Developments in the Built Environment 19 (2024) 100499

Contents lists available at ScienceDirect

Developments in the Built Environment

journal homepage: www.sciencedirect.com/journal/developments-in-the-built-environment

Generation of new BIM domain models from escape route analysis results

Simon Fischer *, Harald Urban, Christian Schranz, Matthias Haselberger, Florian Schnabel

Research Unit Digital Building Process, TU Wien, Karlsplatz 13/E235-03, Vienna, 1040, Austria

ARTICLE INFO

Keywords:
BIM
IFC
Digital building permission
Code compliance checking
Escape route analysis

ABSTRACT

The Industry Foundation Classes (IFC) schema is an open format for BIM models that secures permanent data availability independent of software vendors. This is particularly important for public stakeholders such as building authorities. However, during code compliance checking of the permit process, new explicit data are generated from information implicitly contained in the building models. This information has been available only in the checking application used and not independent of it. This paper presents an approach for generating a new IFC domain model from generated escape route results to be used by the building authority. This was achieved by defining the general concepts of how escape routes can be represented by building models and mapping these to IFC entities. A prototype implementation demonstrated the applicability of this approach. In addition to independent permanent archiving, the IFC domain model for escape routes enables the use of this data during building operations.

1. Introduction

The building industry is undergoing digital transformation. Herein, conventional manual processes are transformed and partially automated using digital data. This transformation is called Construction 4.0 (El Jazzar et al., 2021; Klinc and Turk, 2019). This global trend has also affected public authorities. A widely discussed digitisation topic for building authorities is digital building permits using Building Information Modelling (BIM) (Noardo et al., 2022a). The use of digital building models during the design phase of construction projects is established. However, it is hardly utilised in the building permit process. Typically, building designs are still reviewed and validated based on 2D plans that are available in printed form or as a digital 2D plan (e.g. PDF) (Urban et al., 2024). Adapting to a BIM-based building permit process using digital building models enables many advantages like automatic code compliance checking (Amor and Dimyadi, 2021; Beach et al., 2020; Eastman et al., 2009; Malsane et al., 2015) and dissemination through advanced technologies such as augmented reality (Gerger et al., 2023; Schranz et al., 2021).

An essential aspect of the use of BIM by building authorities is the use of open formats (Schranz et al., 2021; Krischmann et al., 2020). A building authority should provide all applicants with equal opportunities independent of the software they use. Moreover, long-term accessibility of data must be ensured. Building submission documents are legal documents that are subsequently granted and therefore, must be accessible in the future. The conventional approach of using printed plans provides safe long-term accessibility. The plans can be archived in libraries and remain readable. Digital plans in PDF format also provide good long-term accessibility because it is a standardised format. This implies that accessibility does not rely on particular software or vendor.

For BIM-based permissions, the equivalent format could be Industry Foundation Classes (IFC). It is an ISO-standardised schema (International Organization for Standardization, 2024) that has been used in many digital building permit initiatives, such as in Austria (BRISE-Vienna) (UIA – Urban Innovative Actions, 2019), Estonia (Ullah et al., 2022), and South Korea (Kim et al., 2020). Thus, all explicitly modelled information can be accessed securely in the future. However, in code compliance checking during the building permit process, decisions on the information implicitly contained in the models are also made. An example is an escape route. Escape routes are generally not modelled explicitly. Rather, these are generated from information implicitly available in the building models and validated subsequently. The results of these checks, which generate new data, exist only in the checking applications used. Thus, their accessibility depends on the availability and presence of the checking application and its compatibility with older versions. Because these elements are approved in the building permission, their data also need to be stored permanently.

Therefore, the introduction of a new IFC domain model for building authorities containing generated escape route data is the subject of this study. An IFC model can permanently store generated permitted objects

* Corresponding author.
E-mail address: simon.fischer@tuwien.ac.at (S. Fischer).

https://doi.org/10.1016/j.dibe.2024.100499
Received 3 April 2024; Received in revised form 2 July 2024; Accepted 3 July 2024
Available online 14 July 2024
2666-1659/© 2024 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

in the same way as the initial submitted model. This ensures that building authorities can meet the long-term data archiving requirements for permitted objects. Furthermore, the building authority gains access to a more comprehensive information source than that provided by the 2D plan. The generated objects can include additional characteristics that are not always visible on submission plans, such as the number of people using a route. Introducing an individual domain model for the generated escape route information allows for a clear separation of the submitted architectural model. The submitted architectural model represents the application and should not be edited by the building authority. However, the escape route domain model can still be included in subsequent workflows as a reference, since existing software applications can process and visualise it. This enables seamless integration in the BIM workflow and is advantageous compared with other open formats such as JSON or XML. The model can be used for permanent storage and as an input in other BIM use cases during the building lifecycle.

This study focused on the requirements and technologies for generating a new domain model for escape route data. We retrieved the requirements for the model from the building code and escape route checking results generated by Fischer et al. (2023). These requirements are mapped using a suitable IFC representation. We implemented the developed concept using two programming languages and validated the results on two BIM models of BRISE-Vienna. Finally, conclusions were drawn from the results. These steps are based on the preliminary work of the co-authors Haselberger (Master's thesis (Haselberger, 2023)) and Schnabel (Bachelor's thesis).

2. Literature review

Two research fields are decisive for the permanent storage of escape route data in the digital building permit process: digital building permission and escape route analysis. Digital building permits have recently become an important topic in research and practice. This is evident from the founding of the European Network for Digital Building Permits (EUnet4DBP, 2024). They unite scientists and stakeholders to develop a common strategy for digitising building permits. Digital building permission is a complex process in which various stakeholders interact at different steps. This includes preparing and submitting the required documents, checking code compliance, and presenting and visualising the results to the relevant parties.

In addition to pioneering initiatives such as CORENET in Singapore (Hjelseth, 2015; Shahi et al., 2019; Kaneta et al., 2016), where BIM permission has been applied for several years, other countries recently introduced the topic of digital building permission. These include Estonia (Ullah et al., 2020, 2022), South Korea (Kim et al., 2017, 2019, 2020; Lee et al., 2016; Park et al., 2016; Song et al., 2019), and Austria (BRISE-Vienna) (Gerger et al., 2023; Schranz et al., 2021; Krischmann et al., 2020; UIA – Urban Innovative Actions, 2019).

In 2022, Noardo et al. (2022a) surveyed digital building permit research to reveal recent focus fields. They clustered the literature into different steps of the digital building permit process. They observed that code compliance checking as well as interpretation and the digitisation of regulations were addressed the most. Examples of studies on code compliance checking are presented in (Amor and Dimyadi, 2021; Eastman et al., 2009; Battisti et al., 2022; Noardo et al., 2022b). The interpretation of regulations written in natural language and translation into a computer-interpretable format is addressed in (Malsane et al., 2015; Kim et al., 2019; Han et al., 1998; Lee, 2011; Macit İlal and Günaydın, 2017; Nisbet et al., 2008; Uhm et al., 2015).

However, not mentioned by Noardo et al. (2022a) is handling newly generated data in the building permit process, such as escape routes. The concept of using IFC for the permanent storage of generated data is hardly on the map of digital building permit research. An exception is the work of Ciccone et al. (2023) on data preparation for e-permissions. They addressed the problem of the results of structural safety assessments using a BIM model as input are not being returned to the BIM environment. If a structural safety assessment is required for permission, the documents are submitted additionally to the BIM model. Therefore, they used IFC to represent the calculation results as digital model. The concept was validated by implementing a prototype that generates an individual IFC model to be submitted during the permission process. The concept of information representation with IFC also displays potential for the permanent storage of data generated during code compliance checking. This applies to escape route analysis.

In the field of BIM-based escape route analysis, research has focused on the automated assessment of escape possibilities. This includes the automated generation and checking of escape routes and the simulation of pedestrian flow. The basis is an indoor navigation model representing indoor circulation. The definition and generation of an indoor navigation model suitable for representing the course (geometry and connectivity) of escape routes has been addressed by various researchers (Fischer et al., 2023; Borrmann et al., 2012; Fu et al., 2020; Höcker et al., 2010; Kannala, 2005; Kneidl et al., 2012, 2013; Lee et al., 2010; Lin et al., 2013; Lin and Lin, 2018; Liu et al., 2021; Taneja et al., 2016; Yang and Worboys, 2015; Yuan and Schneider, 2010). A few of them also applied path-finding algorithms to generate the course of escape routes automatically (Fischer et al., 2023; Kannala, 2005; Lee et al., 2010; Lin et al., 2013) or simulate pedestrian flow (Borrmann et al., 2012; Höcker et al., 2010; Kneidl et al., 2012, 2013). The building-code compliance of the generated routes is included in (Fischer et al., 2023; Kannala, 2005). Kannala (2005) focused on universal building code requirements for escape routes derived from different fire codes in Northern America and Finland. Fischer et al. (2023) considered the escape route checking process of the Vienna Building Authority as part of the BRISE-Vienna project (UIA – Urban Innovative Actions, 2019). They developed a checking rule that generates and validates escape routes for the code compliance checking system of BRISE-Vienna. The project BRISE-Vienna included the BIM-based transformation of the entire permission process of the Vienna Building Authority (Gerger et al., 2023; Schranz et al., 2021; Krischmann et al., 2020).

However, the aforementioned research projects did not consider further processing of analysis results. Which parts of a generated escape route network are required for further use and how these can be stored are questions that remain to be addressed. This study used the results of Fischer et al. (2023) to address this research deficiency and generated a permanent IFC representation with the relevant escape route data.

3. Material and methods

The concept of creating an escape route model in the IFC format for the permanent storage of escape route data is part of the overall escape route permitting process developed in BRISE-Vienna (see Fig. 1). Generation of escape route models for building authorities requires relevant data on escape routes. Fischer et al. (2023) developed the concept of generation and validation for creating and assessing escape routes in compliance with the legal system in Vienna. Escape route generation is a tool used by planners to generate legally compliant escape routes from implicitly contained information in BIM models. Escape route generation includes algorithms for the general steps of escape route analysis described in Section 2: indoor navigation model creation, path finding, and element checking. From the collection of potential escape routes, planners can select the desired network of escape routes for the entire building. Subsequently, escape route validation evaluates the entire network. Planners use this concept before submission, and the building authority does so after. The result of the validation is a code-compliant network of escape routes. These explicit data are the input for escape route model generation, which generates a new permanent IFC domain model from the previous analysis results.

