

ITEA4 22003 FireBIM

Deliverable 2.2

FireBIM Ontology Stack

Deliverable type:	Other
Deliverable reference number	ITEA 2203 D2.2
Related work package	WP2
Submission date	
Dissemination level	Public
Contributors	Alex Donkers, Ekaterina Petrova, Cedric Driesen
Abstract	
Keywords	

Change log

Description	Author	Date
V0.1	Alex Donkers	01-09-2026
V1.0	Alex Donkers	23-09-2026

1. Abbreviations

ACC	Automated Compliance Checking
AEC	Architecture, Engineering, and Construction
BIM	Building Information Model/Modeling
IFC	Industry Foundation Classes
LBD	Linked Building Data
SWT	Semantic Web Technology

2. Introduction

Automated Compliance Checking (ACC) in the Architecture, Engineering, and Construction (AEC) sector has long been pursued to reduce human error, minimize project delays, and ensure rigorous adherence to safety standards. Historically, interpreting building regulations and evaluating Building Information Models (BIM) heavily relied on manual processes. This manual approach is inherently prone to subjective interpretation, is time-consuming, and difficult to scale, particularly as modern buildings become more complex. Designers therefore often rely on the knowledge of domain specialists. In the case of fire safety regulatory compliance checking, fire safety experts are often hired to assess designs. Fire safety expertise is often only included relatively late in the design process, which could lead to costly redesigns. Additionally, fire safety regulations, as well as fire safety terminology, changes when the expert crosses the border to another (European) country. While much knowledge in the fire safety domain stems from the domain of (building) physics, local differences between the fire safety regulations exist. It is therefore difficult for a fire safety expert to understand the terminology and the regulations in other countries, limiting cross-border European collaboration.

Fire safety regulations, however, present an important challenge within ACC in the AEC industry, as failures in fire safety can lead to catastrophic loss of life and property. Regulations are therefore dense and highly prescriptive and often accompanied with a lot of knowledge stored in technical guidelines, local or international norms, and written and unwritten local practices. The information is highly scattered, and finding relationships between the separate pieces of information is a complex task that requires expert knowledge and experience.

The international schema for Building Information Modeling, the Industry Foundation Classes (IFC) schema, contains classes and properties related to buildings. The format is widely applied to enable actors in the AEC industry to digitally model their buildings, and exchange that information with other actors. IFC, however, lacks the explicit, specialized semantics required to model and evaluate fire safety knowledge that reflects the national fire safety codes, such as calculation methods for compartmentation boundaries, specific fire-resistance ratings, or complex travel distance rules.

The FireBIM project addresses these challenges by shifting compliance checking into the domain of Semantic Web Technologies (SWT) and Linked Building Data (LBD). By formalizing regulatory documents and building models into machine-readable RDF (Resource Description Framework) graphs, the project aims to develop a standardized, non-proprietary method to automatically verify whether a designed building complies with fire safety regulations.

2.1. Context and Motivation

The primary driver for this work is the current fragmentation between legal regulatory documents and the digitized building data. When a fire safety engineer reviews a building design, they must mentally translate vague or highly specific legal requirements into (spatial) checks on a floor plan or 3D model. By explicitly modelling fire safety knowledge, and mapping this against the knowledge in the BIM domain, it becomes easier for fire safety experts to assess a building. This would also enable the creation of fire safety-specific modelling guidelines for building information modelers. By explicitly modelling regulatory knowledge, it would be easier for both experts and non-experts to navigate the fire safety codes and the accompanying fire safety knowledge. And by explicitly modelling fire safety terminology across multiple countries, it would be easier for actors in the AEC to participate in cross-border collaborations and develop cross-border business models.

2.2. Objectives of Deliverable 2.2

The primary objective of Deliverable 2.2 is to develop a set of ontologies that contain the necessary sets of terms and definitions to enable fire safety regulatory compliance checking of buildings, especially in their (early) design stage). This results in three ontologies:

- One ontology that semantically describes concepts related to regulatory documents, their parts, their internal and external relationships, and metadata;
- One ontology that semantically describes fire safety terminology, and covers the mapping between that terminology and the domain of BIM;
- One ontology that semantically describes digital regulations and enables human evaluation of such digital regulations.

A secondary objective is to develop a strategy for long-term maintenance and extendibility. The FireBIM project has a limited set of partners from 5 European countries. The project wishes to enable future extensions of the ontologies in two directions:

- Adding new countries;
- Adding new (fire safety) knowledge domains.

2.3. Positioning within Work Package 2 (WP2)

Deliverable 2.2 is an enabler for future tasks in this work package (see Figure 1). The work is heavily based on the results of *D2.1 Fire Code Matrix*, which contains a detailed collection of fire safety terminology from the five participating countries in this project. This deliverable aims to convert that knowledge into an ontology that is both machine- and human-readable. The deliverable then feeds into *D2.3 Generic framework for digitization of fire safety regulations*, where the ontology stack is directly applied in the conversion of textual to machine-readable building codes. Subsequently, the results of D2.2 are also present in D2.4 (which is the software output of D2.3) and D2.5 (which contains the set of machine-readable building codes).

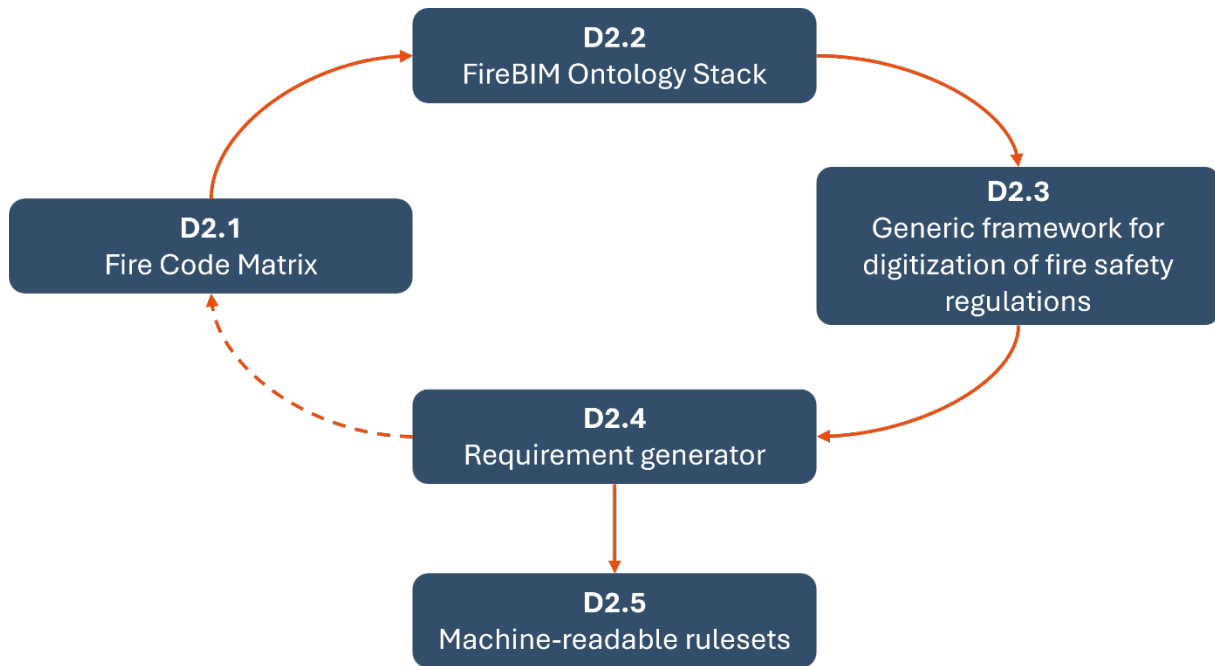


Figure 1 Placement of D2.3 relative to other deliverables in Work Package 2

2.4. Positioning within FireBIM

The FireBIM Ontology Stack is supposed to serve as a central library of terms and definitions in the FireBIM project, and is aimed to be reused in subsequent work packages. WP3's reference architecture will take the ontology stack as input and reuse the ontologies to enable querying and reasoning in the platform. WP4's descriptions of end-user applications will steer the future development of the FireBIM Ontology Stack, and sets of end-user requirements should be transformed to requirements for the FireBIM Ontology Stack (e.g. through the development of competency questions). The development of demonstrators in WP5 is technically connected to the results of D2.2, as either a mapping to or a direct use of the terminology in the FireBIM Ontology Stack is necessary to enable SHACL-based rule checking. Finally, the results of D2.2 are linked to activities in WP6, including T6.2 (Communication and dissemination), T6.3 (Standardisation and certification), and T6.4 (Education and licensing).

