VOL. 9, NO. 1 (2023)

TOWARDS A NEW ETHICS IN BUILDING

TEMA
Technologies
Engineering
Materials
Architecture

Journal Director: R. Gulli

e-ISSN 2421-4574
DOI: 10.30682/tema0901

Guest Editors: A. Sanna, G. Di Giuda, L.C. Tagliabue

Assistant Editors: A.C. Benedetti, C. Mazzoli, D. Prati

Cover illustration: Jewish Museum Berlin by Daniel Libeskind, The Garden of Exile, Berlin, Germany, 2001. © Riccardo Gulli (2009)

TEMA
Technologies Engineering Materials Architecture

e-ISSN 2421-4574
ISBN online 979-12-5477-288-1
DOI: 10.30682/tema0901
Vol. 9, No. 1 (2023)
Year 2023 (Issues per year: 2)

Editor in chief
Riccardo Gulli, Università di Bologna

Associated Editors
Annarita Ferrante – Università di Bologna
Enrico Quagliarini – Università Politecnica delle Marche
Giuseppe Margani – Università degli Studi di Catania
Fabio Fatiguso – Università Politecnica di Bari
Rossano Albatici – Università di Trento

Editorial Board Members
İhsan Engin Bal, Hanze University of Applied Sciences – Groningen
Antonio Becchi, Max Planck Institute – Berlin
Marco D'Orazio, Università Politecnica delle Marche
Vasco Peixoto de Freitas, Universidade do Porto – FEUP
Stefano Della Torre, Politecnico di Milano
Giuseppe Di Giuda, Università di Torino
Luca Guardigli, Università di Bologna
José Luis Gonzalez, UPC – Barcellona
Francisco Javier Neila Gonzalez, UPM Madrid
Alberto Grimoldi, Politecnico di Milano
Antonella Guida, Università della Basilicata
Santiago Huerta, ETS – Madrid
Richard Hyde, University of Sydney
Tullia Iori, Università di Roma Tor Vergata
Raffaella Lione, Università di Messina
John Richard Littlewood, Cardiff School of Art & Design
Camilla Mileto, Universidad Politecnica de Valencia UPV – Valencia
Renato Morganti, Università dell'Aquila
Antonello Sanna, Università di Cagliari
Matheos Santamouris, University of Athens
Enrico Sicignano, Università di Salerno
Lavinia Chiara Tagliabue, Università di Torino
Claudio Varagnoli, Università di Pescara
Emanuele Zamperini, Università di Firenze

Assistant Editors
Cecilia Mazzoli, Università di Bologna
Davide Prati, Università di Bergamo
Anna Chiara Benedetti, Università di Bologna

Journal director
Riccardo Gulli, Università di Bologna

Publisher:
Ar.Tec. Associazione Scientifica per la Promozione dei Rapporti tra Architettura e Tecniche per l'Edilizia
c/o DICATECH - Dipartimento di Ingegneria Civile, Ambientale, del Territorio, Edile e di Chimica - Politecnico di Bari
Via Edoardo Orabona, 4
70125 Bari - Italy
Phone: +39 080 5963564
E-mail: info@artecweb.org - tema@artecweb.org

Publisher Partner:
Fondazione Bologna University Press
Via Saragozza 10
40123 Bologna - Italy
Phone: +39 051 232882
www.buponline.com

TEMA: Technologies Engineering Materials Architecture
Vol. 9, No. 1 (2023)
e-ISSN 2421-4574

TEMA Technologies Engineering Materials Architecture
e-ISSN 2421-4574
Vol. 9, No. 1 - (2023)

DIGITIZATION OF BUILDING SYSTEMS USING IFC TO SUPPORT PERFORMANCE ANALYSIS AND CODE CHECKING: STANDARD LIMITS AND TECHNOLOGICAL BARRIERS. A CASE STUDY ON FIRE SAFETY

Carlo Zanchetta, Maria Grazia Donatiello, Alessia Gabbanoto, Rossana Paparella

DOI: 10.30682/tema0901l

This contribution has been peer-reviewed.
© Authors 2023. CC BY 4.0 License.

Abstract

The study presented in this paper is part of a digitization project developed for the University of Padua. It aims at demonstrating how IFC (Internet Foundation Classes) ISO Standard can be used as a reliable data model to support Performance Analysis (PA) and code checking for construction disciplines. Fire Safety Engineering (FSE) is analyzed as a challenging test field because it highly affects different building aspects and highlights interoperability issues. The methodology proposed in the study consists in checking a digital approach to PA based on information classes that can express both users' requirements and performance specification of technical elements to develop computational code checking. This method is developed by creating virtual classes representing built systems and using relation classes and performance attributes to check if technical elements fulfil users' requirements. By forcing the model to be based on standardized information classes, the study verifies if IFC, as an ISO standard, can be used as a universal and scalable reference model for performance analysis and code checking. More specifically, the study focuses on the availability of IFC's information classes and attributes that define a PA model. This research verifies the achievement of the proposed goals for FSE (Section 2) and then highlights the interoperability limits that affect an IFC-based approach to computational FSE code checking (Section 3). Finally, the technical feasibility of the methodology's market implementation is presented (Section 4). The study's innovative approach is related to the fact that IFC is often analyzed as an information exchange format, not as a data model, where standardized relations between building ontologies can be simulated. Digital ontologies of relational aspects are experimented with by following this approach. These reports support code conformance analyses of the technical element performance specification. The study then indicates how the information modelling discipline could be shaped to encourage standardized code checking better.

Keywords

IFC, Performance analysis, Code checking, System engineering.