Introducing a suitable new domain model requires various steps. The first is an analysis of the required information. Subsequently, we define the general concepts for representing escape routes and map these using

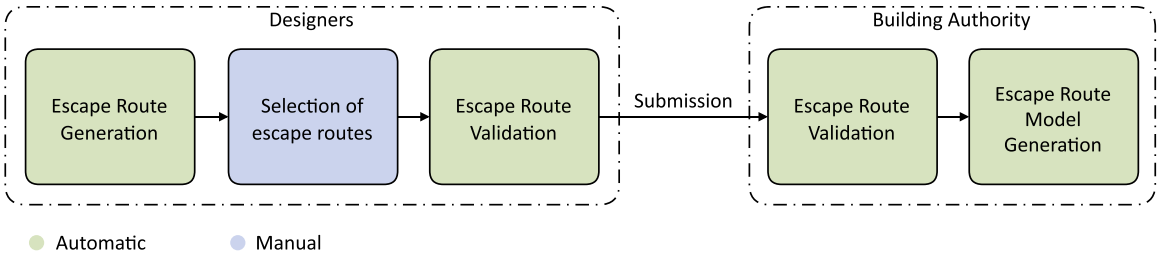

Fig. 1. Overall escape route permitting process (adapted from Fischer et al. (Fischer et al., 2023)).

the IFC schema. With a formal definition, we can implement algorithms to generate IFC escape route models. Finally, two test models were used to validate the results.

3.1. Analysis of requirements for a digital representation

The first step in generating an escape route model is to define the information required to represent the escape routes. To achieve this, it is essential to understand the definition of escape routes in the context of the building permit process and its requirements. In Austria, the definition of an escape route and its requirements are provided in the OIB guidelines (Austrian Institute of Construction Engineering, 2023a), which are nationally applicable building regulations. The OIB 2 (Austrian Institute of Construction Engineering, 2023b) for general fire protection defines the requirements for the course of escape routes. Escape routes are the paths from the required starting locations to a safe space outside the building. In general, an escape route starts at the most remote point from the room exit. An exception is defined for flats with a maximum of two levels without a direct exit to the outside. Such flats require only one escape route for all the rooms, starting from a flat entrance door. Each escape route ends in a safe space outside the building. However, there are safe spaces inside buildings (such as emergency staircases) with additional fire safety requirements. Such safe spaces are interim destinations for escape routes. This is because the path through the safe space to the outside (the actual destination of the escape route) is not part of the permitted route length. Consequently, the permitted route length is not necessarily equivalent to the geometric route length.

The submitted models used in the building permit process do not include elements representing the described course of the escape routes. However, these implicitly include a large amount of information that can be used to derive escape routes, such as the geometry of a building and information regarding space usage. Fischer et al. (2023) developed an algorithm to generate escape routes with a course and geometry according to the building code requirements from the submitted models as input. An escape route is considered as a 3D polyline from the start to the end location. This polyline consists of a sequence of straight lines (called segments) between the nodes. The nodes represent the 3D coordinates of the points of interest (starting point, endpoint, and variation in direction). To generate a route, a sequence of nodes is connected by segments. Inside the rooms, the escape route adopts the shortest path connecting any two points. This implies that the polyline nodes are located directly at the room-bounding corners. Doors are passed through their centres perpendicular to the wall. On the stairs and ramps, the escape route follows the centreline. The curves of the centreline are approximated using polylines. The z-coordinates of the nodes are defined by the elevation of the surface to which the nodes belong (floor level in rooms and tread nose on stairs).

In addition to the exact geometry, a permanent digital representation of escape routes should contain relevant information to validate the escape routes. Therefore, it is essential to distinguish between the requirements of each escape route and the combined network of escape routes. Each starting location has a separate escape route that must reach a safe location within the maximum route length. During an emergency, a certain number of people can escape from a starting location. Several escape routes overlap in public circulation areas such as corridors and staircases. Many individual escape routes generate a common flow of people. This network of escape routes must fulfil the width requirements, which depend on the cumulative number of people from different escape routes passing through a particular section. Such width requirements exist for rooms, doors, stairs, and ramps along escape routes, as defined in the OIB 4 guideline (Austrian Institute of Construction Engineering, 2023c). The required height of the escape route is constant, but it is possible to partially narrow the passage profile.

To conclude, the escape route representation should satisfy the following requirements:

- geometric representation of each escape route,
- assignment of a route length for each escape route, and
- assignment of the cumulative number of people for segments of the escape routes.

These general requirements are also applicable to the representation of escape routes in other countries and regions. A brief review of building codes from across the globe was conducted, including the Template Building Code of Germany (Fachkommission Bauaufsicht zur Musterbauordnung und Muster- Beherbergungsstättenverordnung, 2022), building regulations of England and Wales (HM Government, 2022), the Code of Practice for Fire Safety in Buildings of Hong Kong (Buildings Department, 2023), the New York City Building Code (City of New York, 2022), and the International Building Code (International Code Council, 2021). All of the aforementioned regulations define escape routes as connections with a specific geometry between locations with occupancy and safe exits to the outside. As in Austria, the width of escape routes is always dependent on the cumulative occupant load. The height remains constant at a given value. However, we also found statements that allow protruding elements inside the headroom of escape routes, if they do not exceed specific dimensions (City of New York, 2022; International Code Council, 2021). Furthermore, the maximum length of escape routes is also restricted, and, as in Austria, is measured either to a direct exit or a safe space inside the building. With regard to the international context, it can be seen that the fundamental requirements retrieved from the Austrian building regulations are generally applicable. Consequently, in the next steps, we will focus on implementing them in an IFC model.

3.2. Developing general concepts for the representation

The second step was to identify and analyse the general concepts for modelling escape routes as new geometric objects in the BIM model. An escape route network can be observed from two perspectives: route level

and segment level. At the route level, an escape route network combines individual routes that start and end at certain places and have a defined length. At the segment level, an escape route network comprises several individual segments used by a specific number of people. The route level is suitable if the courses of particular escape routes are of interest. The segment level is noteworthy for the overall flow of people. These two perspectives are related because the segments are part of the routes, and the people of a segment are derived from the routes. We determined three concepts to represent escape routes based on the perspective from which these could be considered.

The concept of *Routes as Objects* corresponds to the route-level perspective. Each route is represented by a geometric object from its start point to its end point (see Fig. 2). All the segments of each route are unified into an object. The composition of escape routes from individual segments is not represented. The object can only contain information that applies to the entire route, such as the route length or the number of people at the start of the route. The routes along the common paths are not related. The routes overlap in these sections. This concept is a simple approach to representing individual escape routes from different starting locations and illustrating their course. However, because of the missing relationship between different routes, information regarding the entire route network (e.g. the cumulative number of people in a specific segment) is lost.

The concept of *Segments as Objects* corresponds to the segment-level perspective. Each segment is represented by an individual geometric object from one node to another (see Fig. 3). The overlapping segments with identical geometric datasets are combined to prevent replication. This enables summing up people from different routes using the same segment and obtaining a cumulative number of people per segment. This concept can illustrate the flow of people. However, removing replicates results in a loss of information on the relationship between segments and routes. Therefore, the routes cannot be reconstructed from the combined segments. For a specific segment in the middle of the escape route network, the starting positions from which the people come cannot be determined. Thus, the lengths of the individual routes are not provided.

The concept of *Routes group Segments* combines previous concepts. Therefore, it also contains the relationship between the two perspectives. Identical to the concept of *Segments as Objects*, only one combined geometrical object with a cumulative number of people exists for each segment. Additionally, routes are interpreted as containers that group individual segments to restore the relationship between different segments to one route (see Fig. 4). This results in geometric objects for segments and abstract objects for routes. Both these contain information. Defining segments as sub-objects of routes enables metadata to vary along the route while maintaining the relationship between the segments and routes. Thus, the concept enables the representation of segments with their cumulative number of people and the identification of individual routes with their route lengths. This combines the advantages of the previous concepts. However, including the relationship between these two perspectives increases the complexity of the model.

With regard to the requirements identified for an escape route representation for building permits (geometric representation, route length, and cumulative number of people per segment), only the concept *Routes group Segments* contains all the necessary information.

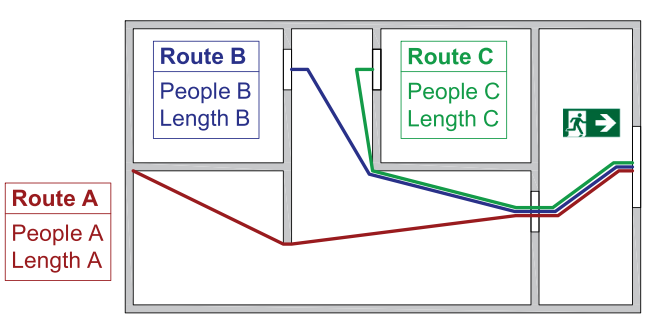

Fig. 2. Concept *Routes as Objects* with individual objects for each route.

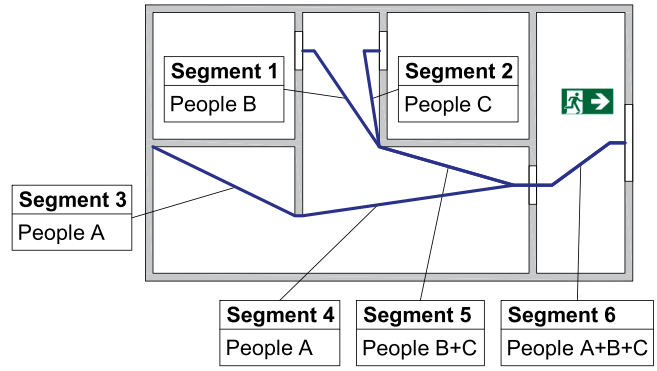

Fig. 3. Concept *Segments as Objects* with individual objects for each segment.