2.5. Methodology

The FBO Ontology Stack covers knowledge from multiple countries, and therefore, a rigorous method was needed to capture and formalize this knowledge. The project follows the ontology development method developed by Donkers et al. (2022), consisting of the following 7 steps: 1. Specification, 2. Knowledge acquisition, 3. Requirement specification, 4. Building, 5. Evaluation, 6. Integration, 7. Documentation and publication. The method considers these steps as non-linear, where one would typically iterate multiple times over steps 2- 5.

The FireBIM Ontology Stack contains three separate ontologies, each with a different goal, and therefore a different development process. The specific development process of each ontology is presented in the respective subsections in Section 5, 6, and 7.

3. State of the Art

3.1. Ontologies for Fire Safety

To find the state-of-the-art on ontologies for fire safety, a systematic literature was performed. An advanced Web of Science query is prompted to find existing literature in this domain. The query aims to find papers that are in the domain of fire safety, that use or present ontologies or other semantic web technologies and that are focusing on or applicable to the construction industry. The query in Listing 1 resulted in 121 papers.

```
ALL=(Fire) AND ALL=(Construction OR Building) AND ALL=(Ontology OR Semantic Web OR RDF)
```

Listing 1 Web of Science query

Further refinement of papers is made based on an abstract search in a literature grid. Papers that did not present or apply an ontology or semantic web technologies, do not focus or are not applicable to the construction industry, or do not focus on fire were filtered out. Papers in other languages than English were also filtered out. The refinement resulted in 31 papers (17 in journals, 14 in conference proceedings). Figure 2 shows the distribution of these papers over time. The production of papers has been relatively stable for 15 years, but has been on the rise in the past 5 years (*note that the results only contain the first six months of 2025, before major conferences in this domain*). Further subdivision shows that the number of journal publications leans more towards those recent years, while the work during 2006-2017 consists for 89% of conference publications. This emphasizes recent maturity in the topic. Figure 3 shows the distribution of papers per country and continent (based on the first author affiliation). Notable is the large number of publications from China, and the relatively large number of papers from Taiwan (all from the same department), Germany (all by the same supervisor), and Brazil (all from different research teams). Also notable is that Asia (colored red in Figure 3) and Europe (colored blue in Figure 3) are both responsible for 42% of all papers.

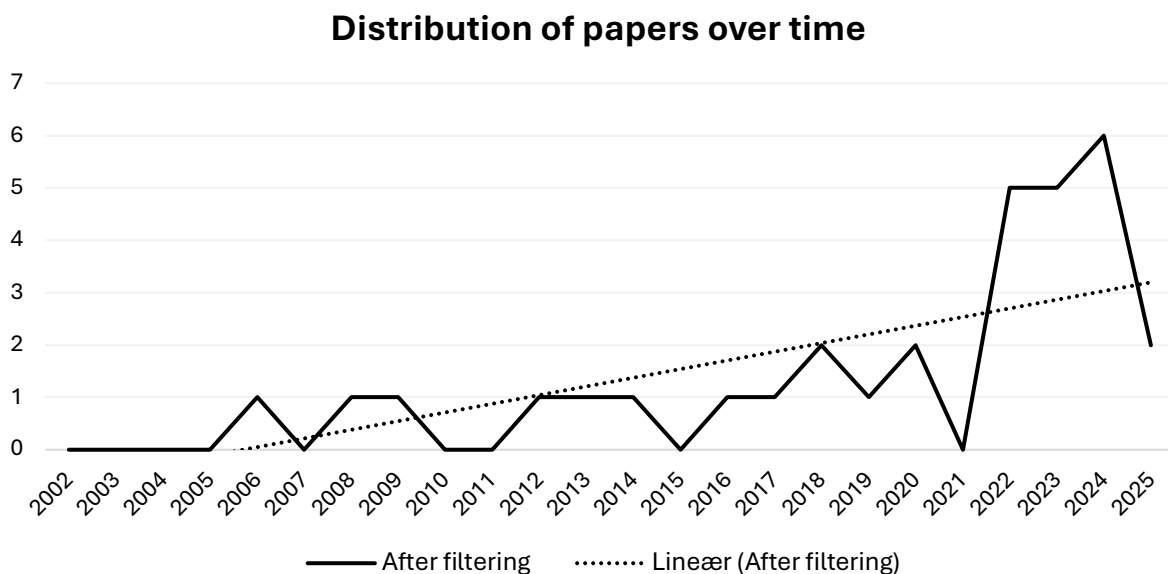


Figure 2 Papers on fire safety ontologies over time

Distribution of papers per country and continent

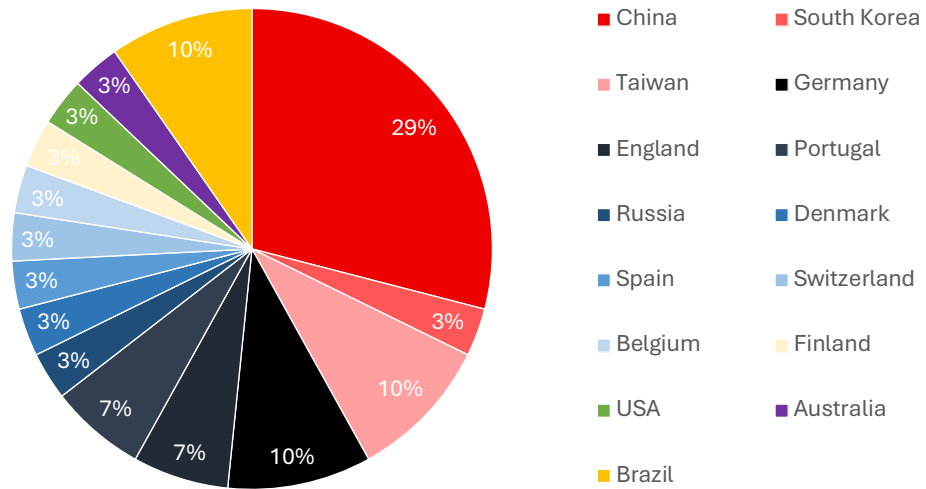


Figure 3 Distribution of papers on fire safety ontologies per country and continent

An analysis of the 31 papers learns that many of these 31 papers are considering fire safety from an operational perspective. Both Guyo et al. (2023) and Wang et al. (2022), for example, use an ontology to help firefighters with indoor building navigation. Others, including Deng et al. (2025), Da et al. (2023) and Neto et al. (2022) use ontology-driven methods to guide occupants during emergency evacuations. Some, including Huang et al. (2022) and Chou et al. (2018) did apply ontologies, but on a city scale. Only a handful of papers directly relates to the subject of compliance checking (Fitkau and Hartmann (2024), Chen and Jiang (2024), Li et al. (2024), and Ul Hassan and Le (2022)). While we could still learn from the other ontologies, they do not sufficiently cover fire safety terms in the built environment to directly support the FireBIM project.

