, it is necessary to have the BIM model enriched with all the data that will be used during the operation and maintenance phase. On the contrary, it may be necessary to supplement the existing BIM model before importing it into the CMMS software. Lavy and Jawadekar [33]. studied that when first developed COBie proved to be beneficial as a data handover tool, to avoid interoperability problems, and as a universal data repository. Later, when BIM software developers implemented the COBie plug-in (a tool that implements the COBie standard through a user-friendly user interface within most common BIM software), they adapted it for commercial needs by changing some of its characteristics to make it more immediate for users from a UI point of view. Such user interface changes resulted in a complication in the use of the COBie standard due to the request for many preference settings before the export. Indeed, to reach the expected benefits of avoiding the paper version of maintenance documentation, there is the need to start the database formulation early in the design phase. In more recent times Kumar and Ai Lin Teo [34] retested COBie to check if the benefits were still surpassing the possible disadvantages since, as they say, 'there is a paucity in published literature, which discusses in detail about the various issues associated with COBie datasheet handling'. Their study showed how COBie nowadays has some main characteristics that can make it obsolete for certain uses: (i) it has high manual data entry requirement, (ii) the COBie plug-in interface in the BIM environment is not user-friendly in its logic of filling in data and (iii) there is a lack of mechanisms to check missing information. This analysis [34] shows how, firstly, the COBie plug-in is not fully automated and the required attribute values, even when implemented in BIM, are not automatically selected by it. COBie needs to be set for it to select and export the needed values. Secondly, it is not possible to add all descriptive information (e.g., technical sheets, use manual) since the data capturing goes beyond the limits of BIM models. Such problem is not related to COBie itself, but it is related to the logic of functioning of the BIM model. In fact, populating the model with all needed operation and maintenance information would result, in many cases, in a file too heavy to work with due to hardware limitations. This heaviness would make it difficult also to export the resulting document even if the chosen format is a.csv file (COBie allows to export data in different formats but.csv is the most common). Concerning the lack of a mechanism to check missing information, it is a predominant issue since the only way to verify whether all the hierarchal information is captured or not is by doing it manually [35]. Lastly, not all data useful for the construction phase might be of use for the operation and maintenance phase but, when exporting from BIM with COBie, all of them are incorporated within the IFC file. Hence, it raises the need of finding a simplified alternative to transfer IFC in CMMS database, since the COBie export system generates an overpopulated file.

2.3. Data management of buildings' spatial aspects

There are some issues related to spatial aspects that have emerged from the BIM software user community and which are reported below.

A common issue that regards the export of a BIM model into CMMS is related to the spatial aspects of the building. CMMS databases have a hierarchical subdivision of the structures that usually involve rooms, levels, buildings, complexes of buildings, etc. Sometimes this hierarchy is variable and can be freely defined by users. In fact, such subdivision might work both with 'unassigned' levels (to be able to fit different cases e.g., an office building with multiple levels and rooms, or a warehouse that consists of an isolated building with an 'empty' space) or with 'assigned' levels that are organized based on needs. This last database management option does not guarantee a perfect match between the IFC schema and the CMMS database, resulting in the need for a further setting of the data after the BIM data export. Indeed, the absence of a 1-1 mapping is not an IFC lack, but it is a general interoperability issue that needs to be addressed.

In the same way, the positioning of different elements within a specific space needs to be carefully studied when defining the BIM model to keep their position once the IFC document is exported. This is because, from a CMMS database logic, each element or object subjected to maintenance has to be placed in a specific location even if components like electrical cables or HVAC channels might cross many rooms and belong to a full level of a building and not to a single space. This software logic is opposite to the BIM one where objects and elements can belong to different spaces.

Furthermore, the data structure of BIM systems does not recognize different buildings as belonging to the same complex of buildings if these are drawn within a single model, but there is the need of managing the complexes through a single CMMS database. Several users have encountered this problem and a possible solution was subsequently integrated into the manuals and help centers of software manufacturers, as they suggested solving the problem by creating different models so that it is possible to have separate metadata for each building. This aspect also turned out to be critical for CMMS data import and therefore, in the work hereby presented, the question of how to manage IFC to converge different buildings into the same complex during conversion and import within the CMMS database is addressed.

2.4. CMMS database maintenance

A particular investigated aspect related to CMMS software is database maintenance. As Abdalaal and Shukri [36] studied, it is widely recognized that the time and effort needed to select and implement CMMS are well spent given the benefits and savings they offer in managing the maintenance function, but constant attention must be given to keeping the system up to date and supplied with input data. Nevertheless, this aspect is not always immediate since assets changes take place after the implementation and deployment of the CMMS database. If new and updated data comes from the revised BIM model, it is usually necessary to make a complete update of the CMMS database. This issue has been confirmed by Facility Managers and CMMS software developers.

When something concerning spatial aspects is changed within the BIM model after the completion of the building, it might create mismatches within the CMMS database, thus making it necessary to fully re-import the needed information. This results in the need to re-doing a complete import of a new database each time that something changes to avoid possible issues. Therefore, it emerges the necessity of developing a strategy that allows to adopt an incremental update of the CMMS database based on subsequent updates of the BIM model.

2.5. Challenges

The review described above, on one hand, confirmed the criticalities anticipated in the introduction and, on the other hand, it permitted to identify specific challenges to face allowing to deal with them adequately.

- The first aspect to be tackled is technical and concerns the difficulty of using the information of the BIM model to generate the CMMS database. As studied in subparagraph 2.3 such difficulties include the (i) reproduction, in the CMMS database, of the geometric-spatial definition of the building and the various rooms (and their relation and link with floors, buildings, and building sites), the (ii) classification of the objects and elements subject to maintenance, and the (iii) positioning of such assets within a specific room. Such technicalities need labor force to implement and check an overly detailed BIM model and the resulting COBie export.
- The second point to handle covers both technical and procedural aspects. In fact, for the operation and maintenance phase not to be independent and disconnected from the design phase, it is necessary that the database of the CMMS software can derive directly from the BIM model without further verification or manual adjustment of the data. Indeed, as described in the introduction and paragraph 2.2, manual verification becomes a cumbersome and lengthy procedure, discouraging the use of BIM for maintenance.
- A further matter, as presented in paragraph 2.3, is the updating of the CMMS database following changes to the original BIM model due to modifications (e.g., extensions, maintenance, changes in the functions of certain parts of the building) of the building components. Such procedure is time-consuming and therefore not convenient for most contexts.
- Lastly, there is a procedural aspect that should be simplified: it concerns the implications of the interoperability requirements on the production phase of the BIM model. As analyzed in paragraph 2.2 tasks needed for the interoperability with CMMS software

might be too heavy – or are not within the competence - for the BIM designer, whose primary objective is the production of a model for the design and construction phase of the building. Such difficulties often lead to missing information that would be useful for the operation and maintenance phase.