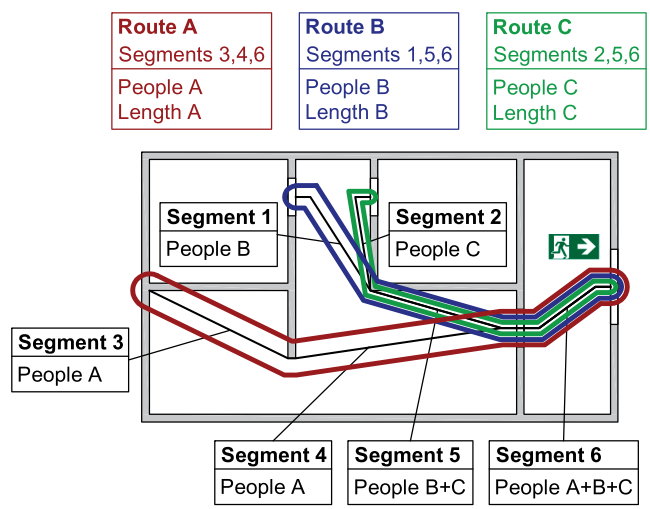

Fig. 4. Concept *Routes group Segments* with individual objects for each segment and abstract objects that group these to routes.

3.3. Mapping of the concepts to the IFC schema

The escape route model development aims to provide data permanently in a form suitable for integration into an openBIM workflow. This can be realised by mapping the identified general concepts in a globally distributed open format for BIM models: the IFC format (International Organization for Standardization, 2024). Therefore, we analysed the IFC schema to identify suitable entities. We used the ISO-standardised IFC4 schema (IFC4.0.2.1) as a basis. However, we also considered new developments in the IFC4.3 schema (IFC4.3.2.0) to integrate upcoming solutions. The escape route model requires IFC mapping for the following three aspects:

- single elements for segments and routes with a defined geometry and alphanumeric properties,
- the relationship between segments and routes, and
- the integration of the elements in the spatial structure of the building model.

3.3.1. Mapping of single elements and geometric representation

The IFC schema does not provide particular entities to represent escape routes or parts of escape routes, such as for walls or doors. However, there are some entities that are suitable for mapping both entire escape routes and escape route segments. Which of both are represented by them depends on which of the previous concepts is used. For example, for *Routes as Objects*, they represent entire routes, whereas for *Segments as Objects*, they represent individual route segments.

An escape route can be considered as a spatial object or zone inside a building with a constant height and variable width (dependent on the number of people). A suitable IFC mapping for this is the entity IfcSpatialZone. According to IFC documentation (buildingSMART International, 2024b), an IfcSpatialZone can describe any zone such as a thermal or useable area zone. The second complies with an understanding of escape routes as spatial objects. A predefined type called transport exists for areas that are mainly used for moving people or goods (buildingSMART International, 2024a). Another option is the entity IfcBuildingElementProxy. An IfcBuildingElementProxy is intended for physical building elements that do not correspond to a particular entity, such as IfcWall. However, these are also used for space provision, which is similar to escape routes. A more abstract option is IfcAnnotation. It is a graphical representation such as a note or a symbol that adds meaning to a building model. Annotation is a highly abstract mapping for an escape route. However, it can include geometry and contain properties in property sets, such as entities for physical and spatial objects.

Regardless of the selection of one of the entities described, geometric representation works according to the same principle. The most straightforward geometric representation of an escape route or segment is based on its axis. The axis is defined as a sequence of nodes connected by straight lines. Only two nodes are required for the segments. The IFC schema can represent such an axis using the entity IfcPolyline. The geometric representation of a polyline corresponds to the creation of escape routes in the escape route analysis by Fischer et al. (2023). However, as mentioned above, an escape route can also be considered as a spatial object with a specific shape. The dimension requirements of building regulations define this shape. In the Austrian OIB guidelines, for example, the profile of an escape route is defined as a rectangle with a particular height (2 m) and width dependent on the number of people using the escape route. Furthermore, building regulations allow the narrowing of escape routes by protruding elements, provided that they do not extend beyond a certain limit (Austrian Institute of Construction Engineering, 2023c; City of New York, 2022; International Code Council, 2021). The precise spatial shape of an escape route could prove beneficial in the validation of such protruding elements. However, the Vienna Building Authority requires escape routes to be represented by polylines for permission. Therefore, the exact three-dimensional shape (volume) of the calculated escape routes is not provided by the escape route analysis (Fischer et al., 2023). To demonstrate shape creation and comply with the requirements of the Vienna Building Authority, the shape of an escape route segment is approximated using a cylinder. To generate the cylinder, a profile is created using IfcCircleProfileDef. IfcExtrudedAreaSolid extrudes a solid along a defined direction from this. The axis and cylinder together form the IfcProductDefinitionShape of the segment. Fig. 5 shows the entities required to create a geometric representation.

The placement of the segments is defined by a local coordinate system. The escape route points are provided by global coordinates by Fischer et al. (2023). From these, the origin of the segments' local coordinate system is defined. The orientation of the local coordinate system is given by the direction to the next point, which is used as the z axis. The x and y axes are defined to span a plane orthogonal to the z direction. The local coordinate systems of the segments are displayed in Fig. 6.

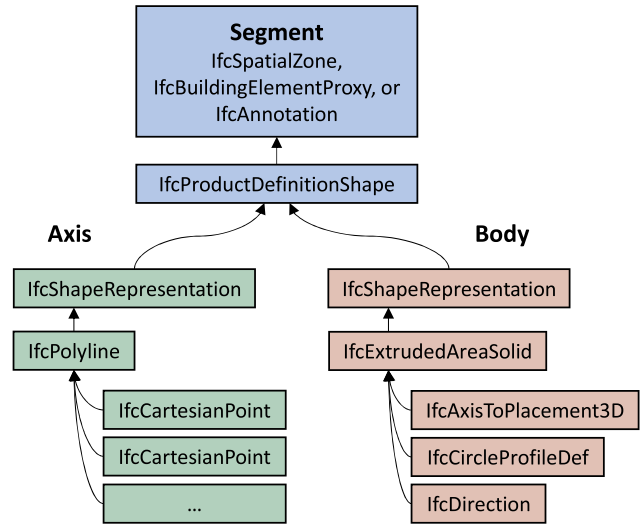

Fig. 5. IFC entities required to create a geometric representation with an axis (left) and a body (right).

3.3.2. Mapping of the relationship between segments and routes

Although the described entities provide a sufficient means to realise the concepts of *Routes as Objects* and *Segments as Objects*, the concept of *Routes group Segments* requires a relationship between segments and routes.

An option is to represent segments as individual elements and add an additional property referencing the escape routes to which the segments belong (*Routes group Segments–Property*). Thus, a checking tool can be used to reconstruct escape routes. However, because no additional elements exist for escape routes, information regarding the routes cannot be saved as properties, such as the escape route lengths. In addition, this characteristic cannot be calculated from the segment lengths. Owing to safe zones inside buildings, the relevant escape route length is not equal to the geometric length of all the route segments.

Another option is to use additional elements or container elements to group sub-elements (*Routes group Segments–Part*), as is common in IFC for building elements. The sub-elements are created using the IfcBuildingElementPart entity. The relationship IfcRelAggregates combines these parts and assigns these to a building element. A typical example is modelling the layers of a wall as IfcBuildingElementParts and aggregating these into an IfcWall. IfcRelAggregates is a relationship with one-way exclusivity. This implies that parts can only be assigned to one container. This significantly constrains the representation of escape routes. Because we wish to prevent duplicate segments, segments should be assigned to several groups.

An alternative is to use IfcGroup as a container element for routes (*Routes group Segments–Group*). Segments (e.g. IfcSpatialZone or IfcBuildingElementProxy) can be assigned to routes (IfcGroup) using the IfcRelAssignsToGroup relationship. A crucial characteristic of IfcGroup and IfcRelAssignsToGroup is the feasibility of assigning elements to several groups. Thus, the shortcoming of the IfcRelAggregate relationship is solved, and geometrically identical segments can be combined into an object that is part of different escape routes. The group does not have a shape representation but combines the shapes of the segments. Assignment of properties is feasible as for any other physical or spatial object. Fig. 6 shows the combination of segments into an IfcGroup and the assignment of properties for the segments and the route.

Both relations, namely IfcRelAggregates and IfcRelAssignsToGroup, represent a non-ordered set of elements. Thus, the order of segments in a route is not semantically given. However, it can be reconstructed from the topological location of the start and end points of the individual segments.

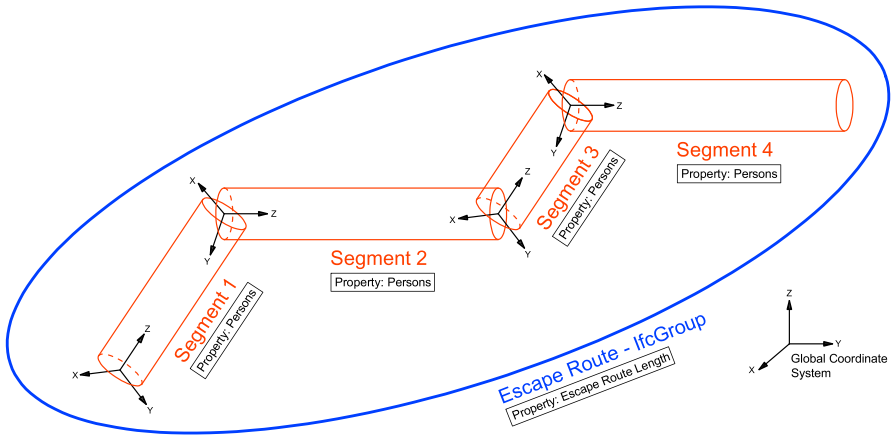

Fig. 6. Combination of segments to one escape route for the concept *Routes group Segments–Group*.

3.3.3. *Mapping of the integration into the spatial structure*

Finally, the elements created for the segments and routes are included in the spatial structure of the IFC model. With regard to building elements, it appears practical to relate segments to the building storey in which these are located. Thus, it is feasible to filter the person flow per building storey. Meanwhile, routes generally extend over several building storeys. However, for a route, a critical information is the starting location. Therefore, it is practical to relate these to the storey from which they start (see Fig. 7). This enables the filtering of all routes starting from a particular building storey.

The IFC schema provides two feasible relationships to realise the assignment to the spatial structure. IfcRelContainedInSpatialStructure assigns elements to the specific level of the spatial structure in which these are primarily contained. This represents a stringently hierarchical relationship. An object can only be contained in one spatial structure element. It is used to contain elements at one level of the spatial structure, such as walls in building storeys. IfcRelReferencedInSpatialStructure is a spatial relationship that assigns elements to the spatial structure in addition to the level in which these are primarily contained. This relationship is non-hierarchical. This implies that an object can be referenced at different spatial levels (e.g. building storeys). A common use is to reference curtain walls spanning multiple storeys.