An analysis of the other ontologies was still made performed, by unifying the visual representations to a standard visual representation, where classes related to building information (orange), systems (blue), (non-)human actors (red), rules (dark green) and operations and events (purple) were modeled. Comparing these visual representations gives a first hint of what classes are modeled. A logical finding is that almost all of the analyzed ontologies model building-related terms, such as the building itself, elements, zones and sites, and functions or types. These building-related classes are almost always modeled as central classes in the ontology.

Another notable finding is the high number of very specific properties. Most of the papers that were analyzed seem to be very use case-specific, and the ontologies are likely also designed with a specific use case or project in mind. It is with this mindset that one should evaluate these ontologies, and it is assumed that this is the reason for the existence of these use case-specific terms. Examples are ‘LoadBearingNonEnclosingComponent’, ‘EmergencyServiceProviderContactAddress’, and ‘PsychologicalHarm’. While important for the execution of these use cases, it is questionable how reusable these terms are.

The focus on using ontologies in direct applications also led to some poor modeling practices. Some ontologies contain abbreviations in class names (‘BldUseAdjFactor’, ‘BuildingG’) or even just roman numbers (‘IX’). Some classes refer to specific software components (‘GraphUpdater’).

Of the four ontologies that are directly related to compliance checking, only one is available online as a serialization, namely the Fire Safety Ontology (FiSa) by Fitkau and Hartmann (2024) as Turtle (available at <https://ifitkau.github.io/fisa/>). The ontology models a wide range of classes, properties, and even individuals. FiSa extends the BOT ontology and models terms based on German fire safety regulations. The paper also applies compliance checking. Although not present for all classes, some classes do contain definitions, that sometimes directly originate from German regulatory documents.

3.2. Ontologies for Building Information

Automated compliance checking in the built environment relies on structured, machine-readable representations of physical assets and spatial topologies. Already in 2009, Beetz et al. Published work on IfcOWL, which is an OWL representation of IFC. Over the years, developments in IfcOWL (Pauwels and Terkaj (2016)) emerged. Later, these ontologies were considered too monolithic, and state-of-the-art semantic models typically employ a more modular architecture, where modules extend each other. The Linked Building Data ontologies, and specifically the Building Topology Ontology (BOT) (Rasmussen et al. (2020)), often serves as a core ontology, defining core topological concepts and the spatial hierarchy of a project through terms such as site, building, storey, space, and element.

Building on top of BOT, specialized ontologies act as plug-and-play modules to enrich a knowledge graph with domain-specific terms. These include properties (OPM) (Rasmussen et al., 2020), geometry (OMG, FOG) (Wagner et al., 2019, Bonduel et al., 2019), sensors and performance information (SOSA/SSN, SAREF, BOP) (Janowicz et al., 2019, Daniele et al., 2015, Donkers et al., 2022), and others. FireBIM could learn from the developments of these modules, and could reuse ontology design patterns from these modules (such as the different levels of property modeling as described in Donkers et al. (2022)). Direct application of these ontologies in the FireBIM project will pose problems, as the definitions of the domain-specific terms do not reflect the local context of these terms in practical compliance checking.

3.3. Ontologies for Regulatory Compliance Checking

Translating natural language building codes into machine-executable formats requires ontologies capable of modelling normative logic. A recent book chapter on semantic web technologies for building permitting covers multiple ontologies in this domain (Werbrouck et al., 2026). Not all are directly of use for this project, as some cover general building permit logic, or administrative and process management. For this specific work, we are mostly interested in ontologies that directly support the compliance checking activities, or at least help bridging normative knowledge and building information.

The AEC3PO ontology (Dridi et al., 2024) introduces a comprehensive framework for this by formalizing regulatory documents through the RASE (Requirement, Application, Selection, Exception) pattern. This approach decomposes normative texts and associates them with executable checking logic. There are other ontologies that were designed prior to AEC3PO that help bridging normative knowledge and BIM, such as the Building Assessment Ontology¹ and the Performance Assessment Ontology (Corry et al., 2015). However, these do focus on a wider range of performance metrics, and while they could theoretically be used in the more strict legal compliance checking, that was not their intended design goal.

¹ <https://alexndonkers.github.io/bao/>

4. The FireBIM Ontology Stack

4.1. Overview

The FireBIM Ontology Stack consists of three novel ontologies: the FireBIM Regulation Ontology (FRO), the FireBIM Building Ontology (FBO), and the FireBIM Regulation Quality Ontology (FRQ). Figure 4 shows how these ontologies are placed in the overall process of this work package.

The FireBIM Regulation Ontology is placed between the national regulatory documents and the set of digital regulations that FireBIM aims to produce. The goal of FRO is to semantically describe the structural components of these national documents, so that we can align structural components (or find differences between them), and make them queryable using a single terminology.

The FireBIM Building Ontology lives between the digital regulations and the building information (models), and covers fire safety concepts. The aim of FBO is to bridge fire safety information in national regulations and fire safety information as modelled in BIM.

The FireBIM Regulation Quality Ontology is a small metadata ontology that enables describing the quality of digital regulations. As there will be a multiple types of text-to-SHACL methodologies developed in this project, FRQ aims to add metadata descriptions about the conversion process. FRQ also enables fire safety experts to store opinions on the quality of the machine-readable regulation. Using FRQ, the FireBIM project could deliver so-called FireBIM Approved Digital Rules, to enhance the trust that third parties will have in the digital rulesets.

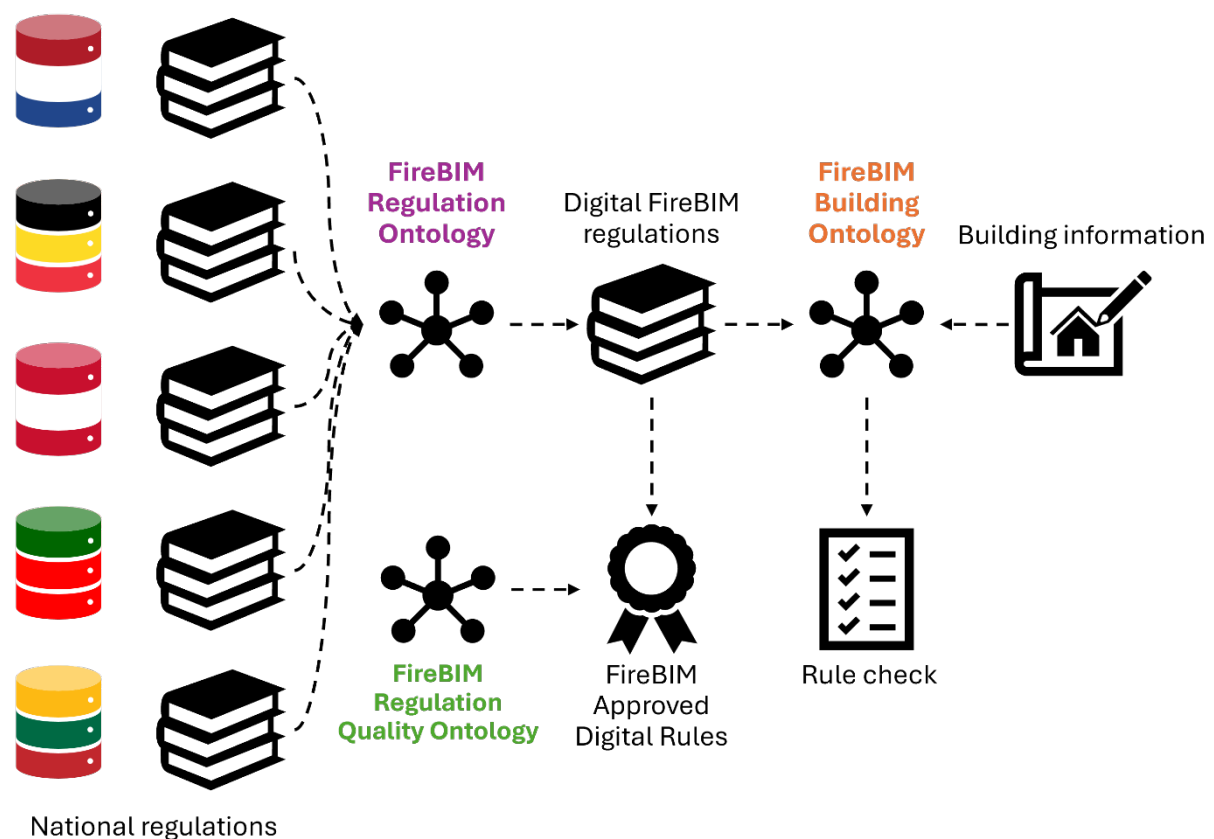


Figure 4 An overview of the ontologies in the FireBIM Ontology Stack, and their role in the overall WP2 tasks