3. Methodology

3.1. Research objectives

Considering the emerged criticalities and related challenges, a series of specific objectives and requirements was defined to reach the research purpose; they are:

1. A data output structure that is not only feasible and obtained independently of the BIM authoring software, but that also uses a common 'taxonomy' to describe the building spatial aspects considering an operation and maintenance perspective.
2. A user-friendly IFC-CMMS import procedure that is easy to use, needs few intermediate checks while preparing the BIM model, and needs no final sample output control when data are exported from BIM systems.
3. A tool that can handle changes or modifications of the original BIM model without having to re-import all data in the CMMS database.
4. Simple BIM modeling requirements, which do not introduce too many constraints and are therefore easily used and accepted by BIM designers during the building design phase.
5. Definition of a strategy to allow easy management of building complexes through a single conversion and import activity.

The first feature stems from a need for interoperability between different operators who often have different digital tools at their disposal. Such need is not something new, but the aim is that of further simplify the needed procedure. It was therefore decided to work based on IFC metadata since it is recognized by the main BIM modeling software and the use of IFC already allows the exchange of information on the same building between different BIM software as studied by Sampaio and Gomes [24].

Given the presence of different CMMS software on the market, which have different database logics, it is necessary to evaluate a 'common' BIM and CMMS 'taxonomy' – to be understood as a database structure – to replicate the strategy here presented to other programs as well. This feature is needed because CMMS software adopts different classifications and hierarchical trees with names that often do not coincide with the content (e.g., some developers name levels with 'plant' and others with 'level'). Based on this knowledge, which is linked to the previous feature and to the tree structure of the IFC that resembles the links of records within a database, it was decided to apply the logic already recognized by BIM software to create easier data imports for the maintenance software.

Regarding the aim of having a user-friendly import procedure, currently plug-ins to implement the COBie standard are used to export IFC data before their import in CMMS database. This double procedure, the first of selection before export and the second of import, has two distinct moments that can create difficulties for end users: (i) the preliminary phase of setting of basic data before exporting, and (ii) the manual verification of what has been exported. Regarding the first, it starts before the design of the architectural model and is not functional for the design of the 3D model nor the CMMS database but without it, it is not possible to export the necessary data. For example, Fig. 1 shows examples of the COBie data export setup page UI in Revit as: setting of the parameters for the datasheet; link of the *IfcSpace* to the Zones and Rooms for the COBie datasheet; setting of which parameters to use when exporting COBie fields. The second, at the moment, is unavoidable given the presence of an enormous amount of data that may present duplicate or missing information, as explained in subparagraph 2.2.

To overcome the abovementioned end users' difficulties, a solution could be of selecting the needed IFC within the CMMS software instead of doing a pre-selection within the BIM framework. By selecting data during the import phase there are two benefits: (i) it is already known what data the Facility Manager needs, and (ii) it is possible to pick necessary asset data without constrainedly importing all non-FM-related information populating the BIM model. This combination avoids the population of the CMMS database with unnecessary information and can make the whole process more feasible.

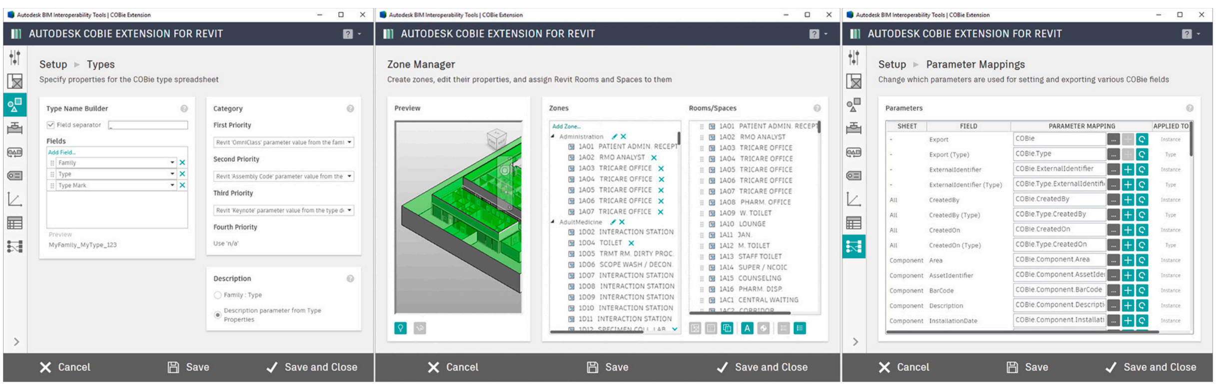

Fig. 1. Screenshots of the UI COBie extension for Revit with part of the many settings to check and information to add to have a correct data export.

The third objective considers the average life of buildings and their facilities, and consequently the possibility that they will undergo changes or upgrades during their period of use. It is, therefore, necessary to be flexible and capable of managing possible changes to the assets that can undergo maintenance e.g., two rooms merged into one or an addition to the existing building. This adaptability, however, must not result in repetitions of the entire procedure, but rather in updates or additions of data within the existing database without having to delete and rewrite it for each possible change.

Regarding the fourth challenge, it considers the need to guide designers to manage BIM models in anticipation of their use for Facility Management purposes. As already mentioned above, the new procedure starts from the idea that almost all data is filled in by the designers with normal data entry, but it will be necessary to add extra actions, or additional strategies, to obtain a BIM model compatible with the requests of maintenance software. However, these extra actions should be reduced to the minimum to keep the procedure user-friendly and acceptable to BIM designers of the architectural model. This also helps to reduce the amount of time spent preparing the BIM model.

Finally, linked to the management of the BIM model, the effort of preparing different drawings for each building of a building complex should be kept at minimum, and the CMMS database should be able to select the needed data as autonomously as possible.

3.2. Method

The development of the practical tool and operational procedure was based on a four-step approach (Fig. 2).