In general, primary containment using IfcRelContainedInSpatialStructure is suitable for relating segments to the building storey in which these are located and routes to the building storey in which these start. An exception is the use of IfcGroup. The IFC schema does not consider an IfcGroup as a physical or spatial element and therefore, prohibits the use of IfcRelContainedInSpatialStructure. In IFC4, IfcGroup cannot be assigned to the spatial structure. Only the objects in the group can be assigned and not the group. IFC4.3 introduced the use of IfcRelReferencedInSpatialStructure for IfcGroup. This enables the referencing of escape routes, defined as IfcGroups, in the spatial structure. Fig. 8 shows the entities necessary to combine segments into groups and assign these to the spatial structure.

3.4. *Software development*

The software development aimed to provide a proof of concept by implementing a prototype application. The previous theoretical description and discussion of the concepts had already revealed their strengths and weaknesses. While the concepts *Routes as Objects*, *Segments as Objects*, and *Routes group Segments–Property* lack the necessary information, they comply to established IFC standards and are, therefore, rapidly implementable in current practice. The concepts *Routes group Segments–Part* and *Routes group Segments–Group* do not fully comply with the IFC standard, at least for IFC4, but can contain the necessary information. These theoretically described strengths and weaknesses are to be validated by the software implementation to provide a reliable

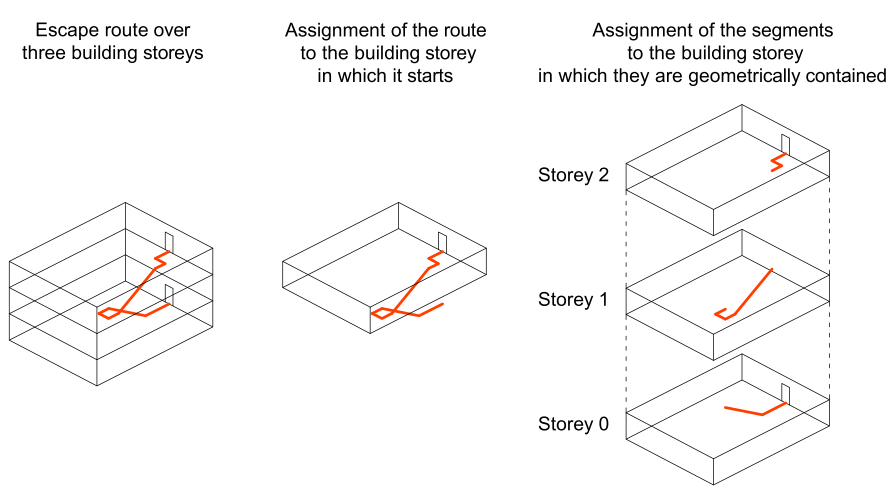

Fig. 7. Assignment of routes and segments to the spatial structure of the model.

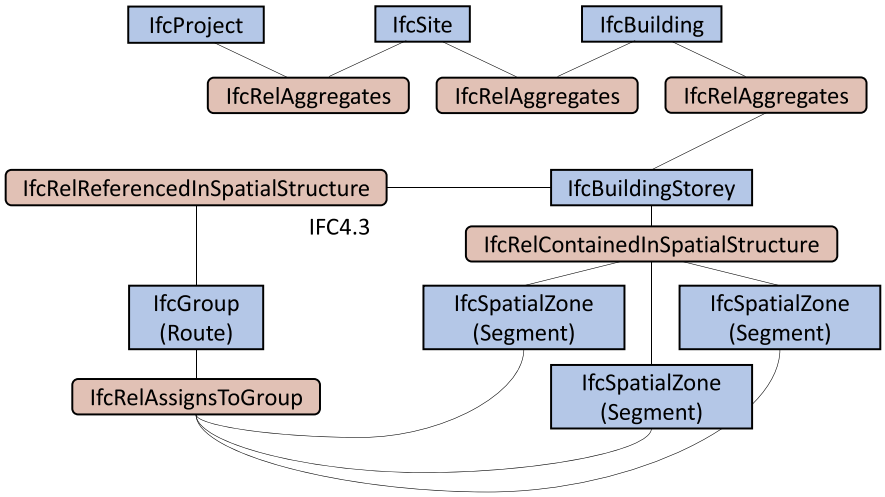

Fig. 8. Combination of entities to create the spatial relationship for the concept *Routes group Segments–Group*.

Table 1
Use of Ifc entities per variant for the implementation.

Variant	Entity for Routes	Entity for Segments
Routes as Objects	IfcAnnotation	–
Segments as Objects	–	IfcSpatialZone
Routes group Segments–Property	–	IfcSpatialZone
Routes group Segments–Part	IfcBuildingElementProxy	IfcBuildingElementPart
Routes group Segments–Group	IfcGroup	IfcSpatialZone

assessment. Therefore, we implemented one prototype for the concepts *Routes as Objects*, *Segments as Objects*, and one for each of the three relationship mappings between segments and routes of the concept *Routes group Segments*: *Property*, *Part*, and *Group* (see Section 3.3.2). For *Routes group Segments–Part* we have omitted the exclusivity of the containment to avoid redundant elements.

Each of the five prototype variants technically enables the use of three identified IFC entities for individual elements (IfcSpatialZone, IfcBuildingElementProxy, and IfcAnnotation). To demonstrate this, we used different entities for individual elements to implement the variants, as listed in Table 1. Note that IfcAnnotation, IfcSpatialZone, and IfcBuildingElementProxy can be substituted. However, although not explicitly constrained to the IFC schema, the IFC documentation indicates that IfcBuildingElementPart should only be part of a building element. This applies only to the IfcBuildingElementProxy of the three considered entities. Therefore, IfcBuildingElementProxy was used for the *Routes group Segments–Part* variant. IfcAnnotation is the simplest element mapping method. It was used for the simplest variant, *Routes as Objects*. IfcSpatialZone was used for the other three concepts. This was because it is best suited for the element mapping of escape routes.

We developed prototypes using the Python and JavaScript programming languages. The Python implementation was realised using the Ifc-OpenShell library (version 0.7.0) (IfcOpenShell, 2024). This provided a broad concept portfolio for processing IFC data. In JavaScript, the web-ifc library (version 0.0.42) (That Open Company, 2024) provides similar prospects. Both the libraries used are open-source. Enunciation 1 shows the code for creating an object as IfcSpatialZone using IfcOpenShell and web-ifc. Both provide functions with parameters that represent the attributes of the IFC entity. These functions can be used for any IFC entity. A difference is that web-ifc requires the additional provision of expressID as a functional parameter. In IfcOpenShell, the expressID is assigned in the background. However, both the packages can be used to generate new IFC models. We used IfcOpenShell as the basis for our implementation and created prototypes for all five variants. In web-ifc, we focused on the two most potential and elaborate variants: *Routes group Segments–Part* and *Routes group Segments–Group*. The implementation using two different open-source packages aims to validate the independence of the concepts from a particular software package. The availability of different implementation options is crucial for the building authority to avoid limitations.

Listing 1. Functions to create an IfcSpatialZone object in IfcOpenShell and web-ifc and the corresponding output in the STEP physical file.

```
IfcOpenShell:
ifcfile.createIfcSpatialZone(guid, owner_history, name, description,
            object_type, object_placement, representation, long_name, predefined_type)
STEP physical file:
#102=IFCSPATIALZONE('1FrsP2PkHDIOwgh8oIsmAw',#5,'EscapeRouteSegment1',$,$,#101,#96,$,$);

web-ifc:
new WebIFC.IFC4.IfcSpatialZone(expressID, guid, ownerHistory, name, description,
            objectType, objectPlacement, representation, longName, predefinedType)
STEP physical file:
#121=IFCSPATIALZONE('m3zdL2zKH8QJZgF_zafUPp',#12,'EscapeRouteSegment01',$,$,#115,#120,$,$);
```

```
[
1 {
   "Elevation": 156.68, "Sitename": "ExampleSite",
   "Latitude": {"Degrees": 48, "Seconds": 0, "Minutes": 13, "MicroSeconds": 0},
   "Longitude": {"Degrees": 16, "Seconds": 0, "Minutes": 22, "MicroSeconds": 0}
  },
2 { "ExampleBuilding": {
    "relativePlacement": [0.0, 0.0, 0.0], "Elevation": 0.0
   }
  },
3 {
   "Floor -1": {
    "Building": "ExampleBuilding", "relativePlacement": [0.0, 0.0, -2.89], "Elevation": -2.89
   },
   "Floor 0": {
    "Building": "ExampleBuilding", "relativePlacement": [0.0, 0.0, 0.0], "Elevation": 0.0
   },
   ...
  },
4 {
   "Path": "Space Büro[A_1_F00_A02_02], 1. Path",
   "StartBuildingStorey": "Floor 0", "Length": 12.23450112387901, "Persons": 54.0,
   "Nodes": [
    [2.8519426384008306, -9.614670654293262, 0.0],
    [6.201943019870557, -2.814670225139819, 0.0],
    [6.201943019870557, -2.664670129772387, 0.0],
    [3.187800526618957, -0.028812259435653693, 0.0],
    [2.187800526618957, -0.028812259435653693, 0.0],
    [-11.092198848724365, -1.3366940617561354, 0.0],
    [-11.592198848724365, -1.3366940617561354, 0.0]
   ]
  },
  ...
]
```


Fig. 9. Excerpt of the input JSON file showing the information for site (1), building (2), two building storeys (3), and one path (4).

3.5. Validation

In this study, two building models from the BRISE-Vienna project were used to generate new escape route models. Both are publicly available and were used in Fischer et al. (2023) to validate the developed escape route analysis. One is a custom test model for escape route analysis (Fischer et al., 2024a) that includes test cases for the characteristics of escape routes (e.g. different feasible routes and door dimensions). This served to validate the general functionality of the escape route analysis. The second model is a real-world model (Fischer et al., 2024b) that was provided during the pilot phase of BRISE-Vienna.

The required information from the building models and escape routes was exported by the developed escape route analysis into a JSON file (see Fig. 9). The first three elements in the file are designated for information regarding the IfcSite, IfcBuilding, and IfcBuildingStoreys of the existing project. These were required to create a reference for the final escape route model. The other elements represent escape routes. The escape route information was divided into metadata and geometric data. Metadata includes the name, length, number of people, and floor from which the escape route originates. The geometric data consists of a sequence of 3D coordinates representing the course of the escape route. With this input, we regenerated the escape routes of the two building models, located these correctly, and finally overlaid the escape routes and original models.