5. FireBIM Regulation Ontology

The FRO ontology aims to cover structural components of regulatory documents, so that these regulatory documents can be represented as RDF graphs. The goal of FRO is to create a terminology that is applicable across regulatory documents in multiple European countries, so that a query returning data from a regulatory document (in an RDF format) looks similar across those countries.

The following subsections describe the methodology that was used to create FRO, and the details of FRO itself.

5.1. Methodology

The development of the ontology follows the following 7 steps: 1. Specification, 2. Knowledge acquisition, 3. Requirement specification, 4. Building, 5. Evaluation, 6. Integration, 7. Documentation and publication.

The specification of the ontology is done in an early stage and is based on the objectives of WP2, namely SO.2.1 and SO.2.2:

- **SO.2.1:** Identify commonalities and variations across international fire codes,
- **SO.2.2:** Harmonize the terminology of the relevant fire assessment codes and regulations,

where we assumed that harmonization would be necessary both on a structural level as well as on a domain knowledge level. Splitting these explicitly enables using the FireBIM ontology stack for new knowledge domains, while having a stable ontology describing structural components of the regulations that are relevant to that knowledge domain.

After specifying the aim of the ontology, knowledge acquisition is performed by performing a systematic literature review on Web of Science, as well as a structured search for ontologies in Linked Open Vocabularies. The results can be found in the State of the Art section.

Requirements are specified based on a set of competency questions, as well as based on the two major requirements for this ontology stack, namely that future additions of new countries and new knowledge domains should be possible. Competency questions for FRO include:

- | | |
|----------------|--|
| FRO.CQ1 | What authority issues a specific legal document, and to what jurisdiction does it apply? |
| FRO.CQ2 | What parts does a legal document consist of? |
| FRO.CQ3 | What are the regulatory statements in a specific section of a legal document? |
| FRO.CQ4 | What is the original text of a specific regulatory statement? |
| FRO.CQ5 | What sections or documents are referenced by a regulatory statement, and what sections or documents does the statement refer to? |
| FRO.CQ6 | What topic is a section of a legal document related to? |
| FRO.CQ7 | What is the version of a document or a document subdivision? |
| FRO.CQ8 | What regulatory document is replaced by the current regulatory document? |

After specifying requirements, the ontology is built. A major challenge was to incorporate (fire safety) domain experts in the ontology design process, as there is a knowledge gap between these experts and ontologists. The ontology development process, therefore, follows an iterative, human-in-the-loop process that aims for collaboration between fire safety experts and ontologists. Based on the requirements, a first ontology was drafted in Stanford Protégé. The ontology was then moved from Protégé into Excel (see Figure 5). The Excel file serves as an intermediate layer that can be understood and adapted by domain experts. Domain experts from the various participating countries were able to copy the ontology to new sheets, after they were requested to perform two actions:

1. Translate terms and descriptions to your local language
2. Add terms and definitions that cover structural components in regulatory documents that are not covered by this version of FRO.

The provided ideas were then incorporated in an update of the FRO ontology. Thanks to the translations, the FRO ontology is now also available in English (en), Dutch (nl), Belgian Dutch (nl-be), Danish (da), Portuguese (pt), and Lithuanian (lt).

The ontology is then evaluated by two methods: first, it was tested whether the concepts in the ontology could be used to answer the competency questions using SPARQL queries. Second, a few regulatory documents from different countries were converted to RDF graphs using the FRO ontology, to test whether it was practically possible to generate complete RDF graphs using the ontology.

	A	B	C	D	E
1	Title			FireBIM Regulation Ontology	
2	URI			FRO	
3	Ontology IRI			http://www.firebim.org/ontologies/fro#	
4	Ontology Version IRI			http://www.firebim.org/ontologies/fro-25022025#	
5					
6	Class	Subclass	Subclass	Title	Description
7	owl:Class			rdfs:label	rdfs:comment
8	fro:Authority			Authority	A legal body with the power to make laws for a given jurisdiction.
9	fro:Document			Document	A record of information.
10	fro:DocumentSubdivision			Document Subdivision	A part of a document.
11		→ fro:Article		Article	A legal rule, constituted by one or more members.
12		→ fro:Member		Member	Part of an article. The exact form of the member is rather arbitrary, and can for example be a sentence.
13		→ fro:Section		Section	Part of a document.
14			→ fro:Appendix	Appendix	A separate part at the end of a document with extra information.
15			→ fro:Chapter	Chapter	Part of a document, usually referred to with a number.
16			→ fro:Equation	Equation	A mathematical statement expressing the equality of two mathematical expressions.
17			→ fro:Figure	Figure	A picture, drawing, or other form of visual information.
18			→ fro:Paragraph	Paragraph	A short part of a text, arranged as a separate block of text.
19			→ fro:Subchapter	Subchapter	A part of a chapter, usually referred to with a number.
20			→ fro:Table	Table	An arrangement of data in rows and columns.
21	fro:Reference			Reference	A mention of something showing where particular information was found.
22	fro:license			License	Document licensed under certain use (not sure of data type, but could be a simple bool, True=Open)

Figure 5 A snippet of an early version of the Excel sheets used to collaborate on FRO

5.2. Ontology overview

Figure 6 shows an overview of the FireBIM Regulation Ontology. It covers six main classes:

- fro:Authority: A legal body with the power to make laws for a given jurisdiction;
- fro:Document: A record of information;
- fro:Jurisdiction: The geographic or political area within which a particular authority has legal power, and within which documents are valid;
- fro:DocumentSubdivision: A part or subdivision of a document;

- fro:Reference: A mention of something showing where particular information was found;
- fro:Topic: A subject, issue or theme that a document or subdivision of a document addresses.