Carlo Zanchetta*
ICEA - Dipartimento di Ingegneria Civile, Edile e Ambientale, Università degli Studi di Padova, Padova (Italy)

Maria Grazia Donatiello
ICEA - Dipartimento di Ingegneria Civile, Edile e Ambientale, Università degli Studi di Padova, Padova (Italy)

Alessia Gabbanoto
ICEA - Dipartimento di Ingegneria Civile, Edile e Ambientale, Università degli Studi di Padova, Padova (Italy)

Rossana Paparella
ICEA - Dipartimento di Ingegneria Civile, Edile e Ambientale, Università degli Studi di Padova, Padova (Italy)

** Corresponding author:*
e-mail: carlo.zanchetta@unipd.it

1. INTRODUCTION

The performance verification of a technical element depends on the requirements that the element is expected to satisfy. These requirements can be expressed by the technological system to which this element belongs. Therefore, it is necessary to set up a compliance control policy to compare the systems' requirements and the related element's performance. Thus, the relationship between the technical element and the technological system must be communicated. At a methodological level, this approach finds equivalence in the discipline of value analysis. This discipline applied to the AEC industry (Architecture, Engineering and Construction) compares the cost of technical elements part of built systems (parametric or metric) to the performance of the systems these elements constitute (envelope, plant systems, load-bearing structures, fire prevention systems, etc.). In order to support evaluation processes in Value management procedures, there is a need to standardize the definition of Performance Analysis (PA) of building systems to compare the performances and costs of homogeneous technical solutions [1, 2].

The relation between the performance of functional units and their technical specifications is represented in the General Architecture Reference Model (GARM) [3]. This model is therefore replicated in the IFC (Internet Foundation Classes) data model published by buildingSMART, which is, at the same time, a widely diffused reference in the AEC industry and an information standard for software applications, as stated in IFC ISO 16739 standard. Concerning this aspect, there are several technological and disciplinary limits. Technical limits are related to the fact that many software applications don't fully implement the IFC data model but only a part of it. Disciplinary limits depend on the fact that building performance simulation and analysis are usually developed employing software applications that recreate disciplinary models in proprietary environments. Then, relational classes of IFC at the basis of the PA approach are not cited in published scientific works, while conceptual models representing disciplinary aspects, such as the GARM, are.

The study's methodology consists of creating virtual classes representing built systems and using relation classes and performance attributes to describe the relations between building systems and constituent technical elements. This relationship allows for developing the computational checking of technical elements' performance compliance to requirements specification.

This information modelling and code-checking technique exceeds current software limitations both in writing the *.ifc* file and reading it. The *Ifcopenshell* library (available at http://ifcopenshell.org) was tested to overhead these limits. Using this python library, IFC models can be edited and enriched when software applications can't write classes that exist in the current version of the IFC standard. The result of these tests was then validated using the application FZKviewer (available at https://www.iai.kit.edu/english/1648.php) to demonstrate full IFC compliance of the tested models to the ISO standard.

The correctness of the data model and reliability of the experimentation demonstrates that IFC, as an ISO standard, can be used as a reference model for performance analysis and code checking.

2. TEST ON IFC DATA MODEL MATURITY TO SUPPORT PA ATTRIBUTES

The test on the availability of adequate classes and attributes is organized into the following:

- analyze the availability of adequate attributes for compilation of domain-specific requirements in parent classes and then check for performance specifications attributes both in parent and child classes;
- monitor the relationship between spatial or functional superclasses and the related technical elements they host to compare a technical element's performance with the requirements of the space or system in which it is installed.

2.1. REQUIREMENTS AND PERFORMANCE SPECIFICATION IN IFC HIERARCHICAL STRUCTURES: FROM ATTRIBUTES TO THE USE OF IFCRELASSOCIATESCLASSIFICATION CLASS

In the IFC schema, most domain-specific attributes are assigned directly to constituent materials using the *IfcMaterialDefinition* resource. In this way, each technical element inherits domain-specific properties by material layer, profile or constituents. Further performance information of technical elements is given because each class hierarchically child of *IfcElement* (including *IfcBuildingElement*, *IfcSystem*, *IfcGroup*, and *IfcZone*) indicates several environmental and performance indicators.) shows several environmental and performance indicators.

To support a PA approach, spatial containment and functional superclasses should express general requirements to check if referring technical elements develop consistent performances.

To date, the validity of IFC's superclasses, such as *IfcBuilding* and *IfcSystem*, has been confirmed concerning their potential in terms of spatial organization and item instantiation. On the contrary, an effective standardization of information modelling and management regarding performance and condition assessment is not yet fully valid for the higher classes and the hierarchy of IFC.

In IFC, functional and technical systems are described using object grouping entities such as *IfcAsset*, *IfcInventory*, and *IfcSystem*. *IfcSystem* allows grouping elements sharing the same functions or objectives (construction systems, sectors, plants). This information can be defined by the Pset_ServiceLifeFactors, in which there are properties concerning the adjustment of the service life. Although the presence of these properties is appreciable, the information range is too restricted to face a fully interoperable PA methodology. The IFC implementation of classification references offers one possible solution to this limit. Further attributes can be implicitly linked using the *IFCRelAssociatesClassification* relationship. The assignation of an item to an IFC class and its reference to a classification system allows the possibility of compiling requirements, superclasses performances and constituent elements just by mapping the references in a specification matrix as the one expressed in the Specifiers Properties information exchange (SPie) [4].

2.2. FUNCTIONAL AND SPATIAL RELATIONS MAPPING

Creating relations between building elements and functional or spatial entities is fundamental for the correct setting of PA procedures. The objectified relationship, *IfcRelContainedInSpatialStructure*, is the typical procedure used to assign elements to a specific spatial structure: *IfcSite*, *IfcBuilding*, *IfcBuildingStorey* or *IfcSpace*. The IFC exporter automatically builds that relationship for both software used, Graphisoft ArchiCAD 24 and Autodesk Revit 2021. Sometimes further spatial connections are needed to trace the presence of a technical element in a spatial container such as *IfcZone* or *IfcSpatialZone*. These classes group two or more spaces by function (i.e., a fire compartment) or performance specification (i.e., thermal zone). Zones represent virtual groups of spaces that have no spatial structure assigned, while spatial zones have a spatial containment given through the *IfcRelSpaceBoundary* class. In this case, the connection is managed by the relationship *IfcRelReferencedInSpatialStructure*.