The final step was to validate the results. Three criteria were used to assess the quality of the escape route models. The first was the completeness of the data provided according to the requirements identified in the building code. The escape routes require geometric representations and alphanumeric properties. The course of each escape route should be specified clearly, and the route length should be assigned to individual courses. Moreover, as the dimensional requirements of the building code depend on the number of people, the cumulative number of people from different escape routes using a particular escape route section (segment) should be provided.

Second, the generated models should satisfy certain data quality requirements. In our concept, this concerns the conformity with the syntax and schema requirements of IFC. This criterion is essential to ensure that the data can be interpreted using software and that permanent and secure storage is provided. We validated the data quality using the buildingSMART Validation Service (version 0.6.2) (buildingSMART International, 2024c) for IFC files. It can evaluate the conformity of an IFC file with the syntax of the STEP Physical File, referenced IFC schema,

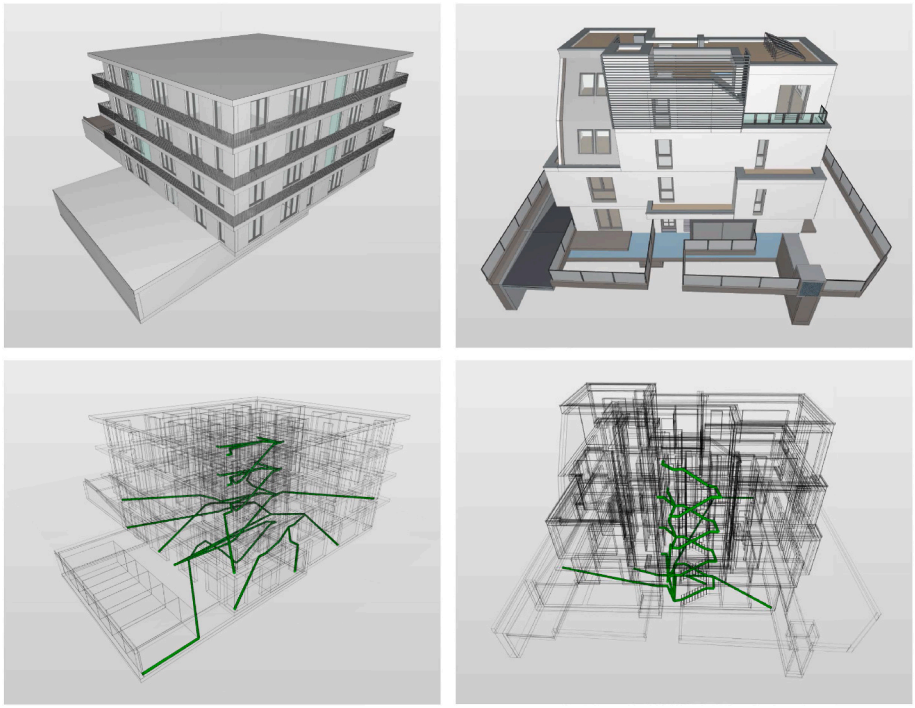

Fig. 10. Generated escape route models for the custom test model (left) and real-world test model (right). The first row shows a 3D view of the original models. The second row shows a combination of the generated escape route models and a transparent 3D view of the original models.

other normative rules of the IFC specification (implementer agreements and informal propositions that are not part of the schema), industry practices, and buildingSMART Data Dictionary (bSDD) content used. Industry practices and bSDD content were not applicable to the escape route models. The criteria data completeness and data quality were already theoretically discussed in the description of the concepts and are to be validated by the implementation.

Finally, the use of the generated models in commercial software was examined. The aim was to assess whether the information was processed correctly and displayed comprehensively. Therefore, we used three software applications (Solibri Office (Solibri Inc., 2024), usBIM.viewer (ACCA software, 2024), and Trimble Connect (Trimble Inc., 2021)) to display the models and assess if all the information can be obtained.

4. Results

The results of this research are escape route models in IFC format that geometrically and alphanumerically represent the escape route information calculated by Fischer et al. (2023). Fig. 10 shows the escape routes generated for the two tested models.

Initially, all the models were generated in the IFC4 format. However, because the spatial relationship for IfcGroup objects has been available only since IFC4.3, we created an IFC4.3 model version for the variant *Routes group Segments–Group* to examine the concept with the new standard. All generated escape route domain models for the custom test model (Haselberger et al., 2024) and the real-world test model (Schnabel et al., 2024) are publicly available.

The five implemented prototypes created five escape route models. Although the models appear highly similar, these differ in terms of the entities used, division in individual elements, spatial relationships, and property assignments. The following detailed description of the generated models is based on the input of the custom test model from BRISE-Vienna (Fischer et al., 2024a): Data on all the escape routes from flats, office rooms, and underground parking garages were used to generate new domain models. Table 2 lists the number of objects generated per variant. In the model of the variant *Routes as Objects*, there are 18 escape routes as IfcAnnotation. Each escape route has three properties in the form of IfcPropertySingleValue. Besides the core property *EscapeRouteLength*, we added the properties *StartLocation* and *PeopleOfRoute* to indicate the room or door where the route starts and the number of people at the start position. This results in an overall number of 54 properties for the variant. *Segments as Objects* and *Routes group Segments–Property* contain no objects for the escape routes. However, these have 154 segments each as IfcSpatialZone. The *Segments as Objects* variant carries a property for the cumulative people in each segment. The number of properties becomes two-fold for the variant *Routes group Segments–Property* because the segments additionally carry a property for the related routes. The names of related routes are stored in IfcPropertyListValues. *Routes group Segments–Part* and *Routes group Segments–Group* combine the routes and segments of the variants before and are identical in terms of the number of objects. The *Routes group Segments–Part* consists of 18 IfcBuildingElementsProxy objects and 154 IfcBuildingElementsParts, whereas the *Routes group Segments–Group* uses 18 IfcGroups and 154 IfcSpatialZones. Three properties per route and one per segment results in 208 properties for both the variants. Furthermore, for each variant containing segments, the geometric length of a segment from its start point to its end point is stored in an IfcElement-Quantity (quantity set).

Table 2
Number of generated objects and properties per variant.

Variant	Escape routes	Segments	Properties
Routes as Objects	18	–	54
Segments as Objects	–	154	154
Routes group Segments–Property	–	154	308
Routes group Segments–Part	18	154	208
Routes group Segments–Group	18	154	208

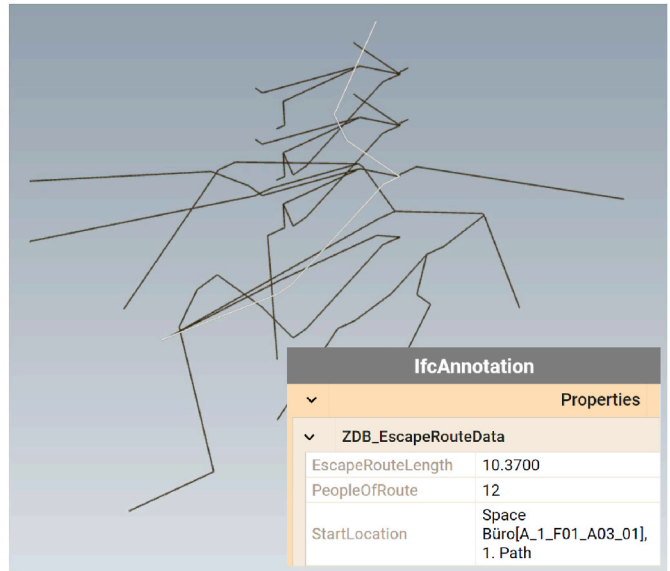

Fig. 11. Geometry and alphanumeric information of the generated escape route model *Routes as Objects*.

Fig. 11 shows the escape route network and the properties of a particular escape route for the *Routes as Objects* variant. Each route is represented by an IfcAnnotation that can be selected to display its properties. For this model, we used the option to represent the geometry of an element simply by its axis. Thus, this matches the visualisation of escape routes in Fischer et al. (2023). However, the geometrical representation of an escape route by its axis adversely affects the model's usage in commercial software applications. When overlaying the original model with escape routes, these can be concealed behind other elements. This is so particularly because the escape route analysis defines the routes to be located on the floor level. Moreover, certain commercial BIM software applications appear to have difficulty displaying objects without volume. This problem occurs in Solibri Office. Other BIM viewers can process these normally, such as usBIM.viewer and Trimble Connect. To solve this problem, IfcAnnotations can maintain a volumetric body representation in the same way as other variants.

Fig. 12 compares the characteristics of the *Segments as Objects* and *Routes group Segments–Property* variants. These appear visually identical and use the same entity as the escape route segments: IfcSpatialZone. The geometry of the segments is defined by cylinders, which increases visibility compared with the first variant. The use of 3D objects as segments results in floor collisions. However, this does not result in any disadvantage for BIM viewers. Individual segments can be selected to retrieve the cumulative number of people in a specific position. Because no objects for escape routes are contained, the *Routes group Segments–Property* variant stores the assignment of segments to routes in a property of the segments. Although entire escape routes cannot be selected directly, these can be filtered as a workaround. The option to filter by property is implemented differently in BIM software applications or may not be available. From the tested software applications, Solibri Office and usBIM.viewer provide suitable filtering options to display the individual escape routes, whereas Trimble Connect does not. Regardless, a major shortcoming is the absence of escape route lengths. However, the available information can be used to display the general flow of people along the escape routes. Using a variable radius that depends on the number of people yields conveniently comprehensible results (see Fig. 13).