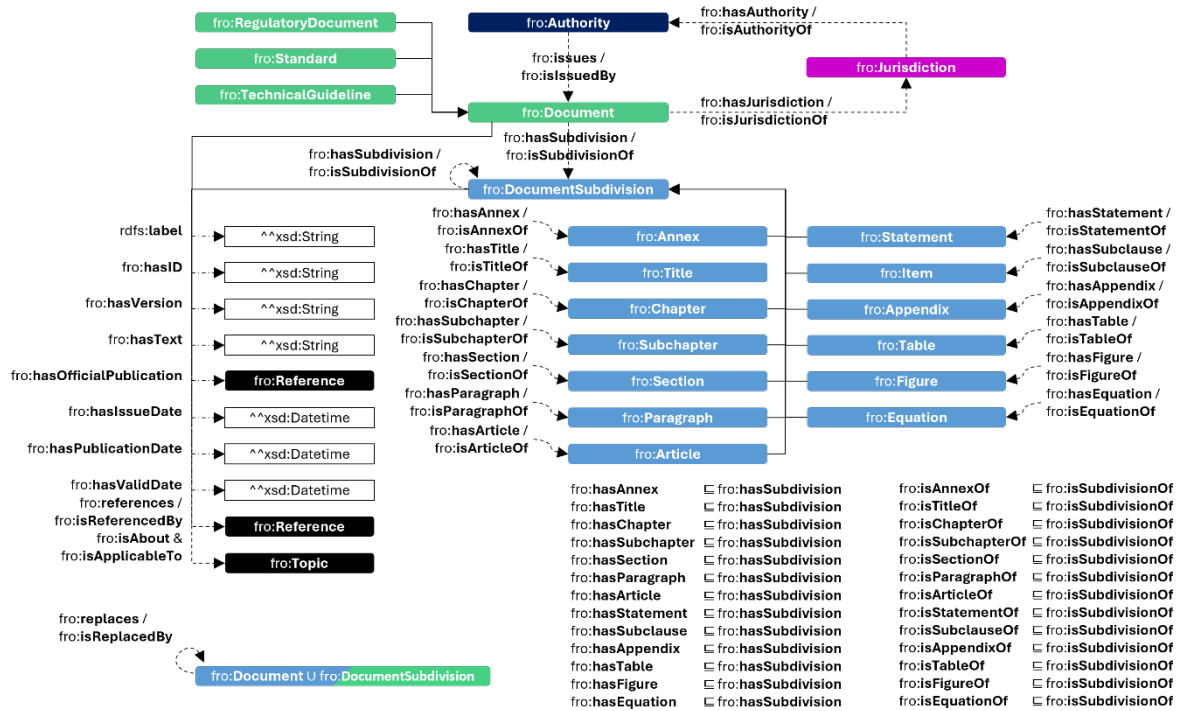


Figure 6 The FireBIM Regulation Ontology

The `fro:DocumentSubdivision` class contains multiple subclasses, covering different types of parts of a document, including `fro:Chapter`, `fro:Paragraph`, `fro:Article`, and `fro:Statement`. These subclasses have incoming and outgoing object properties (e.g. `fro:hasChapter / fro:isChapterOf`) where the `rdfs:domain` and `rdfs:range` are always `owl:Thing`. This modeling decision enables using structural elements rather freely, without imposing a certain hierarchy or order of these structural documents. One legal document can therefore use a Document-Chapter-Paragraph-Article structure, whereas another legal document can use a Document-Appendix-Chapter-Subchapter-Article-Statement structure, while both can be described using the FRO ontology. As all these object properties related to the subclasses of `fro:DocumentSubdivision` are `rdfs:subProperties` of `fro:hasSubdivision / fro:isSubdivisionOf`, one can use these superproperties in SPARQL queries and query parts of legal documents, without knowing the structure of these documents. This enables easier retrieval of legal information, especially when querying with foreign legal documents that are not used on a regular basis (and their structure thus not well known). The properties are `owl:TransitiveProperties`, meaning that is X is a subdivision of Y, and Y is a subdivision of Z, then X is also a subdivision of Z.

The `fro:Document` class currently has three subclasses, `fro:RegulatoryDocument`, `fro:Standard`, and `fro:TechnicalGuideline`. The superclass `fro:Document` enables extending this subclass list in the future, and enables describing more specific types of documents.

Both the `fro:Document` and `fro:DocumentSubdivision` can be described with a list of metadata descriptions, such as an `rdfs:label` and `rdfs:comment`, as well as their `fro:hasText`, their `fro:hasIssueDate`, `fro:hasPublicationDate`, and `fro:hasValidDate`. The `fro:Document` and `fro:DocumentSubdivision` can also be replaced by other documents or document subdivisions using `fro:replaces` / `fro:isReplacedBy`.

Finally, documents and document subdivisions can link to other documents, document subdivisions, or any other type of data, using the `fro:Reference` class. This is often used in regulatory document, e.g. when an article refers to another article as an exception, or when an article refers to an external technical guideline. The `fro:Topic` class enables human experts to link a document or document subdivision to a specific subject, issue, or theme. The class could for example be used to link all articles related to fire compartmentation to a topic `fbo:FireCompartment`, enabling querying the articles based on that topic later on.

Based on the fire experts' input in the Excel sheets, the ontology is translated into multiple languages. Listing 2 shows the TTL representation of the `fro:hasArticle` object property, with the `rdfs:comment` and `rdfs:label` in six languages. The translations enable representing the terms from the ontology in a native language, for example, in user interfaces for regulation discovery.

```
fro:hasArticle rdf:type owl:ObjectProperty ;
               rdfs:subPropertyOf :hasSubdivision ;
               owl:inverseOf :isArticleOf ;
               rdf:type owl:TransitiveProperty ;
               rdfs:range :Article ;
               rdfs:comment "An object property pointing to an article."@en ,
                           "Een 'object property' die verwijst naar een
                           artikel."@nl ,
                           "Een 'object property' die verwijst naar een
                           artikel."@nl-be ,
                           "Et 'object property' der peger på en artikel."@da
                           ,
                           "Objekto savybė, nukreipianti į straipsnį."@lt ,
                           "Uma propriedade de objeto que aponta para um
                           artigo."@pt ;
               rdfs:label "har artikel"@da ,
                          "has article"@en ,
                          "heeft artikel"@nl ,
                          "heeft artikel"@nl-be ,
                          "tem artigo"@pt ,
                          "turi straipsnį"@lt .
```

Listing 2 A TTL representation of the `fro:hasArticle` object property

5.3. Instantiation

Figure 7 shows an instantiation of the FRO ontology, based on the Dutch construction law. The figure shows how the ontology is able to break down the legal document – which is written in natural language – to a hierarchical graph. The top part of the figure splits the document into chapters, sections, paragraphs, and articles. It contains references to the official publication of the specific sections, as well as human-readable labels on each node.

The bottom part of the figure breaks down the legal text into small, operable snippets of text (of type `fro:Statement`). These statement nodes contain the snippet of text from the regulations,

and can contain other metadata if needed. Using `fro:references` / `fro:isReferencedBy`, statements can also refer to other statements explicitly. In Figure 7, one can see how A39L2 refers to A39L1 in text (“The first member does not apply to:”), which is made explicit using the `fro:references` object property. The same principle could also be used to refer to statements (or even entire documents) from external sources, such as technical guidelines or norms.