The first step was the definition of a simple but, at the same time, agreeable CMMS data structure and corresponding 'taxonomy' to describe the building's spatial aspects, recalling as closely as possible reality, and being able to adapt to as many types of buildings. The method was to study existing software and their functionalities. The necessary information was then collected through a comparison with a company that deals with Facility Management (GEMMO SpA an Italian company specialized in Facility Management for different-sized companies) and uses various CMMS software already on the market. After cataloging the functions required by the Facility Manager it was possible to hypothesize a 'taxonomy' that would allow a data structure and software logic adaptable to the various cases that can be encountered in the maintenance field. The information collection carried out with GEMMO referred to a real structure whose maintenance they are responsible for: the Feltre Hospital. Given the complexity and importance of hospital infrastructure, it was taken as a real case for the development phase of the tool and procedure.

The second step focused on the comparison of the data structure defined in step 1 with the IFC open data schema, to understand how the IFC and CMMS match currently works. This step was necessary to identify which IFC fields match the CMMS 'taxonomy' defined in step 1 and, consequently, which should be considered during the import procedure. Only IFC useful for operations maintenance should be elaborated during the import procedure since the complete import of all IFC of the BIM model would be excessive and not useful in its entirety in a CMMS database. To reach this goal, software developers were consulted and the IFC4 reference view was analyzed.

The third step was oriented on defining an import grid – a query schema – on which to base the development of a tool for reading IFCs and converting them into a database structured according to the 'taxonomy' defined above. Having previously established which information should be imported from the.ifcxml or.ifc file (two formats of IFC that can be exported from a BIM model), and what hierarchy to hold, the software house started the development of the tool.

The final step involved the validation. A test BIM model was created following the modeling requirements, and it was designed to present useful features which could help validate both the operating procedure and the import tool. Tests were carried out with the BIM model to both assess the effective functioning of the tool and to check how to manage the update of the CMMS database according to any subsequent modification of the building and its assets.

4. Results

4.1. Taxonomy definition and mapping with IFC

As there are many different people involved in the process from different professional backgrounds (e.g., architects, engineers, BIM designers, builders, facility managers, CMMS software developers), the creation of an agreed 'taxonomy' was considered necessary (Table 1) to create a shared knowledge level between various stakeholders. Therefore, even if such 'taxonomy' can be obvious for many actors, it allows to better bridge the differences between the design, construction, operation and maintenance stages and what happens in a real application, and it further simplifies the process and the interoperability system.

From the analysis of the state of the art, it was possible to verify the 'taxonomy' used in most common CMMS software to describe the building structure and the logic that links each record contained within CMMS databases. This logic is not universal, but we can consider it the most common and intuitive based on what emerged from Facility Managers. This taxonomy closely reflects the categories used by BIM software and reflects the structure of real buildings.

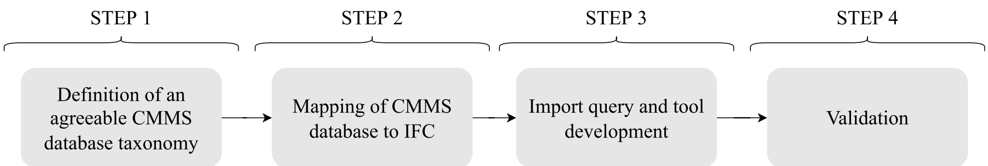

Fig. 2. Four-step approach adopted to reach the research purpose.

Table 1
IFC-CMMS matching.

IFC METADATA	CMMS DATABASE TAXONOMY
IfcSite	Building complex
IfcBuilding	Building
IfcBuildingStorey	Floor
IfcSpace	Room
Ifcs* linked to IfcSpace	Objects and Elements

* We refer to IFCs defining e.g., furniture, HVAC, lights

[a] We refer to IFCs defining e.g., furniture, HVAC, lights.

Furthermore, the analysis allowed to better study existing advanced tools, such as IBM Maximo, which are already integrated within CAD and other engineering applications. Such tools, by being already integrated with CAD, do an automatic taxonomy redefinition to create the link with the CMMS database. However, they are designed to manage big building complexes, making them inaccessible to small and medium-sized enterprises which do not have the necessary budget or skill requirements. Hence, this method demonstrates how a simpler strategy does not limit the functionality of this type of system, but rather facilitates their adoption when managing small buildings.

To create a functional procedure, a schema explaining the correspondence between IFC - which defines a common standard among BIM software - and the structure of the CMMS database was initially studied. Therefore, this schema (Table 1) was elucidated by using a table that explains the matching between the IFC tree logic and the structure used by the CMMS database. This matching table considers not only the BIM standard but also the standard structure of CMMS software, allowing the procedure to be compatible with different CMMS and different cases. Nevertheless, it is recommended to define such schema within the actors before the final import into the CMMS database.

4.2. Query definition to support the software development

The tool development has been carried out by Prometeo Srl, and it used the query schema hereafter described. The logic used while developing the tool was of keeping an agile and intuitive procedure, thus without losing the links with spatial aspects. Such links are of extreme importance when using the data within the CMMS environment. In fact, to simplify the work of maintenance men in bigger locations, an option to locate and visualize objects and elements inside a visual 2D or 3D representation of the building would be extremely helpful.