The *IfcSpatialZone* class is not yet correctly implemented in the modelling software analyzed. In Graphisoft Archicad 24, exporting this entity at all is impossible. Instead, in Autodesk Revit 2021, no system families represent the *SpatialZone*. The only way would be to create with the tool "Area" a tridimensional object and, through the *IfcExportAs* parameter, export the object as an *IfcSpatialZone*.

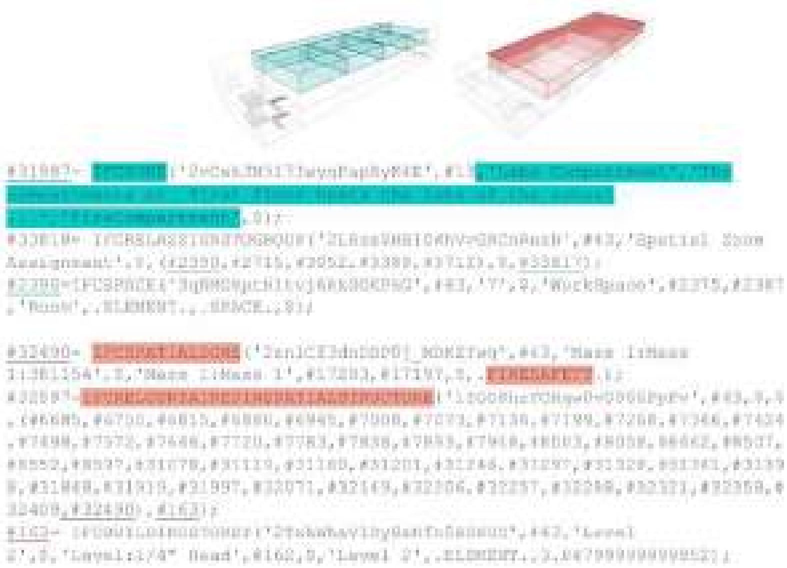

Fig. 1. *IfcZone and IfcSpatialZone in STEP file exported from Autodesk Revit 2021.*

IfcRelAssignsToGroup is used instead to aggregate spaces and entities in zones [5]. The IFC exporter manually adds that relationship for both software used, Graphisoft ArchiCAD 24 and Autodesk Revit 2021. Revit does not have a concept of architectural/fire safety zones. However, IFC zones are exportable using the shared room parameters (Fig. 1). By adding the shared text parameter "*ZoneName*" to the Revit "*Rooms*" category in the "*IFC Parameters*" parameter group, a user can specify the name of the zone that the room belongs to. Revit will then create one *IfcZone* for each unique "*ZoneName*" value and associate all the rooms with that value to that zone. In addition, the shared parameters "*ZoneObjectType*" and "*ZoneDescription*" can be used to set the object type and description of the *IfcZone*. A Revit room can be associated with up to 1000 zones by adding extra shared text parameters: including the shared parameters "*ZoneName #*" (where # = 2, 3, 4, etc.) adds more zones to a room. In addition, the "*ZoneObjectType*" and "*ZoneDescription*" parameters can also be similarly extended to, e.g., "*ZoneObjectType 2*". A room will be associated with each zone defined for that room. It is possible to export the properties of the *Pset_ZoneCommon* for a Zone, as shown in Figure 2, containing an extract of the IFC shared parameter file. However, since these parameters are associated with spaces, if a space belongs to different zones, it is not yet possible to associate Property Sets (such as *Pset_ZoneCommon*th *IfcZone*).

Other properties, such as *Pset_SpaceFireSafetyRequirements* or *Pset_SpaceOccupancyRequirements* that, according to IFC standards, could be assigned to the class *IfcSpace* or *IfcZone*; in Revit, can only be associated with spaces. A shortcut would be to use Revit's Zones in the "Spaces and Zones Panel" of the Analyze tab. However, this second way is correct only for HVAC zones. On the other hand, Graphisoft Archicad 24 exporter does not show any problems in this respect: its IFC project manager allows the creation of new spatial relations among spaces creating an IFC Zone.

Functional or logical relations between physical entities are then managed with *IfcRelAssignsToGroup* relation that can instantiate objects in any *IfcGroup* subclass: asset, inventory, or system.