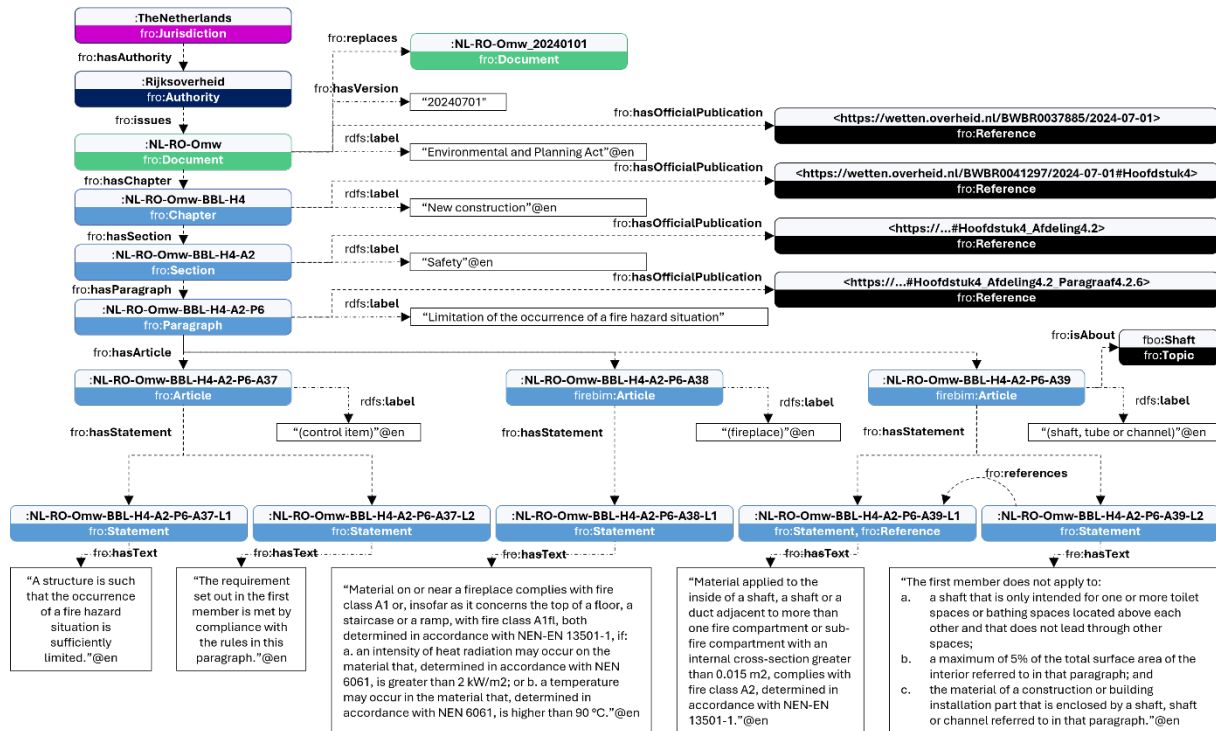


Figure 7 Instantiation of the FRO ontology using the Dutch construction law

A variety of ways to generate this instantiation has been explored within the FireBIM project. A more detailed overview of these methods will be shared in Deliverable 2.3. These methods include rule-based methods, Natural Language Processing-based methods, Large Language Model-based methods, manual tagging approaches, and combinations. A general understanding is that there is no single approach that could be generalized for all legal documents of the project partner countries (and hypothetically also other countries), as these documents come in different formats, each bringing advantages and disadvantages for conversion.

5.4. Answering competency questions

To test the applicability of the ontology in practice, we’re testing whether we can answer the eight competency questions (posted in Section 5.1) using the instance graph in Figure 7. This is done by converting these questions to SPARQL queries, using the terms from the FRO ontology. The subsections below answer these competency questions one by one.

5.4.1. What authority issues a specific legal document, and to what jurisdiction does it apply?

The first question the ontology should enable answering relates to matching a legal document to the responsible authority and the jurisdiction in which it applies. It enables filtering legal documents, for example, per country, as well as comparing legal documents of multiple

countries. Listing 3 presents the SPARQL query that answers this question, and Table 1 shows the results of this query.

```
PREFIX fro: <http://firebim.org/ontologies/fro#>
SELECT ?document ?authority ?jurisdiction
WHERE {
  ?document a fro:Document ;
            fro:issuedBy ?authority ;
            fro:appliesTo ?jurisdiction .
  ?authority a fro:Authority .
  ?jurisdiction a fro:Jurisdiction .
}
```

Listing 3 SPARQL query answering CQ1

Table 1 Results of CQ1

document	authority	jurisdiction
:NL-RO-Omw	:Rijksoverheid	:TheNetherlands

5.4.2. What parts does a legal document consist of?

The second competency question helps the end user to understand the content of the legal document. Listing 4 shows the SPARQL query that uses `fro:hasSubdivision+` to query subdivisions in a transitive fashion (meaning that if a subdivision also has a subdivision itself, it also returns this). Table 2 shows a snippet of the results, which clearly states the different parts and their type in a tabular format.

```
PREFIX fro: <http://firebim.org/ontologies/fro#>
SELECT ?document ?part ?partType
WHERE {
  ?document a fro:Document ;
            fro:hasSubdivision+ ?part .
  ?part a ?partType.
}
```

Listing 4 SPARQL query answering CQ2

Table 2 A snippet of the results of CQ2

document	part	partType
:NL-RO-Omw	:NL-RO-Omw-BBL-H4	fro:Chapter
:NL-RO-Omw	:NL-RO-Omw-BBL-H4-A2	fro:Section
:NL-RO-Omw	:NL-RO-Omw-BBL-H4-A2-P6	fro:Paragraph
:NL-RO-Omw	:NL-RO-Omw-BBL-H4-A2-P6-A37	fro:Article
:NL-RO-Omw	:NL-RO-Omw-BBL-H4-A2-P6-A37-L1	fro:Statement
...	...	...

5.4.3. What are the regulatory statements in a specific section of a legal document?

Competency question 3 asks for all the specific legal statements in a section of the document. Listing 5 shows the query that takes these statements from a section (independent of the hierarchical structure of the document), and table 3 shows the results of this query.

```
PREFIX fro: <http://firebim.org/ontologies/fro#>
PREFIX : <http://example.org/data#>
```

```

SELECT ?section ?statement
WHERE {
  BIND(:NL-RO-Omw-BBL-H4-A2 AS ?section)

  ?section fro:hasSubdivision+ ?statement .
  ?statement a fro:Statement .
}

```

Listing 5 SPARQL query answering CQ3

Table 3 Results of CQ3

section	statement
:NL-RO-Omw-BBL-H4-A2	:NL-RO-Omw-BBL-H4-A2-P6-A37-L1
:NL-RO-Omw-BBL-H4-A2	:NL-RO-Omw-BBL-H4-A2-P6-A37-L2
:NL-RO-Omw-BBL-H4-A2	:NL-RO-Omw-BBL-H4-A2-P6-A38-L1
:NL-RO-Omw-BBL-H4-A2	:NL-RO-Omw-BBL-H4-A2-P6-A39-L1
:NL-RO-Omw-BBL-H4-A2	:NL-RO-Omw-BBL-H4-A2-P6-A39-L2

5.4.4. What is the original text of a specific regulatory statement?

The fourth competency question asks for the regulatory statement in natural language, which corresponds to a snippet of text in the original regulatory document. The result (table 4) of this query (listing 6) can be consumed by humans, and for example be used to print relevant rules in a user interface.

```

PREFIX fro: <http://firebim.org/ontologies/fro#>
PREFIX : <http://example.org/data#>

SELECT ?statement ?text
WHERE {
  BIND(:NL-RO-Omw-BBL-H4-A2-P6-A37-L1 AS ?statement)

  ?statement fro:hasText ?text .
}

```

Listing 6 SPARQL query answering CQ4

Table 4 Results of CQ4

statement	text
:NL-RO-Omw-BBL-H4-A2-P6-A37-L1	"A structure is such that the occurrence of a fire hazard situation is sufficiently limited."@en

5.4.5. What sections or documents are referenced by a regulatory statement, and what sections or documents does the statement refer to?

The fifth competency question aims to help exploring references of specific statements. Statements can contain references to other statements themselves, or can be referenced by another statement. The SPARQL query in listing 7 returns both types of references, and the result could be seen in table 5. The `fro:isReferencedBy` is inferred, given that `fro:references` and `fro:isReferencedBy` are inverse properties (using `owl:inverseOf`).

```

PREFIX fro: <http://firebim.org/ontologies/fro#>

SELECT ?statement ?references ?isReferencedBy

```

```

WHERE {
  ?statement a fro:Statement .

  OPTIONAL {
    ?statement fro:references ?references .
  }

  OPTIONAL {
    ?statement fro:isReferencedBy ?isReferencedBy .
  }

  FILTER (BOUND(?references) || BOUND(?isReferencedBy))
}

```

Listing 7 SPARQL query answering CQ5

Table 5 Results of CQ5

statement	references	isReferencedBy
:NL-RO-Omw-BBL-H4-A2-P6-A39-L1		:NL-RO-Omw-BBL-H4-A2-P6-A39-L2
:NL-RO-Omw-BBL-H4-A2-P6-A39-L2	:NL-RO-Omw-BBL-H4-A2-P6-A39-L1	