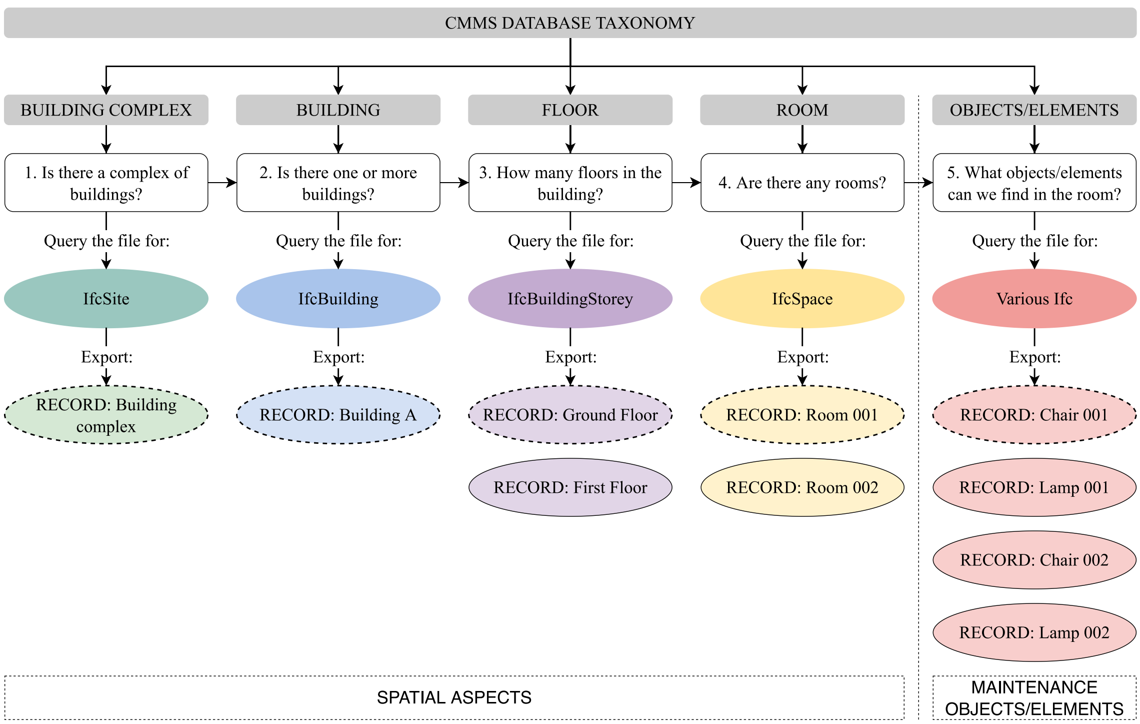

Fig. 3. Tool query logic.

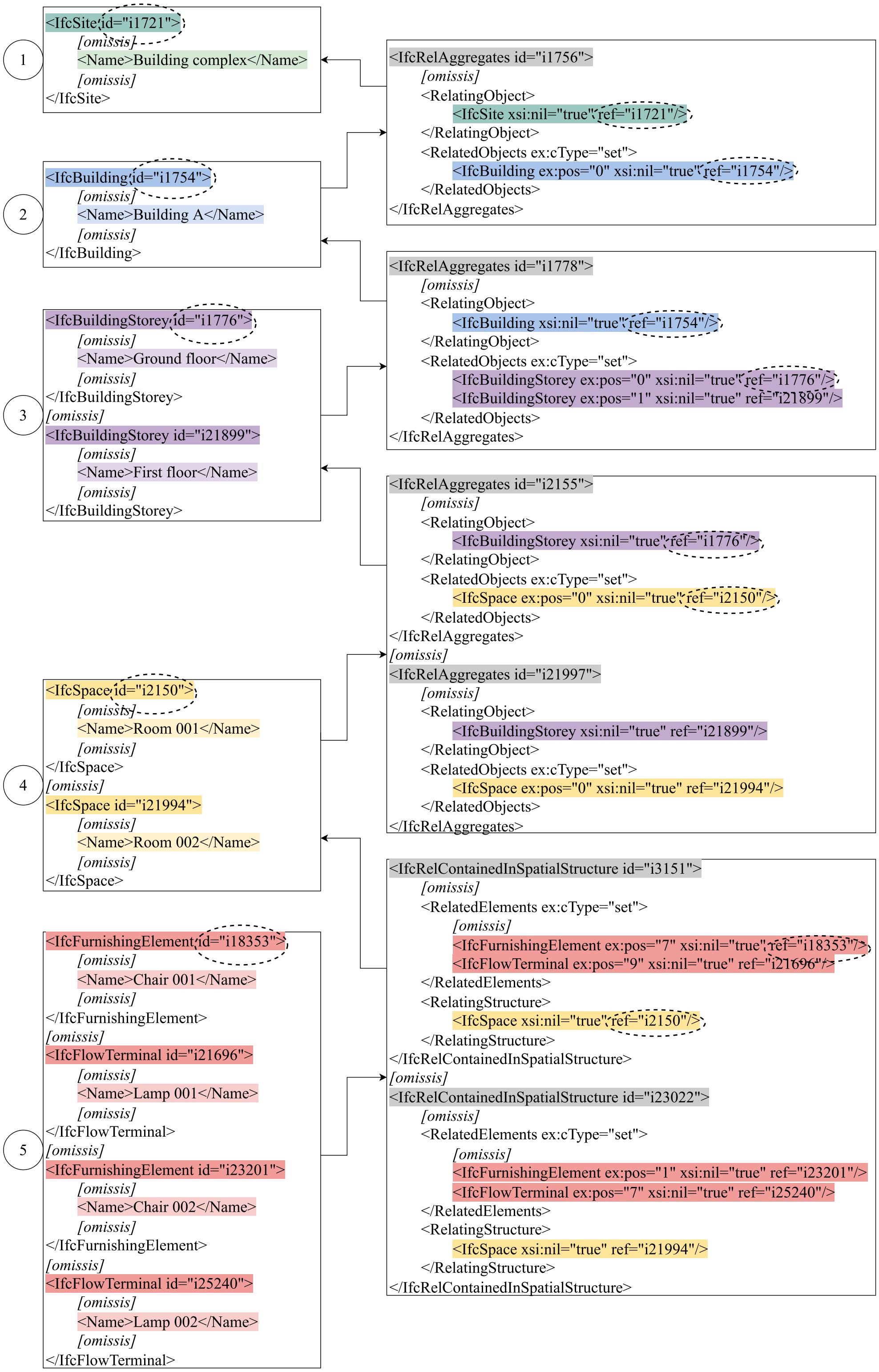

Fig. 4. Detail of the data exported from the IFC file.

The conversion and import tool extracts data from the.ifcxml or.ifc file by making a query based on the different categories that define the CMMS taxonomy. In Fig. 3 it is possible to see the query logic starting from the CMMS category (e.g., is there a complex of buildings?), the associated query (e.g., search for IfcSite metadata) and the result (e.g., ‘building complex’ record). The query is then repeated for each CMMS category. The schema of Fig. 3 is an example based on the test model (Fig. 5 in 4.3 Validation).

By analyzing more in detail the functioning of the tool, it is possible to see how each information contained in the file is linked (via tree structure in.ifcxml file, or via recalled ‘id’ in.ifc and.ifcxml files) to the entity above and it follows a structure that is similar to the levels within the CMMS database.

Fig. 3 shows the tool’s query logic. It selects the data, contained in the IFC file, to be imported based on questions on the presence of building complexes, how many buildings, how many floors per building, how many rooms per floor, and how many objects and elements requiring maintenance are there per room. These queries, made by the tool, investigate the data contained in an.ifcxml file as shown in Fig. 4. Indeed, all the information within the.ifcxml file is linked to each other by several IFC and strings of code e.g., *IfcRelContainedInSpatialStructure* and its properties. This results in the capability to define where a specific object and element is located. Hereafter the link of the object ‘Chair 001’, to ‘Room 001’, to ‘Ground floor’, to ‘Building A’ up to ‘Building complex’ from Fig. 3 is used as an example to show how the IFC are interlinked one another (the data hereby mentioned are the ones within dashed ovals in Fig. 3, and their ‘id’ are highlighted with a dashed oval in Fig. 4).