The last two variants solve the shortcomings with regard to the completeness of the data: *Routes group Segments–Part* and *Routes group*

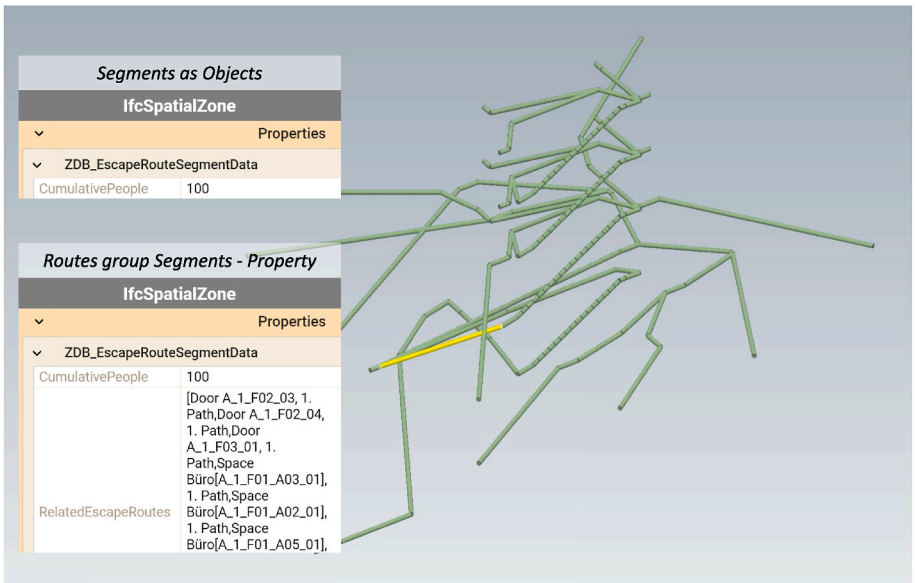

Fig. 12. Geometry and alphanumeric information of the generated escape route models *Segments as Objects* (top) and *Routes group Segments–Property* (bottom).

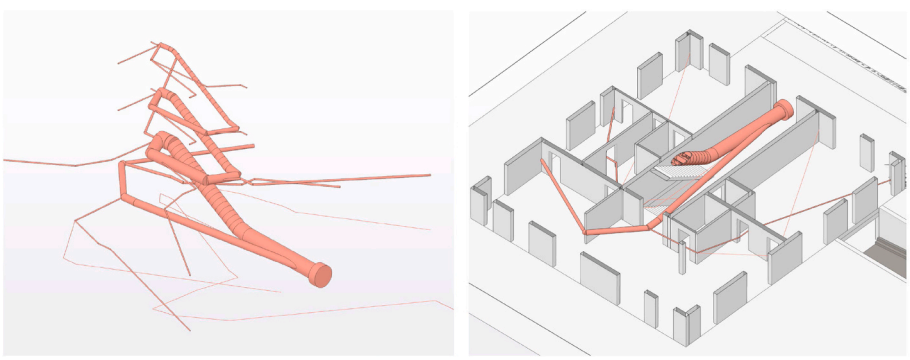

Fig. 13. Escape route network with a variable radius depending on the number of people along the routes.

Segments–Group. Both contain individual elements for segments and routes (see Fig. 14). Thus, routes and segments can be selected, and their properties can be retrieved. The only difference is in the entities used for individual objects and spatial relationships.

Routes group Segments–Part uses IfcBuildingElementPart for segments and IfcBuildingElementProxy for routes. Both the element types are related to the spatial structure of the model by the IfcRelContainedInSpatialStructure relationship. *Routes group Segments–Group* combines segments as IfcSpatialZone in routes as IfcGroup. The segments can be related to the spatial structure of the model using the IfcRelContainedInSpatialStructure as well, whereas IfcGroup requires the use of IfcRelReferencedInSpatialStrucutre. However, the relationships between the individual groups and the building hierarchy could not be processed by the software applications tested, regardless of the IFC format used (IFC4 or IFC4.3). These groups are separate from the building hierarchy. This can be attributed to the deficiency of IFC4.3-certified software applications.

In general, the usage and reliability of building models are closely linked to data quality. For our escape route models, this concerns the conformity with the syntax of the STEP Physical File and the specified IFC schema. The results of the data quality validation using the buildingSMART Validation Service for the models generated with IfcOpenshell (Python) are shown in Fig. 15.

The first three simple variants generated technically defectless files. All the rules of the IFC schema were adhered to. However, the more complex variants exhibited the theoretical issues that had been discussed. In our implementation of *Routes group Segments–Part*, we omitted the exclusivity of the IfcRelAggregates relationship when assigning IfcBuildingElementPart (segments) to IfcBuildingElementProxy (routes). Rather, we assigned segments to several routes to prevent replicating elements. The use of a single part in several assemblies was not intended conceptually. Therefore, if a segment occurs in several escape routes, exclusivity is violated (see Fig. 16 top panel). The error message indicates that by using IfcRelAggregates (a subtype of IfcRelDecomposes), an element can be related to only one element. For example, IfcBuildingElementPart #2636 is related to two elements and induces an error. In addition, a normative rule error occurred (see Fig. 16 bottom panel). The particular normative rule specifies that elements that are aggregated in another IfcElement must not be assigned their own containment via IfcRelContainedInSpatialStructure. This issue pertaining to the variant *Routes group Segments–Part* was not identified during the theoretical analysis, as the normative rules address specifications beyond the scope of the standardised IFC schema. However, the errors did not cause any problem in the tested software applications.

The schema error in the *Routes group Segments–Group* variant is owing to the use of IfcRelReferencedInSpatialStructure for the spatial relationship of IfcGroup (see Fig. 17). The error message indicates that this relationship requires subtypes of IfcProduct for the related elements. This does not apply to IfcGroup. However, IFC4.3 introduces the use of IfcRelReferencedInSpatialStructure for IfcGroup in addition to IfcProduct. Therefore, the *Routes group Segments—Group* variant does not have a schema error in the IFC4.3 format (see the last line of Fig. 15).

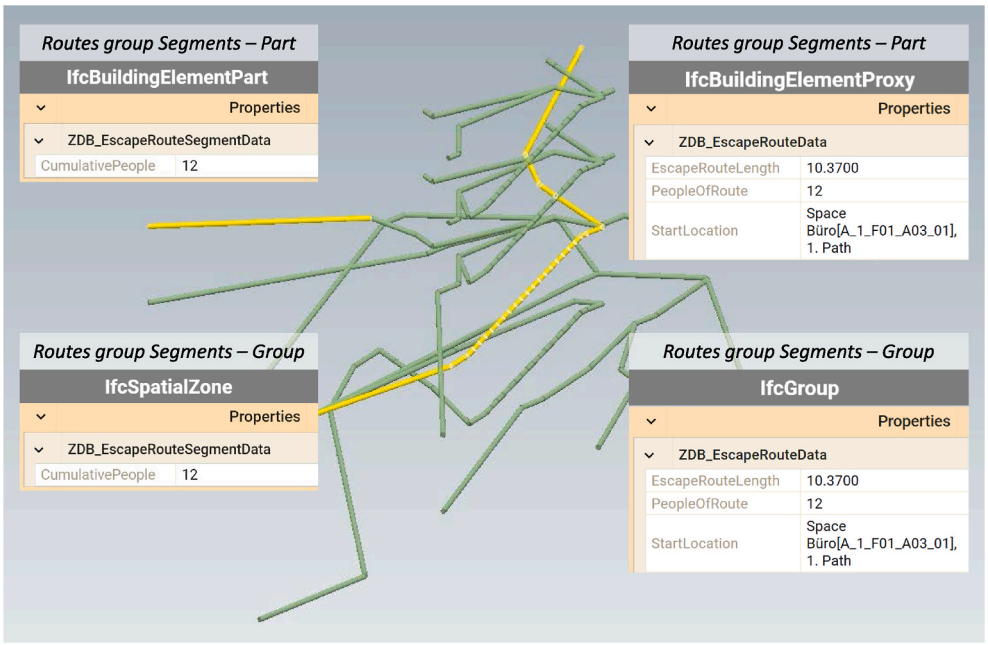

Fig. 14. Geometry and alphanumeric information of segments (left) and routes (right) of the generated escape route models *Routes group Segments–Part* (top) and *Routes group Segments–Group* (bottom).

File Name	STEP Syntax	IFC Schema	Normative IFC Rules
Py-CustomEscapeRouteModel-RoutesAsObjects.ifc	✓	✓	✓
Py-CustomEscapeRouteModel-SegmentsAsObjects.ifc	✓	✓	✓
Py-CustomEscapeRouteModel-RoutesGroupSegments-Property.ifc	✓	✓	✓
Py-CustomEscapeRouteModel-RoutesGroupSegments-Part.ifc	✓	!	!
Py-CustomEscapeRouteModel-RoutesGroupSegments-Group.ifc	✓	!	✓
Py-CustomEscapeRouteModel-RoutesGroupSegments-Group-IFC4X3.ifc	✓	✓	✓

Fig. 15. Overview of the data quality assessment of the implemented model variants with the buildingSMART Validation Service 0.6.2.

The web-ifc library for JavaScript causes some additional errors. Some values that require an integer are stored as a decimal and vice versa (put a dot at the end of an integer or not). However, these errors are not related to the concepts developed and did not affect the use in commercial software. They are small software bugs and should be fixed by the web-ifc developers.

Table 3 summarises the performances of the variants in terms of completeness, data quality, and usage. The implementation confirmed the theoretically discussed strengths and weaknesses of the variants.

Routes group Segments–Part and *Routes group Segments–Group* are the only variants that fulfil the data completeness requirements regarding an escape route network. However, both the variants have schema errors. This is not permissible from the perspective of data archiving. Compliance with the standardised IFC schema is fundamental to the permanent availability and interpretability of data. The schema error of the *Routes group Segments–Group* variant is solved using IFC4.3. From IFC version 4.3.x onwards, the entity IfcGroup can be related to the spatial structure. The use of this model remains constrained because commercial software applications remain incapable of processing IFC4.3. However, considering the progressive implementation and application of IFC4.3 schema, the variant *Routes group Segments–Group* is the most potential method to represent an escape route network in an IFC model. The concept covers three fundamental requirements derived from the Austrian Building Code and found in fire safety regulations around the world. The geometric approaches presented offer the possibility to define an escape route either by its axis as a polyline or by a 3D shape. The 3D shape representation allows the actual width and height of the escape route to be reconstructed, if these are available from the preliminary calculation. We have also demonstrated the ability to store properties for both escape routes and their segments. In the same way, regionally required characteristics or other calculation results can be stored. This ensures adaptability to regional specifications and demonstrates the general potential of our approach.