5.4.6. What topic is a section of a legal document related to?

Competency question 6 returns the topics that a specific part of a document (or the document itself) is about. It enables experts to add a ‘softer’ layer of communication on top of the legal documentation, and for example group sections that are related to a specific topic (e.g. `fbo:FireCompartment`). Listing 8 shows a SPARQL query returning the topic of a statement in the instance graph, and table 6 shows the result of this query.

```

PREFIX fro: <http://firebim.org/ontologies/fro#>
SELECT ?section ?topic
WHERE {
  ?section fro:isAbout ?topic .
}

```

Listing 8 SPARQL query answering CQ6

Table 6 Results of CQ6

section	topic
:NL-RO-Omw-BBL-H4-A2-P6-A39	fbo:Shaft

5.4.7. What is the version of a document of document subdivision?

The seventh competency question, that is presented as a SPARQL query in listing 8, aims to return the version of a document or a part of the document. It enables to perform version control of legal documents, and track changes between sections of documents over time. Table 7 shows how the version of a specific document is returned.

```

PREFIX fro: <http://firebim.org/ontologies/fro#>
SELECT ?document ?version
WHERE {
  ?document a fro:Document ;
    fro:hasVersion ?version .
}

```

```
}
```

Listing 9 SPARQL query answering CQ7

Table 7 Results of CQ7

document	version
:NL-RO-Omw	"20240701"

5.4.8. What regulatory document is replaced by the current regulatory document?

The final competency question is also related to version control. The SPARQL query in listing 10 returns documents that replaced older documents, for example when a new version of the regulations is published. Table 8 returns both the current version of the regulatory document, as well as the previous version. By enabling to make these relationships explicit, we could not only track versions of these legal documents, but also of the digital rulesets that are linked to these parts of the knowledge graph.

```
PREFIX fro: <http://firebim.org/ontologies/fro#>
SELECT ?currentDocument ?oldDocument
WHERE {
    ?currentDocument a fro:Document ;
                     fro:replaces ?oldDocument.
}
```

Listing 10 SPARQL query answering CQ8

Table 8 Results of CQ8

currentDocument	oldDocument
:NL-RO-Omw	:NL-RO-Omw_20240101

6. FireBIM Building Ontology

The FireBIM Building Ontology (FBO) aims to cover national fire safety terms and their legal definitions. The ontology aims to serve as a bridge between terminology in legal documentation and (BIM) modeling practice. Another goal is to bridge terminology between different countries, or at least make similarities and differences more explicit and visible.

The following subsections describe the methodology that was used to create FBO, and the details of FBO itself.

6.1. Methodology

As the FBO ontology aims to cover fire safety knowledge from multiple countries, a rigorous method was needed to capture all this knowledge. The project follows the ontology development method developed by Donkers et al. (2022), consisting of the following 7 steps: 1. Specification, 2. Knowledge acquisition, 3. Requirement specification, 4. Building, 5. Evaluation, 6. Integration, 7. Documentation and publication. The method considers these steps as non-linear, where one would typically iterate multiple times over step 2-5. This iteration was necessary, based on a few givens:

- There is a wide variety of project partners, as well as potential users, that have different expertise, speak different jargon, and work on different activities. An iterative approach is necessary for people to understand each other and to find miscommunications and semantic differences.
- The project itself takes an iterative standpoint based on use cases, where it should be possible to add new use cases using a similar methodology later on. This imposes that the ontology should be able to grow or change over time.
- Not all partners start this project at the same starting date. Terminology from new countries will come in at later stages, and might impose changes to the entire ontology or generally the entire way we think about the role of this ontology as a bridge between legal documents and BIM files.

The project started with a stakeholder workshop in Lund, Sweden, in which fire safety experts, BIM experts, and semantic web experts from four countries (Belgium, Denmark, Portugal, and the Netherlands) participated. The goal of this workshop was to define the scope and purpose of the ontology (*Specification*). The workshop contained in-depth sessions on the specific tasks of a fire safety expert during the design process, to better understand the role this ontology plays in broader software solutions later in the process. The workshop resulted in the decision that this ontology will be developed specifically for fire safety rule checking in early design phases of a project, and that the first iterations of software products will focus on building codes related to compartmentation and escape routes. These decisions were translated into competency questions.

As a result of this stakeholder workshop, the fire safety experts from these four countries started building a Fire Code Matrix. This Fire Code Matrix is a collection of (fire-specific) terms and definitions that are used in building codes specific to the scope of this project. Next to these terms and definitions, the Fire Code Matrix also collects flow charts (using the Decision Modeling Notation style) that capture the decision-making process of a fire safety expert for a specific rule. These flow charts represent the (process) knowledge that should be mimicked by a software solution using the ontology.

Based on this Fire Code Matrix, a literature review is performed to find the existing ontologies related to the terms in the matrix. Ontologies were found using three methods:

- A Web of Science search;
- A search on Linked Open Vocabularies using terms in the Fire Code Matrix;
- A search for projects, commercial tools and research activities that were known by members of the consortium.

Ontologies that were found in this search were then visualized as graphs to enable evaluating whether the ontologies covered the terms and definitions in the Fire Code Matrix and were able to fulfill the desired software functionalities. This concluded the *Knowledge acquisition* phase. A first version of the FBO ontology was created using Stanford Protégé.

A second stakeholder workshop (with the same composition of participants) was held in Eindhoven, the Netherlands, in which the Fire Code Matrix and the first version of the FBO ontology were presented. The goal of the ontology at this stage was mostly to communicate with all participants what an ontology is, and how it can be used to structure (fire safety) expert knowledge. The goal of the workshop was to identify the most important requirements for the ontology from an end user perspective, and to identify how the ontology would be aligned with

existing standardization initiatives. The result of this workshop is a list of requirements (*Requirement specification*).

Based on the identified requirements, a second literature review is performed, this time related to OWL. Especially the international alignment of terms posed challenges that were difficult to solve with regular ontology modeling patterns, hence the investigation whether OWL has constructs that would solve these challenges. This resulted in a document with modeling patterns.

This document, the requirements and competency questions, and the previous ontology were input for the redevelopment of the FBO ontology in Stanford Protégé (*Building*). This ontology was published on a temporary web location so that it could be used to develop demonstrators (*Evaluation*). A range of software tools was built on top of the FBO ontology to perform various tasks related to building codes about compartmentation and escape routes. During these two phases, various smaller (online) workshops were organized to quickly iterate intermediate results.

An updated FBO ontology and the demonstrators were presented in a third stakeholder workshop in Porto. Compared to the previous workshops, this third workshop had an extra participating country, being Lithuania. In this workshop, discussions related to the ontology revolved around:

1. the mapping with the IFC schema, bSDD concepts, ISO terms, and W3C ontologies;
2. the representation of national and international fire safety knowledge;
3. the ability to help software tools in fulfilling demonstrator tasks.

As a result of this workshop, the Fire Code Matrix was updated and the relevant Lithuanian terms and definitions were added. Based on the feedback from the third stakeholder workshop and the updated Fire Code Matrix, the FBO ontology was redeveloped into a stable version, which was published online with documentation (*Documentation and publication*). Alongside the FBO ontology, this online documentation contains alignment modules with existing standards (*Integration*). Finally, demonstrators were updated to match the stable ontology.

6.2. Requirements and competency questions

The FireBIM Building Ontology aims to represent terms related to fire safety-specific building codes, in both a harmonized European set of terms as well as specific national representations of those terms, in order to bridge semantics between fire safety regulations and building information models. To design this ontology, a set of requirements is developed. These come in three forms. First, a set of ontology requirements is listed. Then, we made use of competency questions as a guidance for the design (and later validation) of the ontology. Given that the intended use of this ontology is compliance checking and given that compliance checking would best be performed under the closed world assumption, SHACL is the language in which FBO will be used most often.