To explain the link logic, it is possible to start from the box ‘5’ that presents the various IFC within a room, and specifically, the *IfcFurnishingElement* data is the one defining ‘Chair 001’ and ‘Chair 002’ by using the string ‘<Name>’, while *IfcFlowTerminal* defines the object that has ‘<Name>’ as ‘Lamp 001’ and ‘Lamp 002’.

It can be seen that ‘Chair 001’ is defined by ‘id = “i18353”’. In the box on the right such objects are linked to the IfcSpace. The data *IfcRelContainedInSpatialStructure* links ‘id = ”i18353′′′ with ‘<RelatingStructure>’ that is ‘<IfcSpace xsi:nil = "true" ref = "i2150"/>’, meaning that ‘Chair 001’ is located in the IfcSpace defined by ‘id = ”i2150′′′. The ‘id = ”i2150′′′ can be found in box ‘4’ and it is called ‘Room 001’.

Such ‘Room 001’ is linked to the floor by the data *IfcRelAggregates* that links ‘id = ”i2150”’ with ‘<RelatingObject>’ that is ‘<IfcBuildingStorey xsi:nil = "true" ref = "i1776"/>’, meaning that ‘Room 001’ is located in the IfcBuildingStorey defined by ‘id = "i1776′′′. The ‘id = "i1776′′′ can be found in box ‘3’ and it is called ‘Ground floor’.

This floor is linked to the building by the data *IfcRelAggregates* that links ‘id = ”i1776”’ with ‘<RelatingObject>’ that is ‘<IfcBuilding xsi:nil = "true" ref = "i1754"/>’, meaning that ‘Ground floor’ is located in the IfcBuilding defined by ‘id = "i1754′′′. The ‘id = "i1754′′′ can be found in box ‘2’ and it is called ‘Building A’.

This building is linked to the building complex by the data *IfcRelAggregates* that links ‘id = ”i1754′′′ with ‘<RelatingObject>’ that is ‘<IfcSite xsi:nil = "true" ref = "i1721"/>’, meaning that ‘Building A’ is located in the IfcSite defined by ‘id = "i1721′′′. The last link is with box ‘1’ where ‘id = ”i1721”’ can be found with its ‘<Name>’ ‘Building complex’.

The developed tool, by querying this structure, collects the information to create linked records that will feed the CMMS database. In the output file (.xmls,.csv,.xml), as it is possible to see in the example of Table 2, data are organized linearly, which means that to each row corresponds an entity e.g., an object, a room, a building. The hierarchy is indicated in the code; for example, the code of Room 001 which is located on the Ground Floor in Building A is the sum of the codes of the previous entities. In the tool procedure, each piece of information considered to be useful for maintenance purposes and linked to the IfcSite, will be copied on the corresponding record into the database (address). After the information is collected, the tool will search for and import the information about the IfcBuilding and, since it is linked to the previously defined IfcSite, this record will be on a lower hierarchical level. This logic will be used to retrieve all useful data; for example, IfcBuildingStorey will be a lower level and will be linked to IfcBuilding, the same for IfcSpace that goes underneath IfcBuildingStorey and is linked to it. Once the tool gathers all data from the spatial aspects levels it starts to import each object and element that needs maintenance. All these data will be on the same level, but each will be linked to the space (room) where it is located. To avoid the conversion and import of objects and elements that do not need maintenance it is defined in advance which data to add based on a specific nomenclature.

Regarding the issue of handling updates to the BIM models used to acquire the data for the first import, and to allow subsequent modifications of the database without performing a new complete import, the developed tool includes the data import logic based on

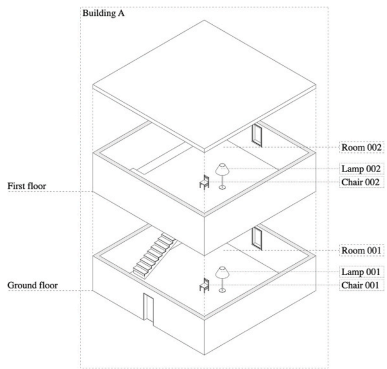

Fig. 5. Axonometric exploded view of the sample model used for the test.

Table 2
Database structure resulting from the conversion and import operation of the tool.

ADDRESS	SPATIAL ASPECTS CODE	SPATIAL ASPECTS DESCRIPTION	OBJECT OR ELEMENT CODE	OBJECT OR ELEMENT DESCRIPTION
Building complex	01	Building A		
	0101	Ground Floor		
	01010001	Room 001	010100010001	Chair 001
			010100010003	Lamp 001
	0102	First Floor		
	010200001	Room 002	010100010002	Chair 002
			010100010004	Lamp 002

the use of the *GlobalID*. Every single object or element contained within a BIM model has a personal and univocal *GlobalID*, which doesn't change if the same BIM document is used. This allows for a control logic where the tool, for imports subsequent the first, checks and compares every *GlobalID* in both the IFC file and the CMMS database to verify possible differences.

The import logic according to *GlobalID* does not exclude the use of the 'id' logic abovementioned, but it is complimentary. During an import procedure aimed at updating an already existing database, the tool follows these rules:

- if the *GlobalID* is not present in the CMMS database, the import procedure creates a new element e.g., building, room, plant, component.
- if the *GlobalID* is already present in the CMMS database, the procedure updates the characteristics and information related to the specific system or component.
- if *GlobalID* present in the CMMS database are no longer present in the import IFC file, the relative element is marked as 'Deactivated' in the database and will no longer be displayed to end users in the lists of the various web and desktop applications belonging to the software suite.

This *GlobalID* management shows that the tool has a metadata capability. See section 4.4 for the *GlobalID* exceptions management.

4.3. Validation

To validate the method a test was carried out. A sample building (Fig. 5) was created within different BIM environments (Archicad 24 and Revit 2021). The import presented in Fig. 6, based on the model from Fig. 5, was done by using the tool developed in the research and embedded in a CMMS software. The sample building ('Building A') was designed with two floors ('Ground floor' and 'First floor') that presented a room on each ('Room 001' on 'Ground floor' and 'Room 002' on 'First floor'). To check if the import and relations between different elements of a BIM model and the CMMS database were correct, a chair ('Chair 001' and 'Chair 002') and a lamp ('Lamp 001' and 'Lamp 002') were added for each room to verify the object-room-floor link.