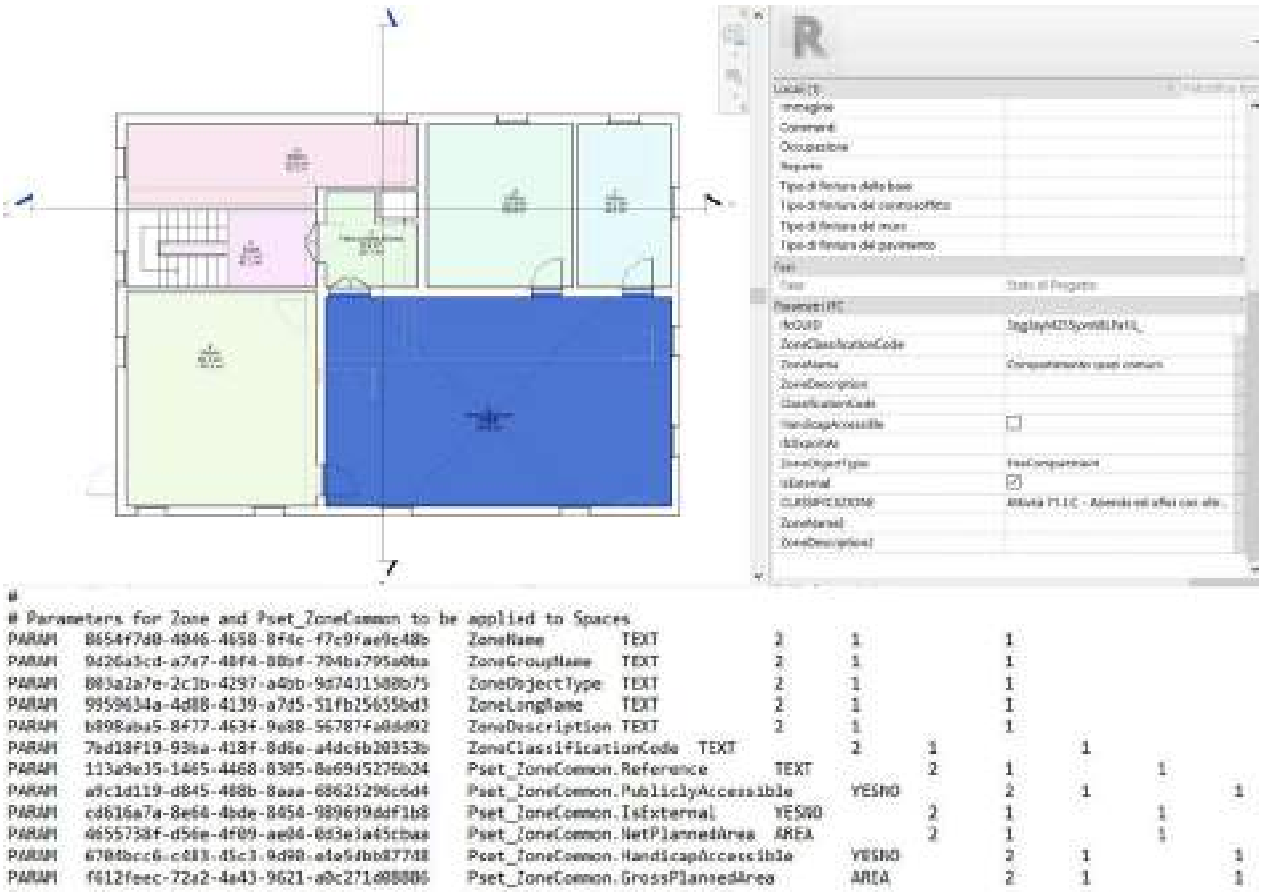

Fig. 2. *Revit IFC shared parameters.*

As demonstrated, the data model offers a complete set of possible connections so that it is possible to map any relation underlying performances of building systems and activate class-based PA procedures.

3. INTEROPERABILITY LIMITS

Not many types of software write the relations highlighted before, and even fewer can show their presence in a graphical or hierarchical representation. Besides the limits of information mapping and visualization, it is relevant to highlight that the maturity of the data model is higher than the possibility of writing those classes using commercial software applications. Several types of software do not read all the classes and attributes of IFC. Therefore, these applications can't develop PA procedures based on the methodology proposed. These issues are irrelevant to the study because they depend on the IFC certification of each software. As tested, every application can solve these issues just by adding some libraries to read the full schema, as well as official IFC visualizers do. The evidence of this statement is supported in this work by the successful editing of IFC files through the *IfcOpenShell* library and the correct and verified visualization of the information in FZK Viewer, as visible in Section 4.

As mentioned, the problem is to gather information about building elements' performances and compile system requirements specifications to verify if the design complies with national codes. The following part of the study shows the methodological problems that arise when translating a discipline into IFC, in this case, the FSE discipline. These are divided into three groups:

- property definition;
- spatial and functional requirements;
- functional and logical representation.

Most of the issues mentioned are related to code-based relations, but it is relevant to analyze IFC capability to map those relations to support model checking.

Fire Safety Engineering discipline is analyzed as a representative example that embodies all of these issues but, on the other hand, highlights how IFC can possibly collect system and building data to support fire safety engineering (FSE) and fire safety monitoring.

3.1. DOMAIN-SPECIFIC PROPERTY DEFINITION

Several studies have highlighted the potential and weaknesses of IFC concerning FSE [6]. Data model analysis shows that *IfcMaterial*'s PropertySets provide thermophysical properties of a building element and fire-specific properties. Instead, the common PropertySets of *IfcBuildingElement* subtypes contain properties defined for regulatory or standardization purposes, such as *Fire Rating*, *FlammabilityRating*, *SelfClosing*, *SmokeStop*, *Compartmentation*, *SurfaceSpreadOfFlame* [7]. On the other hand, the definition of the fire reaction in IFC needs clarification. The *FlammabilityRating* property is only attributed to the class *IfcCovering*, while *FireRating* is attributed to all the *IfcBuildingElement* subclasses.

Furthermore, the property *Combustible* is assigned to elements with a two-dimensional extension. The *FireRating* property can be used to define the resistance to fire, and the *FlammabilityRating* property defines the reaction to fire. The only class which covers all three properties is *IfcCovering*. However, the potential contribution of a product to a fire does not only depend on the covering, but the underlying layers could also influence it. Three scenarios are possible (Fig. 3):

- a generic wall with a structural package is exported through the class *IfcWall*, and an architectural package is exported with the class *IfcCovering*, to which it is assigned the property *FlammabilityRating*;
- a generic wall where the structural part also coincides with the architectural part. An example would be an XLAM wall that constitutes an important fraction of the total compartment area and must be treated as structure (fire resistance) and cladding (reaction to fire). In this case, to map its reaction to fire in an IFC model, it is necessary to create a fictitious layer outside the wall exported as *IfcCovering* to which the *FlammabilityRating* property is associated;
- an internal partition must be exported as an *IfcWall* of type Partitioning. Again, to map the fire reaction of the partition, it is necessary to create fictitious *IfcCovering* layers.