Schema - IfcBuildingElementPart.Decomposes

Id	Entity	Severity	Message
#2636	IfcBuildingElementPart	Error	With inverse: Decomposes : SET [0:1] OF IfcRelAggregates FOR RelatedObjects Value: [1. #4131=IfcRelAggregates('0b2OHFZAr5OQDrMU4XVajA',#5,'Proxy Aggregate',$,#4130,(#2506,#2532,#2558,#2584,#2610,#2636,#2662,#2688,#2376,#2402,#1882)) 2. #4171=IfcRelAggregates('0JbKZc6nPAzOOtivDE0JnJ',#5,'Proxy Aggregate',$,#4170,(#2948,#2974,#3000,#2610,#2636,#2662,#2688,#2376,#2402,#1882))] Not valid

SPS003 - Correct Containment Of Assemblies

ⓘ The rule verifies that IfcElement that are aggregated in another IfcElement must not be contained using IfcRelContainedInSpatialStructure.

https://github.com/buildingSMART/ifc-gherkin-rules/tree/178cb9bc177ecd8f178584384845c8895ef1c808/features/SPS003_Correct-containment-of-assemblies.feature

Severity	Id	Entity	Expected	Observed
Error	#998	IfcBuildingElementPart	*The value of attribute ContainedInStructure must be empty*	IfcRelContainedInSpatialStructure(#1005)

Fig. 16. IFC schema error (top) and normative rule error (bottom) from the buildingSMART Validation Service for the variant *Routes group Segments–Part*.

Schema - RelatedElements

Id	Entity	Severity	Message
#4161	IfcRelReferencedInSpatialStructure	Error	With attribute: <attribute RelatedElements: <set [1:?] of <entity IfcProduct>>> Value: #4154=IfcGroup('3S$zrX_b5D5PtX5UkHOHII',#5,'EscapeRoute Space Büro[A_1_F00_A02_04], 1. Path',$,$) Not valid

Fig. 17. IFC schema error from the buildingSMART Validation Service for the variant *Routes group Segments–Group*.

Table 3
Assessment of the implemented variants with regard to the validation criteria (+passed; ∼ restricted; – failed).

Variant	Completeness	Schema	Usage
Routes as Objects	–	+	∼
Segments as Objects	–	+	+
Routes group Segments–Property	∼	+	∼
Routes group Segments–Part	+	–	+
Routes group Segments–Group	+	+[a]	∼

[a] From IFC4.3.x onward.

5. Conclusions

This study focused on generating a new domain model for escape route data generated during code compliance checking. The purpose of this new domain model saved in the IFC format is to provide permanently available data archiving. Open formats are commonly used for digital building models. However, the additional storage of newly created data from automated calculations or analyses of these models has not been addressed sufficiently.

The new domain model is intended to be part of the escape route permit process of the Vienna Building Authority for the independent and permanent archiving of escape route analysis results. This study demonstrated the approach's feasibility by identifying the relevant data and requirements for escape routes, defining general concepts for representation, mapping these with the IFC standard, and implementing algorithms to automatically create escape route models. From variants with different complexities, we identified a suitable solution for integration into the escape route permitting process: *Routes group Segments–Group*. The variant *Routes group Segments–Group* complies with the IFC standard (from IFC4.3 onwards) and contains all the necessary information. It combines abstract objects for escape routes (IfcGroup) with geometric objects for escape route segments (IfcSpatialZone). The use of this variant in commercial software applications is currently marginally restricted because of the incomplete IFC4.3 implementation in the software. With IFC4.3 being ISO-standardised since January 2024, commercial software applications would adapt to IFC4.3. This should address this shortcoming.

Our implementation revealed that the results of code compliance checking can be stored in IFC. In addition to permanent and safe data storage, this enables integration with other types of building data. It allows for the convenient sharing and exchange of information between different software applications, building authority departments, and project stakeholders. Using it as a digital escape route plan can facilitate maintenance and management activities such as conducting fire drills, developing emergency plans, and defining the location of exit signs. Combined with augmented reality, it can also support fire brigade operations in the event of emergencies (Gerger et al., 2023). With regard to the escape route permit process, the IFC model can function as the data transfer between escape route generation and escape route validation of (Fischer et al., 2023). This would reduce the manual effort required by planners.

However, in the current implementation, the escape routes are simplified to cylinders. A noteworthy extension is the geometric representation of the required space. The required space is a rectangular

profile with variable width extruded along the course of the escape route. However, for this extension, the output of the previous escape route analysis should be extended. It is important to note that this study focuses on the requirements identified by the Austrian building regulations and Vienna Building Authority. Even though other building codes also contain these general requirements, other cities and countries may require additional regulations and data. In general, our results promote the development of additional domain models for newly created data during code compliance checking, other calculations, or simulations. Therefore, new cases should be identified for this concept.

CRediT authorship contribution statement

Simon Fischer: Writing – original draft, Visualization, Validation, Methodology, Investigation, Data curation, Conceptualization. **Harald Urban:** Writing – review & editing, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization. **Christian Schranz:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Matthias Haselberger:** Visualization, Validation, Software, Investigation. **Florian Schnabel:** Visualization, Software, Investigation.

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

We have referenced the used data in the article.

Acknowledgements

Funding: This work was supported by Urban Innovative Actions [grant number UIA04-081], an initiative of the European Union. The authors acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme.

References

ACCA software, 2024. IFC-Viewer – usBIM. https://www.accasoftware.com/en/ifc-viewer. (Accessed 18 March 2024).
Amor, R., Dimyadi, J., 2021. The promise of automated compliance checking. Dev. Built Environ. 5, 100039 https://doi.org/10.1016/j.dibe.2020.100039.
Austrian Institute of Construction Engineering, 2023a. OIB guidelines. https://www.oib.or.at/en/oib-guidelines. (Accessed 7 March 2024).
Austrian Institute of Construction Engineering, 2023b. OIB Guideline 2: safety in case of fire. https://www.oib.or.at/en/node/9323711. (Accessed 7 March 2024).
Austrian Institute of Construction Engineering, 2023c. OIB Guideline 4: safety in use and accessibility. https://www.oib.or.at/en/node/9323717. (Accessed 7 March 2024).
Battisti, K., Dörn, M., Eggeling, E., Eichler, C., Loës, J.M., Scherret, J., Tsoggerel, Z., Ullrich, T., 2022. An automatic process for the application of building permits. Buildings 13, 78. https://doi.org/10.3390/buildings13010078.
Beach, T.H., Hippolyte, J.-L., Rezgui, Y., 2020. Towards the adoption of automated regulatory compliance checking in the built environment. Autom. ConStruct. 118, 103285 https://doi.org/10.1016/j.autcon.2020.103285.
Borrmann, A., Kneidl, A., Köster, G., Ruzika, S., Thiemann, M., 2012. Bidirectional coupling of macroscopic and microscopic pedestrian evacuation models. Saf. Sci. 50, 1695–1703. https://doi.org/10.1016/j.ssci.2011.12.021.
Buildings Department, 2023. Code of practice for fire safety in buildings 2011 (June 2023 Edition). https://www.bd.gov.hk/en/resources/codes-and-references/codes-and-design-manuals/index.html. (Accessed 8 May 2024).
buildingSMART International, 2024a. 5.4.2.23 IfcSpatialZoneTypeEnum - IFC 4.3.2 documentation. https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcSpatialZoneTypeEnum.htm. (Accessed 21 March 2024).
buildingSMART International, 2024b. 5.4.3.70 IfcSpatialZone - IFC 4.3.2 documentation. https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcSpatialZone.htm. (Accessed 21 March 2024).
buildingSMART International, 2024c. Validation Service. https://technical.buildingsmart.org/services/validation-service/. (Accessed 21 March 2024).
Ciccone, A., Ciotta, V., Asprone, D., 2023. Integration of structural information within a BIM-based environment for seismic structural e-permits. J. Civ. Eng. Manag. 29, 171–193. https://doi.org/10.3846/jcem.2023.18460.
City of New York, 2022. Building code – chapter 10 – means of egress. https://www.nyc.gov/site/buildings/codes/2022-construction-codes.page. (Accessed 8 May 2024).
Eastman, C., Lee, J.-M., Jeong, Y.-S., Lee, J.-K., 2009. Automatic rule-based checking of building designs. Autom. ConStruct. 18, 1011–1033. https://doi.org/10.1016/j.autcon.2009.07.002.
El Jazzar, M., Schranz, C., Urban, H., Nassereddine, H., 2021. Integrating construction 4.0 technologies: a four-layer implementation plan. Front. Built Environ. 7, 671408 https://doi.org/10.3389/fbuil.2021.671408.
EUnet4DBP, 2024. European network for digital building permits. https://eu4dbp.net/. (Accessed 7 March 2024).
Fachkommission Bauaufsicht zur Musterbauordnung und Muster-Beherbergungsstättenverordnung. https://www.bauministerkonferenz.de/suchen.aspx?id=762&o=759O762&s=musterbauordnung, 2022–. (Accessed 8 May 2024).
Fischer, S., Schranz, C., Urban, H., Pfeiffer, D., 2023. Automation of escape route analysis for BIM-based building code checking. Autom. ConStruct. 156, 105092 https://doi.org/10.1016/j.autcon.2023.105092.
Fischer, S., Schranz, C., Urban, H., Pfeiffer, D., 2024a. Custom test model for escape route analysis in IFC format. TU Wien Res. Data v2. https://doi.org/10.48436/hx8gz-zw339.
Fischer, S., Schranz, C., Urban, H., Pfeiffer, D., 2024b. Real-world test model for escape route analysis in IFC format. TU Wien Res. Data v2. https://doi.org/10.48436/fnmrh-crh59.
Fu, M., Liu, R., Qi, B., Issa, R.R., 2020. Generating straight skeleton-based navigation networks with Industry Foundation Classes for indoor way-finding. Autom. ConStruct. 112, 103057 https://doi.org/10.1016/j.autcon.2019.103057.
Gerger, A., Urban, H., Schranz, C., 2023. Augmented reality for building authorities: a use case study in Austria. Buildings 13, 1462. https://doi.org/10.3390/buildings13061462.
Han, C.S., Kunz, J.C., Law, K.H., 1998. Client/server framework for on-line building code checking. J. Comput. Civ. Eng. 12, 181–194. https://doi.org/10.1061/(ASCE)0887-3801(1998)12:4(181).
Haselberger, M., 2023. Realisierung von Fluchtwegsdaten im IFC-Format mithilfe von IfcOpenShell (Realization of escape route data in the IFC format using IfcOpenShell). MSc thesis. Faculty of Civil and Environmental Engineering, TU Wien, Vienna, Austria. https://doi.org/10.34726/hss.2023.117407.
Haselberger, M., Schnabel, F., Fischer, S., Urban, H., Schranz, C., 2024. Custom escape route models in IFC format. TU Wien Res. Data v1. https://doi.org/10.48436/dpwd5-33k50.
Hjelseth, E., 2015. Public BIM-based model checking solutions: lessons learned from Singapore and Norway, in: building information modelling (BIM) in design. Construction and Operations. WIT Press, Southampton. https://www.witpress.com/books/978-1-84564-914-2.
HM Government, 2022. Approved Document B (fire safety) volume 2: buildings other than dwellings. In: 2019 Edition Incorporating 2020 and 2022 Amendments. https://www.planningportal.co.uk/applications/building-control-applications/building-control/approved-documents/part-b-fire-safety/approved-document-b-volume-2-buildings-other-than-dwellings. (Accessed 8 May 2024).
Höcker, M., Berkhahn, V., Kneidl, A., Borrmann, A., Klein, W., 2010. Graph-based approaches for simulating pedestrian dynamics in building models. In: eWork and eBusiness in Architecture, Engineering and Construction – Proceedings of the European Conference on Product and Process Modelling 2010, pp. 389–394. https://doi.org/10.1201/b10527.
IfcOpenShell, 2024. GitHub – IfcOpenShell. https://github.com/IfcOpenShell/IfcOpenShell. (Accessed 12 March 2024).
International Code Council, 2021. International building code 2018 – chapter 10 – means of egress. https://codes.iccsafe.org/content/IBC2018P6/chapter-10-means-of-egress. Accessed 8 May 2024.
International Organization for Standardization, 2024. ISO 16739-1:2024 Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries — Part 1: Data Schema. Genf, Switzerland. https://www.iso.org/standard/70303.html. (Accessed 16 June 2024).
Kaneta, T., Furusaka, S., Tamura, A., Deng, N., 2016. Overview of BIM implementation in Singapore and Japan. J. Civ. Eng. Architect 10. https://doi.org/10.17265/1934-7359/2016.12.001.
Kannala, M., 2005. Escape Route Analysis Based on Building Information Models: Design and Implementation, MSc Thesis, Department of Computer Science and Engineering. Helsinki University of Technology, Helsinki, Finland. http://mattikannala.fi/projects/masters-thesis/Escape-Route-Analysis-Based-on-Building-Information-Models-Design-and-Implementation.pdf. (Accessed 31 March 2023).
Kim, J., Lee, J.-K., Shin, J., Kim, H., 2017. Classification of objects and their properties defined in Korea building act to translate into KBimCode and their application. In: ISARC Proceedings, IAARC, pp. 148–155. ISBN (9781510844735). https://www.proceedings.com/35372.html.
Kim, H., Lee, J.-K., Shin, J., Choi, J., 2019. Visual language approach to representing KBimCode-based Korea building code sentences for automated rule checking. J. Comput. Des. Eng. 6, 143–148. https://doi.org/10.1016/j.jcde.2018.08.002.
Kim, I., Choi, J., Teo, E.A.L., Sun, H., 2020. Development of KBim E-submission prototypical system for the openBIM-based building permit framework. J. Civ. Eng. Manag. 26, 744–756. https://doi.org/10.3846/jcem.2020.13756.
Klinc, R., Turk, Ž., 2019. Construction 4.0 – digital transformation of one of the oldest industries. Econ. Bus. Rev. 21 (3), 393–410. https://doi.org/10.15458/ebr.92.