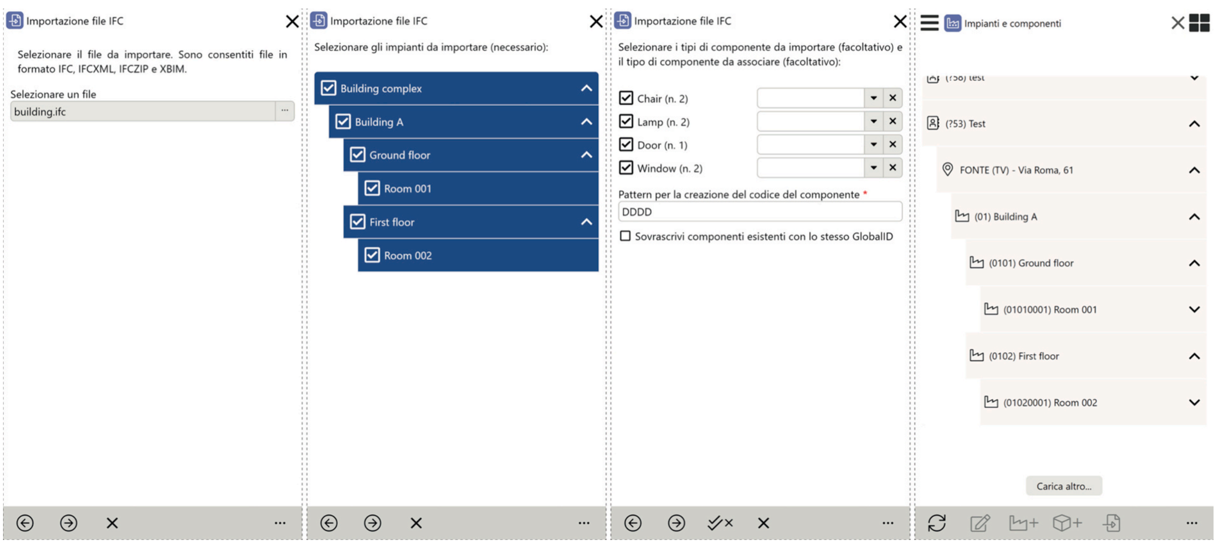

Fig. 6. IFC import procedure (from the left: select file; select which building complex, building, floor, room to import; select which objects and elements to import; CMMS interface after import) [Interface is in Italian, the mother tongue of the software house Prometeo Srl].

The import test produced a database with the spatial aspects recorded with the correct hierarchy, incorporating the building structure, the objects and the elements with the spaces they belong to. The test thus made it possible to have a first validation of the developed tool.

After the test and validation, an extra feature was added to allow the selection of groups of objects and elements of the same type. This simplifies the import procedure since the user does not have to select each object and element, e.g., it is possible to import the group 'Chair' without selecting each of them. Hereafter, some screenshots from the import test.

The procedure (presented in Fig. 6) starts by selecting the IFC file to be imported (*importazione file IFC*) within the CMMS interface. Once the tool page is open it is possible to select the file to import (*selezionare un file*). After moving to the next step, it is possible, but not mandatory, to choose which building complex, building, floor, or room to import. Once the spatial aspects are defined, the further step is to choose which objects and elements to add to the database. After the import procedure is concluded, the filled database can be accessed through the CMMS interface.

4.4. Modeling requirements

When BIM is used for operation and maintenance activities, it is necessary to comply with few extra modeling requirements compared to the design phase.

The main relevant rule is related to the definition of IfcSpace. When modeling in BIM, the software is not set to automatically register a 'room', defined by walls and slabs and roof, as 'space'. This means that there is the need to manually create a 'zone' – as it is called within the BIM environment – in the plan view to generate an IfcSpace. Once the perimeter of the room is defined by using the 'zone' instrument, the height will be defined automatically based on the level settings defined for the 3D model. Once the structural elements of the building are drawn, this activity should be performed for each room to create the IfcSpace data and their link with IfcBuildingStorey, IfcBuilding, and IfcSite. This operation can be done at any time (e.g., either when the building structure is defined or when a floor is completed).

Another relevant rule is related to the management of building complexes. To manage multiple buildings in the same IfcSite, it is necessary to model each building in a separate BIM file, otherwise, the exported IFC will not be able to recognize which element is from which building. This difficulty is linked to the BIM logic, which works based on levels (floors) and, from an IFC point of view, it results in data from different buildings all linked to the same floor. For this reason, besides creating separate models, it is mandatory to define the same IfcSite name within each model property to allow the identification of a shared *GlobalID* for the IfcSite when uploading data to the CMMS database. The first export from BIM will result in different IFC files, one for each building that, after the conversion and import operations, will be recognized by the CMMS software as buildings from the same complex.

Regarding the management of objects and elements that are of interest for maintenance, it is mandatory to give a custom name to each. Once the first custom name is defined e.g., Chair 001, each object from that category will be automatically renamed by the BIM software with the custom name followed by progressive numbering. This helps during the import procedure to only select objects and elements that are needed for maintenance instead of importing a higher amount of data that might not be useful.

Concerning the possibility to manage subsequent updates of the CMMS database, it is necessary to pay attention to the *GlobalID* aspect. The GlobalID defines which objects and elements are to import, thus it is necessary to avoid its changes to allow the correct import procedure. Such changes to the *GlobalID* might occur if the model was modified using different BIM software, or if the model was redesigned from scratch. Nevertheless, issues with *GlobalID* can be avoided by always updating the same model that was used for the first import. In addition, the use of a customized name for the objects and elements to be imported, combined with the *GlobalID*, always allows for a counter-verification.

5. Discussion

The research studied and developed a method to allow interoperability between BIM and CMMS software without the need of using sophisticated plug-ins and tools. This demand raised from some limitations found by Facility Managers that, currently, are supposed to use different software or complex processes to gather desired data from BIM models. Moreover, such models do not always comprise all the information that the operation and maintenance phase needs, resulting in the necessity to supplement the derived database with manual data entry.

As these initial requirements were reached, two extra challenges were faced.