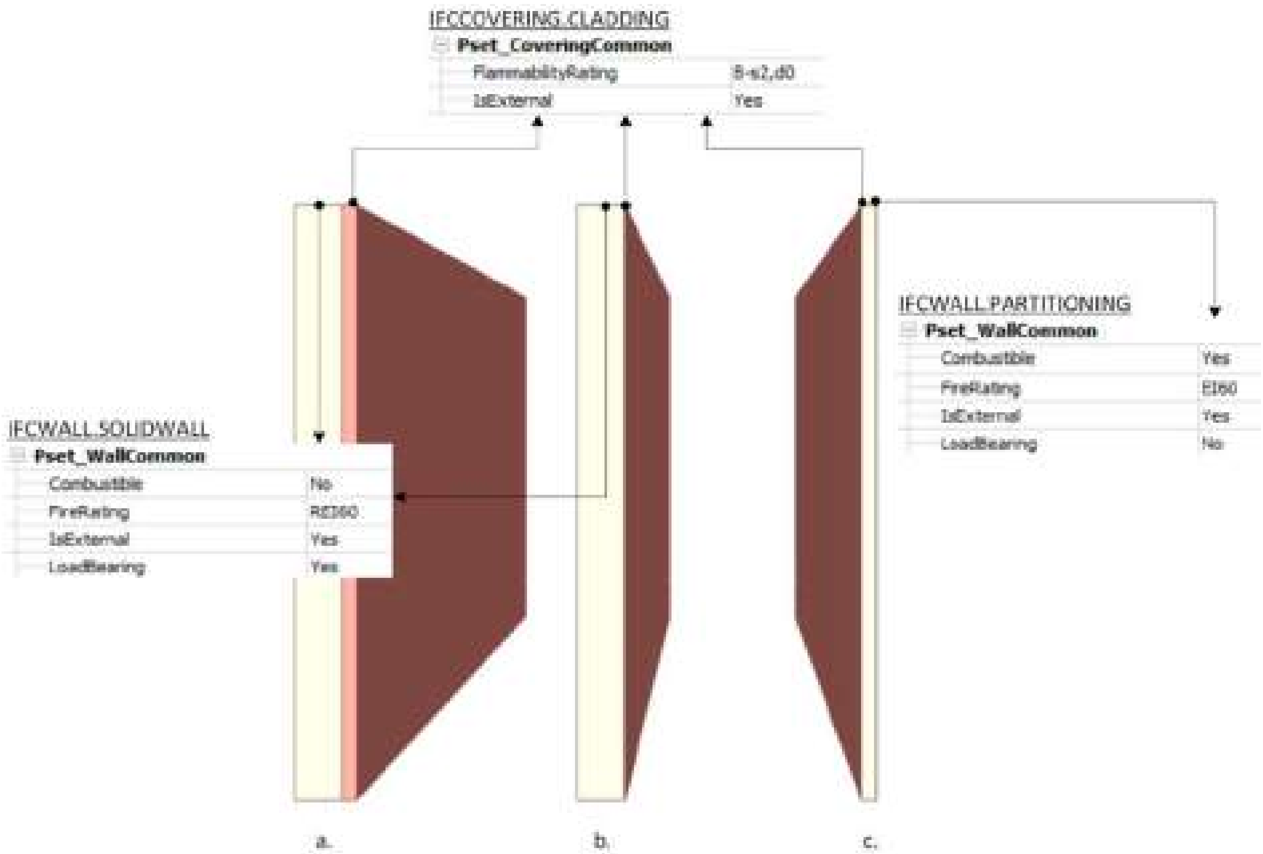

Fig. 3. *Export of Pset_CoveringCommon and Pset_WallCommon.*

The first scenario is the right way of modelling a wall consisting of a structural layer, which generally does not require a reaction-to-fire classification, and an architectural refacing that does.

The problem with the last two scenarios is that they force the modeller to create fictitious entities to map the wall's fire reaction into the model. Ideally, we would like to be able to implement the property set *Pset_WallCommon* with a new parameter to map the fire reaction of the wall as well.

Consequently, IFC development and the disciplinary ontological definition must address domain-specific property definition.

3.2. DETAILED OR RULE-BASED SPATIAL AND FUNCTIONAL REQUIREMENTS

Following the European directives, Italian legislation (Ministerial Decree n. 256, 2019) defines subject activities, which can be contained in a building, part of a building or several buildings.

Presidential Decree n. 151, 2011, contains a list of 80 activities subject to Fire Brigade control. Each activity includes assigned subclasses and risk categories (cat. A, B, and C) based on the severity of the risk rather than the size or degree of complexity of the activity itself.

IFC translation of this kind of entity prefers the entity *IfcZone*, compared to *IfcBuilding*, since it defines «groups of spaces, partial spaces or other zones». The deprecation of the MainFireUse property of *IfcBuilding* class, from the IFC4.1 version, avoids defining the activity category through *IfcBuilding* class and involves the use of the entity *IfcZone* that is therefore specified as a "FireCompartment".

As specified in Building Smart Knowledge base, «in case of a zone denoting a (fire) compartment, the following types should be used, if applicable, as values of the *ObjectType* attribute: '*FireCompartment*': a zone of spaces, collected to represent a single fire compartment. - '*ElevatorShaft*': a collection of spaces within an elevator, potentially going through many storeys. - '*RisingDuct*':

A collection of vertical airspaces. - '*RunningDuct*': A collection of horizontal airspaces».