Kneidl, A., Borrmann, A., Hartmann, D., 2012. Generation and use of sparse navigation graphs for microscopic pedestrian simulation models. Adv. Eng. Inf. 26, 669–680. https://doi.org/10.1016/j.aei.2012.03.006.

Kneidl, A., Hartmann, D., Borrmann, A., 2013. A hybrid multi-scale approach for simulation of pedestrian dynamics. Transport. Res. C Emerg. Technol. 37, 223–237. https://doi.org/10.1016/j.trc.2013.03.005.

Krischmann, T., Urban, H., Schranz, C., 2020. Entwicklung eines openBIM-Bewilligungsverfahrens (Development of an openBIM submission process). Bauingenieur 95 (9), 335–344. https://doi.org/10.37544/0005-6650-2020-09-61.

Lee, J.-K., 2011. Building Environment Rule and Analysis (BERA) Language and its Application for Evaluating Building Circulation and Spatial Program. Ph.D. thesis. Georgia Institute of Technology, Atlanta, USA. https://smartech.gatech.edu/handle/1853/39482. (Accessed 31 March 2023).

Lee, J.-K., Eastman, C.M., Lee, J., Kannala, M., Jeong, Y.-S., 2010. Computing walking distances within buildings using the universal circulation network. Environ. Plann. Plann. Des. 37, 628–645. https://doi.org/10.1068/b35124.

Lee, H., Lee, J.-K., Park, S., Kim, I., 2016. Translating building legislation into a computer-executable format for evaluating building permit requirements. Autom. ConStruct. 71, 49–61. https://doi.org/10.1016/j.autcon.2016.04.008.

Lin, W.Y., Lin, P.H., 2018. Intelligent generation of indoor topology (i-GIT) for human indoor pathfinding based on IFC models and 3D GIS technology. Autom. ConStruct. 94, 340–359. https://doi.org/10.1016/j.autcon.2018.07.016.

Lin, Y.-H., Liu, Y.-S., Gao, G., Han, X.-G., Lai, C.-Y., Gu, M., 2013. The IFC-based path planning for 3D indoor spaces. Adv. Eng. Inf. 27, 189–205. https://doi.org/10.1016/j.aei.2012.10.001.

Liu, L., Li, B., Zlatanova, S., van Oosterom, P., 2021. Indoor navigation supported by the industry foundation Classes (IFC): a survey. Autom. ConStruct. 121, 103436 https://doi.org/10.1016/j.autcon.2020.103436.

Macit İlal, S., Günaydın, H.M., 2017. Computer representation of building codes for automated compliance checking. Autom. ConStruct. 82, 43–58. https://doi.org/10.1016/j.autcon.2017.06.018.

Malsane, S., Matthews, J., Lockley, S., Love, P.E., Greenwood, D., 2015. Development of an object model for automated compliance checking. Autom. ConStruct. 49, 51–58. https://doi.org/10.1016/j.autcon.2014.10.004.

Nisbet, N., Wix, J., Conover, D., 2008. The future of virtual construction and regulation checking. In: Brandon, P., Kocatürk, T. (Eds.), Virtual Futures for Design, Construction & Procurement. Blackwell Publishing Ltd, pp. 241–250. https://doi.org/10.1002/9781444302349.ch17.

Noardo, F., Guler, D., Fauth, J., Malacarne, G., Mastrolembo Ventura, S., Azenha, M., Olsson, P.-O., Senger, L., 2022a. Unveiling the actual progress of Digital Building Permit: getting awareness through a critical state of the art review. Build. Environ. 213, 108854 https://doi.org/10.1016/j.buildenv.2022.108854.

Noardo, F., Wu, T., Arroyo Ohori, K., Krijnen, T., Stoter, J., 2022b. IFC models for semi-automating common planning checks for building permits. Autom. ConStruct. 134, 104097 https://doi.org/10.1016/j.autcon.2021.104097.

Park, S., Lee, Y.C., Lee, J.K., 2016. Definition of a domain-specific language for Korean building act sentences as an explicit computable form. J. Inf. Technol. Construct. 21, 422–433. https://www.itcon.org/2016/26.

Schnabel, F., Haselberger, M., Fischer, S., Urban, H., Schranz, C., 2024. Real-World Escape Route Models in IFC format. TU Wien Research Data, v1. https://doi.org/10.48436/rrd14-t1108 [Data set].

Schranz, C., Urban, H., Gerger, A., 2021. Potentials of Augmented Reality in a BIM based building submission process. J. Inf. Technol. Construct. 26, 441–457. https://doi.org/10.36680/j.itcon.2021.024.

Shahi, K., McCabe, B.Y., Shahi, A., 2019. Framework for automated model-based e-permitting system for municipal jurisdictions. J. Manag. Eng. 35, 04019025 https://doi.org/10.1061/(ASCE)ME.1943-5479.0000712.

Solibri Inc., 2024. Solibri office. https://www.solibri.com/solibri-office. (Accessed 18 March 2024).

Song, J., Kim, J., Lee, J.K., 2019. Converting KBimCode into an executable code for the automated design rule checking system. In: Haeusler, M.H., Schnabel, M.A., Fukuda, T. (Eds.), Intelligent and Informed – Proceedings of the 24th International Conference on Computer-Aided Architectural Design Research in Asia, CAADRIA 2019, the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), ISBN 9789887891710, pp. 795–804.

Taneja, S., Akinci, B., Garrett, J.H., Soibelman, L., 2016. Algorithms for automated generation of navigation models from building information models to support indoor map-matching. Autom. ConStruct. 61, 24–41. https://doi.org/10.1016/j.autcon.2015.09.010.

That Open Company, 2024. GitHub – web IFC. https://github.com/ThatOpen/engine_web-ifc. (Accessed 12 March 2024).

Trimble Inc., 2021. Trimble Connect. https://connect.trimble.com/. (Accessed 18 March 2024).

Uhm, M., Lee, G., Park, Y., Kim, S., Jung, J., Lee, J.-K., 2015. Requirements for computational rule checking of requests for proposals (RFPs) for building designs in South Korea. Adv. Eng. Inf. 29, 602–615. https://doi.org/10.1016/j.aei.2015.05.006.

UIA–Urban Innovative Actions, 2019. BRISE-Vienna – building regulations information for submission envolvement. https://www.uia-initiative.eu/en/uia-cities/vienna-call4. (Accessed 18 March 2024).

Ullah, K., Raitviir, C., Lill, I., Witt, E., 2020. BIM adoption in the AEC/FM industry – the case for issuing building permits. Int. J. Strat. Property Manag. 24, 400–413. https://doi.org/10.3846/ijspm.2020.13676.

Ullah, K., Witt, E., Lill, I., 2022. The BIM-based building permit process: factors affecting adoption. Buildings 12, 45. https://doi.org/10.3390/buildings12010045.

Urban, H., Fischer, S., Schranz, C., 2024. Adapting to an OpenBIM building permit process: a case study using the example of the city of Vienna. Buildings 14, 1135. https://doi.org/10.3390/buildings14041135.

Yang, L., Worboys, M., 2015. Generation of navigation graphs for indoor space. Int. J. Geogr. Inf. Sci. 29, 1737–1756. https://doi.org/10.1080/13658816.2015.1041141.

Yuan, W., Schneider, M., 2010. Supporting 3D route planning in indoor space based on the LEGO representation. In: ISA '10: Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, ISA '10. Association for Computing Machinery, New York, USA, pp. 16–23. https://doi.org/10.1145/1865885.1865890.