The first one is linked to the management of CMMS database updates, which can be of complex handling. This is because there is no intuitive and automatic tool able to have subsequent record updates instead of creating a completely new set. Nevertheless, by using the herein proposed logic that uses the *GlobalID* data with the support of custom 'id', it is possible to update the CMMS database and all its records without final data verifications. The *GlobalID* logic also allows to use the CMMS database not only as a repository but also as a documentation of changes over time, to keep track of the modifications that could affect the operation and maintenance phase.

The second concerns the still open debate on the CMMS level of detail of the information needed for the operation phase [37,38]. In fact, the method developed is independent of the LOD of the original BIM model, allowing the user to choose the best solution based on the specific circumstances without changes in the procedure or the tool.

Although the research results reached a robust technology readiness level, the proposed method can be further implemented and refined by improving the investigation and validation phases. The first improvement might involve the data collected from Facility Managers. A more robust and structured interview with all the stakeholders involved in the operation and maintenance phase, instead of informal confrontations with Facility Managers, could help in better framing the desiderata and existing issues to be afterwards transferred in the procedure or implemented in the tool. The second aspect is linked to the validation phase. The opportunity of testing the practical tool and operational procedure on real case studies, instead of on a sample building, would be beneficial to better frame certain aspects as the actual amount of data linked to IFCs. Implementation on a real case study would help in better understanding how data filling is done in more complex BIM models, and how exported big data are processed by the tool. By further studying these two aspects, the method could build ever stronger foundations for its implementation.

6. Conclusions

This research produced an integrated method, composed of a tool and an operational procedure, to facilitate and enable the use of BIM data in the operation and maintenance phase of buildings of modest size, for which the use of complex software is not cost-effective.

The new tool, and the operational procedure, present:

1. Data output structure that is as feasible as possible, to be obtained independently of the BIM authoring software. Therefore, such structure can use a common 'taxonomy' to describe the building's spatial aspects.
2. A user-friendly import procedure that does not need a final sample output control.
3. A tool that can handle incremental changes or modifications of the original BIM model.
4. Simple modeling requirements that can be implemented even by new BIM users.
5. Definition of a strategy to allow easy management of complexes of buildings.

Nevertheless, some issues are still open. The procedure assumes that it must be clear from the beginning that the BIM model will also be used for maintenance purposes. Such assumption can be correct if the client, or final user, is made aware of the potential of the method. Indeed, if the 3D model is not compliant with the abovementioned modeling requirements, the BIM export will hardly be compatible and consistent with the developed method. One solution to overcome this issue could be to include such requirements in contracts. This would allow to demand the use of BIM models for the design phase, with the clause for it to be produced considering its subsequent use for the operation and maintenance phase.

Regarding the implication of the research, the results and outputs are presented in sufficient detail to allow their free re-use. In particular, the description of the query logic allows for the independent development of the data import tool. Likewise, the given modeling requirements are sufficient to allow others to correctly prepare the BIM model to export the.ifcxml for its subsequent query.

In conclusion, if this method were adopted, the work of Facility Managers would be greatly simplified. This is not only in terms of initial retrieval of maintenance-related information, but also in relation to the optimized management of the building over time. This would allow a better management of the spatial aspects of both building complexes and the buildings themselves and, consequently, a correct positioning of the elements to be maintained. In addition, the correct management with unique codes for each element would allow the use of the CMMS database as an archive not only of the current situation but also of the history of what happened inside the building. Having a record of changes would allow more efficient interventions in case of problems. Furthermore, such interoperable systems could be further linked with real time sensors and with storage records.

CRediT author statement

Massimiliano Condotta: Conceptualization, Methodology, Validation, Investigation, Writing- Original draft preparation, Writing-Reviewing and Editing, Supervision, Funding acquisition.

Chiara Scanagatta: Validation, Investigation, Data Curation, Writing-Original draft preparation, Writing- Reviewing and Editing, Visualization.

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.

Data availability

No data was used for the research described in the article.

Acknowledgements

The work here presented has been realized thanks to the grant *Programma Operativo Regionale del Fondo Europeo di Sviluppo Regionale 2014–2020 (POR FESR 2014–2020)* of the Veneto Region. The paper is the result of the joined research activity of the authors. Prometeo Srl software house and GEMMO SpA Facility Management gave support for technical aspects and tests.

References

[1] UNI EN ISO 19650:2019 (2019).
[2] E. VanDerHorn, S. Mahadevan, Digital twin: generalization, characterization and implementation, Decis. Support Syst. 145 (2021), https://doi.org/10.1016/j.dss.2021.113524.
[3] K. Chen, W. Chen, C. Ting Li, J.C.P. Cheng, A BIM-based location aware AR collaborative framework for facility maintenance management, J. Inf. Technol. Construct. 24 (2019) 360–380. http://www.itcon.org/2019/19.