As mentioned in Section 2.1, this entity can be further and better specified through *IfcClassificationReference* class, thanks to the *IfcRelAssociatesClassification* relationship. By using this class, information regarding the subject activity can be shared with IFC hardcoded attributes.

Similar problems arise when fire compartments have to be digitized. In the FSE discipline, for instance, a fire compartment can coincide with a room, a zone, several parts of a building story or the whole building. These can be exported using IFC*Zone* or *IfcSpatialZone* classes. They share some *PropertySets* like the *Pset_SpaceOccupancyRequirements* with properties such as *OccupancyNumber* or *OccupancyType* and the *Pset_SpaceFireSafetyRequirements* defining, for instance, the risk factor or the equipment of sprinkler systems. On the contrary, the control of the protection level of the compartments is not completed. Currently, no properties can store a compartment-specific fire load calculation. Other weaknesses are typically linked to the Italian context, such as the definition of risk profiles.

The fire risk profiles, *Rlife*, *Renv*, and *Rprop*, are simplified indicators to assess the fire risk as stated by Ministerial Decree n. 256, 2019.

In these cases, it is necessary to introduce a user-defined *PropertySet* containing properties such as *IsSmokeProofCompartment* (of Boolean type), *SpecificDesignFireLoad* (the unit is MJ/m^2 and therefore it will be specified as «UserDefined» since this unit of measurement does not currently exist in IFC), *Rlife*, *Renv* and *Rprop* (of type *IfcLabel*). Obviously, to guarantee interoperability between IFC and BPS (Building Performance Simulation) tools, user-defined PSet needs to be avoided.

3.3. FUNCTIONAL AND LOGICAL REPRESENTATION LIMITS AND ISSUES

Systems modelling and management requirements matching are necessary to map the functional connection of different systems to the zones of the building they serve. Terminals of the piping system for fire suppression are assigned to the *IfcFireSuppressionTerminal* class and further specified through the *PredefinedType* attribute. This class is part of the *PlumbingFireProtectionDomain*. On the contrary, segments composing the piping system for fire suppression are included in *IfcPipeSegment* class which is part of *IfcHvacDomain*. Segments are then grouped in an *IfcDistributionSystem*, which needs to be specified through the enumerative type «*fireprotection*». The system is therefore specified by its reference (*Pset_DistributionSystemCommon*) and its service life attributes (*Pset_ServiceLifeFactors*). *IfcDistributionSystem* class is part of *IfcSharedBldgServiceElements* placed in the interoperability layer of The IFC schema. This means that the translation in information classes of a discipline is not limited to a single domain but needs to collect classes coming from different domains, some of them correctly placed in the interoperability layer and there specified, others part of misleading domains as demonstrated with reference to the *IfcPipeSegment* example.

As well, it is crucial to represent logical connections. In addition to the spatial containment concept template, IFC offers several connections grouped in the object composition template (aggregation or nesting) or the object connectivity template (element connectivity, space boundaries and spatial structure).

Hierarchical representation if IFC does not express logical connections between rooms. This constraint is relevant, for example, in escape route checking, and IFC has no entities and properties concerning the escape route. To overcome the lack of the standard relating to FSE, BuildingSMART launched a project to improve information exchange for Occupant Movement Analysis (OMA) and Fire Dynamic Simulation (FDS), as available online at: https://www.buildingsmart.org/standards/calls-for-participation/fire-safety/.

The design of escape routes has several specific definitions for which the data model is not yet ready. These include dead-end corridors, open, protected, or smoke-proof routes, emergency exits, etc. IFC domain only allows the extrapolation of geometrical information of escape routes (width or length) or the possibility to define which rooms are included in them (with the Boolean property *FireExit*) [6]. This enables the creation of spaces that can be identified as escape routes and then integrated with additional information.

As a result of all these disciplinary limits, the information exchange takes place through manual data entry into specific fire prevention software, e.g., in the definition of Fire Dynamic Simulation (FDS) [7, 8].

This statement points attention to the severe limits to the implementation of IFC as a reference model for fire safety policies and as well as for supporting BPS in FSE. A barrier to implementing the IFC solution is the dependence of engineering procedures on national regulations set on unstandardized assumptions. Hence, the use of IFC can be increased by setting harmonized national standards by higher authorities.

4. IMPLEMENTATION TEST: IFC RELATIONS-BASED FSE CODE CHECKING OF ESCAPE ROUTES

Although the IFC data model has several weaknesses regarding the standardization of escape systems, an approach that relies on assumptions based on the existing IFC dataset was proposed.

Escape routes are unobstructed routes for occupants to reach a safe place and are all part of an escape system made of stairs, corridors, moving walkways, ramps, safe places, exits, doors, safety lighting, signs, etc. The basic idea of this approach is to use the class *IfcSystem* to group all those elements along an escape route through the relation *IfcRelAssignsToGroup* (Fig. 4). All spaces, doors, and stairs belonging to the system will have the boolean property *FireExit* set to *TRUE*. The *IfcRelAssignsToGroup* relation is merely a grouping of elements; then, it avoids the identification of the space sequence along the escape route. This can be achieved through other relations such as:

- *IfcRelSpaceBoundary*, which relates a space (*IfcSpace*) to the physical elements that delimit it, such as a door (*IfcDoor*) (Fig. 5);
- *IfcRelContainedInSpatialStructure* allows for the connection of a physical element, such as a stair *(IfcStair)*, with a space *(IfcSpace)* (Fig. 6).