[4] H. Dawood, J. Siddle, N. Dawood, Integrating IFC and NPL for automating change request validations, J. Inf. Technol. Construct. 24 (2019) 540–552, https://doi.org/10.36680/j.itcon.2019.30.
[5] S. Karimi, I. Iordanova, D. St-Onge, Ontology-based approach to data exchanges for robot navigation on construction sites, J. Inf. Technol. Construct. 26 (2021) 546–565, https://doi.org/10.36680/j.itcon.2021.029.
[6] C. Lemaire, L. Rivest, C. Boton, C. Danjou, C. Braesch, F. Nyffenegger, Analyzing BIM topics and clusters through ten years of science publications, J. Inf. Technol. Construct. 24 (2019) 273–298. http://www.itcon.org/2019/15.
[7] K. Katsigarakis, G.N. Lilis, D. Rovas, A cloud IFC-based BIM platform for building energy performance simulation, in: Proceedings of the 2021 European Conference on Computing in Construction, 2021.
[8] D. Kazado, M. Kavgic, R. Eskicioglu, Integrating building information modeling (BIM) and sensor technology for facility management, J. Inf. Technol. Construct. 24 (2019) 440–458. https://www.itcon.org/2019/23.
[9] A. Mirzaei, F. Nasirzadeh, M. Parchami Jalal, Y. Zamani, 4D-BIM dynamic time-space conflict detection and quantification system for building construction projects, J. Construct. Eng. Manag. 144 (7) (2018), https://doi.org/10.1061/(ASCE)CO.1943-7862.0001504.
[10] A. Preut, J.P. Kopka, U. Clausen, Digital twins for the circular economy, Sustainability 13 (18) (2021), https://doi.org/10.3390/su131810467.
[11] H. Halttula, H. Haapasalo, R. Silvola, Managing data flows in infrastructure projects – the lifecycle process model, J. Inf. Technol. Construct. 25 (2020) 193–211, https://doi.org/10.36680/j.itcon.2020.012.
[12] N. Jung, T. Hakkinen, M. Rekola, Extending capabilies of BIM to support performance based design, J. Inf. Technol. Construct. 23 (2018) 16–52. http://www.itcon.org/2018/2.
[13] G. Sibenik, I. Kovacic, Interpreted open data exchange between architectural design and structural analysis models, in: *Journal Of Information Technology In Construction (ITcon)*, Special Issue: 'CIB World Building Congress 2019: Constructing Smart Cities', vol. 26, 2021, pp. 39–57, https://doi.org/10.36680/j.itcon.2021.004.
[14] M. Viklund Tallgren, M. Roupé, M. Johansson, P. Bosch-Sijtsema, BIM-tool development enhancing collaborative scheduling for pre-construction, J. Inf. Technol. Construct. 25 (2020) 374–397, https://doi.org/10.36680/j.itcon.2020.022.
[15] M.K. Watfa, A.E. Hawash, K. Jaafar, Using building information and energy modelling for energy efficient designs, J. Inf. Technol. Construct. 26 (2021) 427–440, https://doi.org/10.36680/j.itcon.2021.023.
[16] F. Shalabi, Y. Turkan, BIM–energy simulation approach for detecting building spaces with faults and problematic behavior, J. Inf. Technol. Construct. 25 (2020) 342–360, https://doi.org/10.36680/j.itcon.2020.020.
[17] J. Angers, As-built drawings: as good as gold, J. AWWA (Am. Water Works Assoc.) 93 (12) (2001) 46–48, https://doi.org/10.1002/j.1551-8833.2001.tb09351.x.
[18] G.A. Adams, GIS—an alternative to as-built drawings, J. AWWA (Am. Water Works Assoc.) 94 (2) (2002) 50–51, https://doi.org/10.1002/j.1551-8833.2002.tb09404.x.
[19] M.C. Carnero, J.L. Novés, Selection of computerised maintenance management system by means of multicriteria methods, Prod. Plann. Control 17 (4) (2006) 335–354, https://doi.org/10.1080/09537280600704085.
[20] Y. Alani, M.H. Hafeez, J. Patacas, S. Rodriguez, S. Dawood, An exploration of the owner's requirements for BIM in facilities management, in: Proceedings of the CIB World Building Congress 2019, 2019.
[21] J. Patacas, N. Dawood, V. Vukovic, M. Kassem, BIM for facilities management: evaluating BIM standards in asset register creation and service life planning, J. Inf. Technol. Construct. 21 (2015) 434–455. http://www.itcon.org/2016/27.
[22] J. Patacas, N. Dawood, D. Greenwood, M. Kassem, Supporting building owners and facility managers in the validation and visualization of asset information models (AIM) through open standards and open technologies, J. Inf. Technol. Construct. 20 (2016) 313–331. http://www.itcon.org/2015/20.
[23] J. Patacas, N. Dawood, M. Kassem, BIM for facilities management: a framework and a common data environment using open standards, Autom. ConStruct. 120 (2020), https://doi.org/10.1016/j.autcon.2020.103366.
[24] A.Z. Sampaio, A.M. Gomes, BIM interoperability analyses in structure design, CivilEng 2 (2021) 174–192, https://doi.org/10.3390/civileng2010010.
[25] H. Lai, X. Deng, Interoperability analysis of IFC-based data exchange between heterogeneous BIM software, J. Civ. Eng. Manag. 24 (7) (2018) 537–555, https://doi.org/10.3846/jcem.2018.6132.
[26] M. Trzeciak, A. Borrmann, Model Exchange between Revit and AllPlan Using IFC: a Case Study for a Bridge Model, Technische Universitat Munchen, Department of Civil, Geo and Environmental Engineering, Chair of Computational Modeling and Simulation, 2018.
[27] S. Ait-Lamallam, R. Yaagoubi, I. Sebari, o. Doukari, Extending the IFC standard to enable road operation and maintenance management through OpenBIM, Int. J. Geo-Inf. 10 (2021) 496–522, https://doi.org/10.3390/ijgi10080496.
[28] M. Laakso, A. Kiviniemi, The IFC standard – a review of history, development, and standardization, J. Inf. Technol. Construct. 17 (2012) 134–161. http://www.itcon.org/2012/9.
[29] F. Noardo, Ohori K. Arroyo, T. Krjinen, J. Stoter, An inspection of IFC models form practice, Appl. Sci. 11 (2021) 2232–2259, https://doi.org/10.3390/app11052232.
[30] B.W. East, Construction Operations Building Information Exchange (COBie), Construction Engineering Research Laboratory (Publication No. ERDC/CERL TR-07-30), 2007.
[31] Kristine Fallon Associates, Assessment of Life Cycle Information Exchanges (LCie), Construction Engineering Research Laboratory (Publication No. ERDC/CERL, 2013. CR-13-6).
[32] B.W. East, N. Nisbet, T. Liebich, Facility management handover model view, J. Comput. Civ. Eng. 27 (1) (2013) 61–67.
[33] S. Lavy, S. Jawadekar, A case study of using BIM and COBie for facility management, Int. J. Facilit. Manag. 5 (2) (2014).
[34] V. Kumar, E. Ai Lin Teo, Development of a rule-based system to enhance the data consistency and usability of COBie datasheets, J. Comput. Des. Eng. 8 (1) (2020) 343–361, https://doi.org/10.1093/jcde/qwaa083.
[35] V. Kumar, E. Ai Lin Teo, Perceived benefits and issues associated with COBie datasheet handling in the construction industry, Facilities 39 (5/6) (2020) 321–349, https://doi.org/10.1108/F-09-2019-0100.
[36] O.A.A. Abdalaal, M.I. Shukri, Implementation of computerized maintenance management system in maintenance workshop, Int. J. Eng. Appl. Sci. Technol. 4 (9) (2020) 100–103, https://doi.org/10.33564/IJEAST.2020.v04i09.010.
[37] Y. Hong, A.W.A. Hammad, A. Akbarnezhad, Forecasting the net costs to organisations of building information modelling (BIM) implementation at different levels of development (LOD), J. Inf. Technol. Construct. 24 (2019) 588–603, https://doi.org/10.36680/j.itcon.2019.033.
[38] S. Russo Ermolli, La digitalizzazione dei flussi informativi per la fase operativa: il caso della Apple Developer Academy, Techne 18 (2019) 235–245, https://doi.org/10.13128/techne-7535.