Escape routes are distinguished according to their degree of protection and compartmentalization concerning the rest of the spaces. They can be protected, smoke-proof, external, or unprotected. Additionally, when entering an escape route with a single directionality, the spaces that make it up must have the same protection level. Identifying escape route segments with the same protection degree is crucial. Indeed, an escape route can lead directly to the emergency exit and another escape route with a higher de-

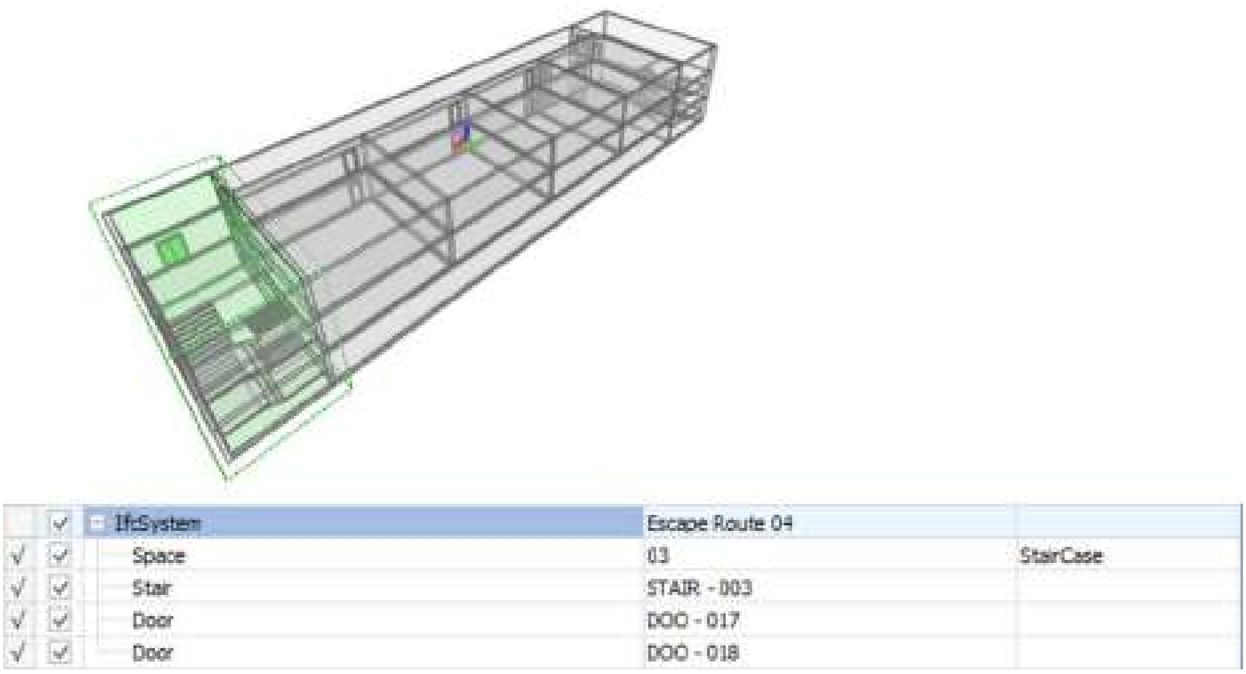

Fig. 4. *Entities grouped in an IfcSystem representing an escape route.*

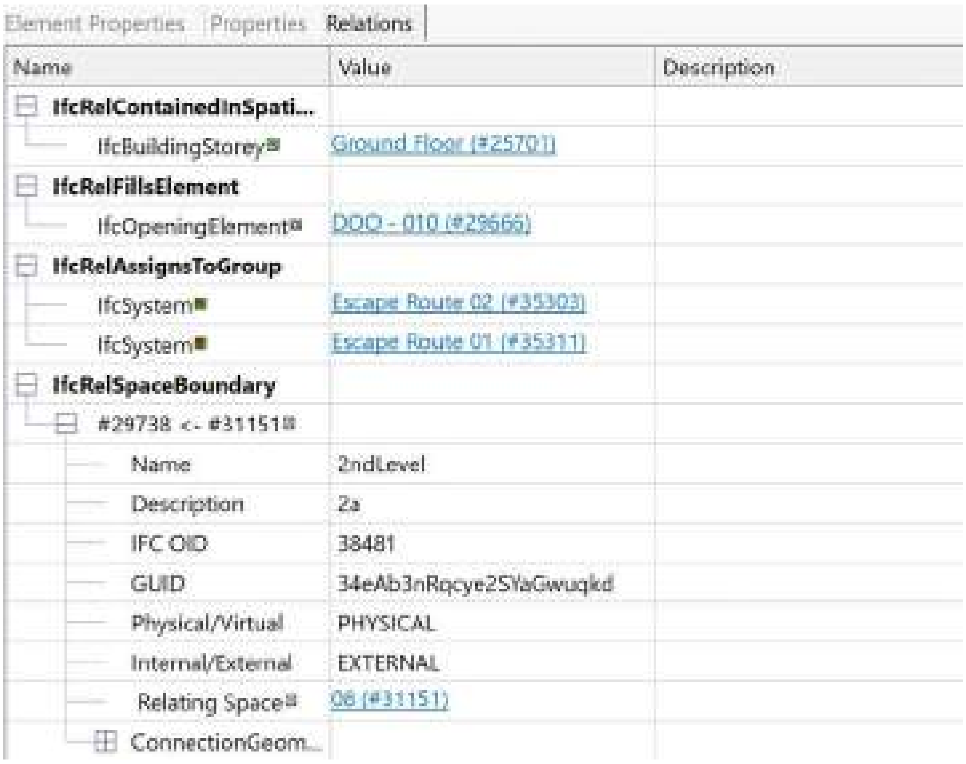

Fig. 5. *IFC relationship assigned to the IfcDoor entity: the data model can express its belonging to different systems.*

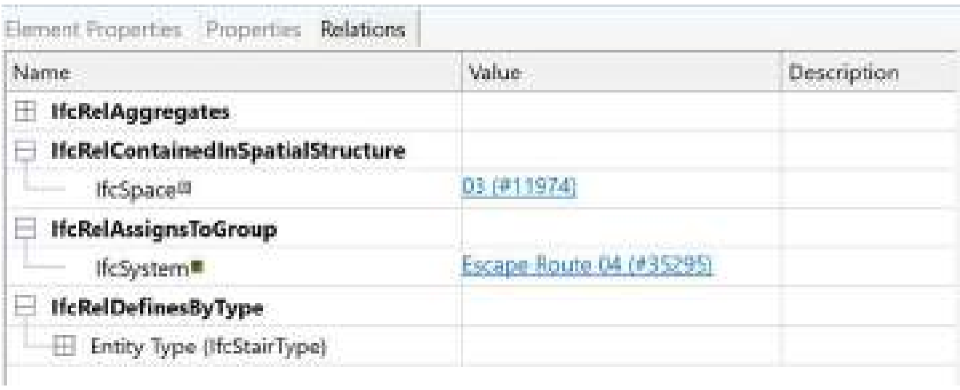

Fig. 6. *IFC relationship assigned to the IfcStair.*

gree of protection. For example, in a multi-storey building, each floor may have a protected corridor leading to a smoke-proof staircase. The approach proposes to create as many *IfcSystem* as there are one-way sections of escape routes with different levels of protection, and an *IfcSystem*, which groups all of them together through the relation *IfcRelAssignsToGroup* (Fig. 7). Besides, the *ObjectType* attribute might help to define whether the escape route is smoke-proof, protected, or unprotected.

Furthermore, the objectified relationship via *IfcRelReferencedInSpatialStructure* can be used to connect a system to the relevant spatial element that it serves. This allows coupling the escape system to compartments (*IfcSpatialZone*) or spaces (*Ifc*Space) (Fig. 7).

Finally, the strategy allows automating the calculation of the escape route length by filling two new properties for the *IfcDoor* entity of a *User-Defined PropertySet* (*Pset_EvacuationRequirements*):

- *PreviousExitLength* is used to define the distance to reach the emergency exit;
- *NextExitLength* is used to specify the length of the next emergency exit or, in the case of a final exit door, the distance to the safe place.

Fig. 7. *IFC relationship assigned to the IfcSystem.*

As previously expressed, the problem seems more related to the higher classes of IFC that host a limited number of property sets and force to outline user-defined ones, but the methodology proposed is consistent.

5. CONCLUSIONS AND FUTURE WORK

Testing of the methodology highlighted the potentialities of IFC and the correctness of the approach in supporting PA procedures. The method is independent of technical implementation issues, as demonstrated by the correct visualization in validation tools of IFC models that are extended with the above-mentioned relations.

This study shows that activating a PA and a code-checking policy in computer-aided engineering requires investment in standardization. Therefore, it is crucial to capitalize on the availability of a consistent data model as it is offered by the standard ISO 16739. On the other hand, severe limits occur in FSE information exchange, and this limit affects a sustainable digital management policy.

The study clarifies that the above limitations in information exchange depend on standardization processes that are not dependent on IFC. IFC could play a significant role in the future to help harmonize existing standards.

In addition, IFC expresses strict limitations as a data model suitable to collect building system performance. As articulated in the research, the higher levels of the building hierarchy have limited "PropertySets". For this reason, collecting the performance data of technical elements in building system performance is problematic. In addition to the lack of best practices shared in literature, this problem leads to a limited culture of PA procedures based on IFC standards, even if it can be considered methodologically consistent.

The development of this work will allow finding procedures and applications to write the IFC database. Currently, some methods are theoretically allowed but not assisted by major software applications. Future work must be dedicated to implementing the methodology in different disciplines to test the robustness of the data model and the procedure.

Acknowledgments

The research is part of UniONE: a project for the digitization of the built Asset of the University of Padua.

Authors contribution

M.G. Donatiello: analysis on IFC data model in relation to spatial containment, functional systems and relation classes. A. Gabbanoto: analysis on FSE code checking based on IFC classes. C. Zanchetta: definition of IFC based procedures, text editing. R. Paparella: project coordinator, text revision.

Funding

Internal funds of the University of Padua.

6. REFERENCES

[1] Ranjbaran Y (2013) Integrated Computational Model in Support of Value Engineering. Doctoral dissertation, Concordia University, Montréal

[2] Gomes Miron L, Kaushik A, Loskela L (2015) Target value design: the challenge of value generation. In: Proceedings of the Title of host publication23rd Annual Conference of the International Group for Lean Construction (IGLC 23), Perth, Australia, 29-31 July 2015, pp 815–825

[3] Gielingh W (1988) General AEC reference model (GARM): An aid for the integration of application specific product definition models. In: Proceedings of the CIB Seminar Conceptual Modeling of Buildings, Lund, Sweden, 25–27 October 1988, pp 165–178

[4] Froese T (2003) Future directions for IFC-based interoperability. Journal of Information Technology in Construction: 231–246

[5] Genova G, Adachi Y (2020) Building Room section 4 - Potential use cases of IFC SpatialZone. https://vimeo.com/483131538. Accessed on April 26, 2023

[6] Spearpoint MJ (2005) Integration of building product models with fire simulation software, UC Research Repository, University of Canterbury, New Zealand. https://ir.canterbury.ac.nz/handle/10092/6050. Accessed on April 26, 2023

[7] Taciuc A, Karlshøj J, Dederichs AS (2016) Development of IFC based fire safety assessment tools, December 18. BIM in Civil Engineering – Open Data Standards in Civil Engineering, pp 60–69

[8] Shi J, Dao J, Jiang L, Pan Z (2019) Research on IFC- and FDS-Based Information Sharing for Building Fire Safety Analysis. Advances in Civil Engineering, November 29:3604369. https://doi.org/10.1155/2019/3604369