6.2.1. Ontology requirements

REQ1 Digital rules should be both machine-readable and machine-understandable.

REQ2 Terms should be abstract enough to be usable across European borders and simultaneously be detailed enough to be valuable in national or even local automated compliance checking.

REQ3 Terms must, where possible, be aligned with existing standards from various domains, such as ISO 13943:2023 and ISO 19650.

REQ4 The ontology should be compatible with IFC4x3, however, IFC should not be inferred by default.

REQ5 The ontology should enable multiple methods to create RDF-based building information, including the use of different software tools.

REQ6 The ontology should be extendible to other countries and to other application domains.

6.2.2. Competency questions

1. What is the definition of a fire safety term in a national regulation?
2. How is a quantity calculated in a national regulation?
3. What is the preferred IFC4.3 method to model a fire safety term in a national regulation?
4. What is the ISO equivalent of a term in a national regulation?
5. Does country A have terms that are similar to term X in country B?
6. What is the unit of a fire safety property?

6.3. Ontology overview

The FBO ontology consists of a set of modular ontologies that, as a whole, aims to fulfil the requirements and answer the competency questions and constraints. Figure 8 gives a high-level overview of the FBO ontology and its modules. The core principle of FBO is to use an overarching European ‘umbrella’ ontology (FBO) and national subontologies (FBO-BE, FBO-DK, FBO-LT, FBO-NL, FBO-PT, ...), which we later refer to as *vertical modularization*. These ontologies internally consist of modules as well, as can be seen in Figure 8 under the European FBO ontology, which is referred to as *horizontal modularization*. Finally, the ontology has separate *alignment modules*. These three concepts are explained below, after which a more detailed look into the ontology is given.

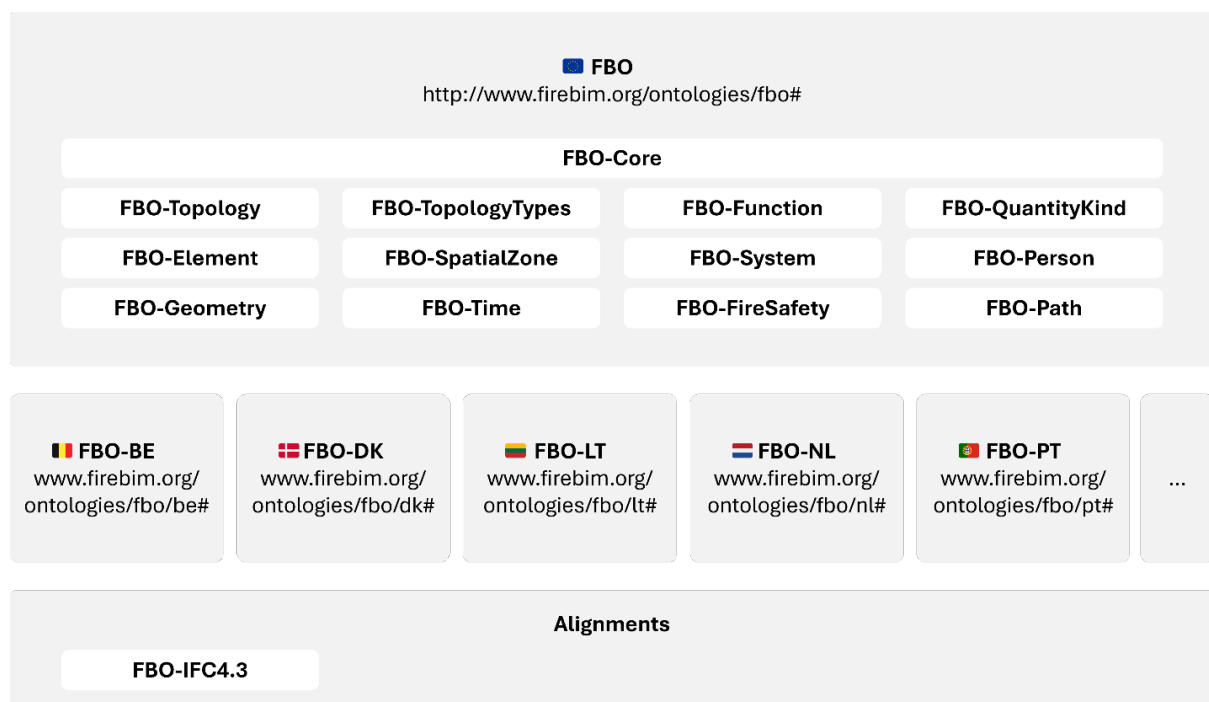


Figure 8 High-level overview of the FBO ontology

6.3.1. Horizontal modularization

The FBO ontology covers harmonized European terms in the <http://www.firebim.org/ontologies/fbo#> namespace. This namespace functions as one ontology, built up from 13 individual modules. The main module is FBO-Core, which aligns all other modules using high-level terms and object properties. The other modules mostly function as taxonomical extensions of FBO-Core. As all 13 modules reside in the same namespace, an end-user does not need to be aware of these, and both FBO-Core terms and terms from the other modules can be called using the fbo: prefix.

The main reason for modularization is the second half of REQ6 (“The ontology should be extendible [...] to other application domains”). By thematically modularizing the FBO ontology, developments (both on the ontology itself and on applications) can take place within the boundaries of such themes, and specific experts on these themes (for example, ‘geometry’) can then develop and maintain such a thematic module without diluting the others.

6.3.1.1. FBO-Core

FBO-Core is the main module in the FBO ontology, and gives a high-level overview of abstract concepts and how they relate to each other. Through these relationships, FBO-Core links the other modules. Figure 9 gives an overview of the ontology.

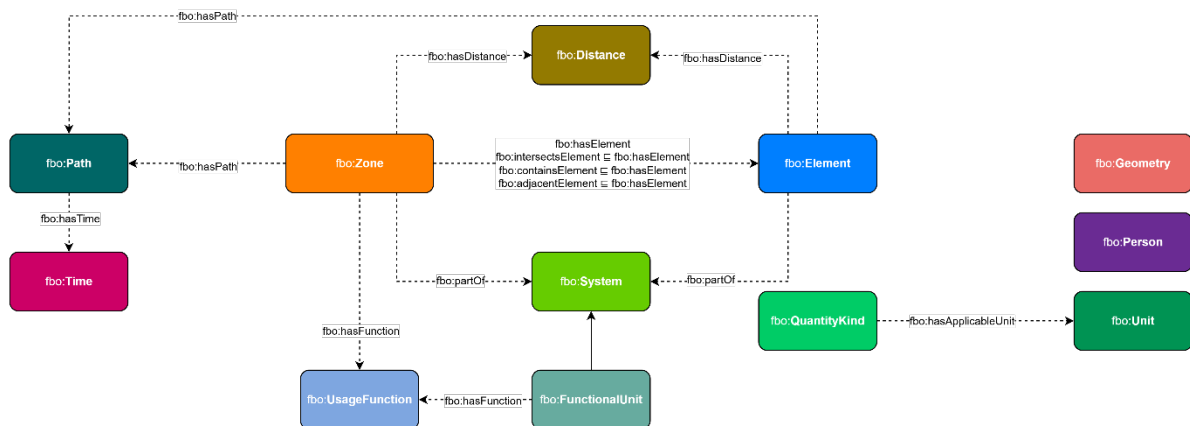


Figure 9 An overview of FBO-Core

FBO-Core is designed in the unique vacuum between BIM knowledge (where we often refer to IFC) and fire safety knowledge. REQ5 demands that “the ontology should enable multiple methods to create RDF-based building information, including the use of different software tools”. IFC is an obvious standard to align to in the domain of building information modeling (hence REQ4 “The ontology should be compatible with IFC4x3 [...]”), however, IFC is not the core way of thinking or conceptually modeling fire safety knowledge, and some fire safety concepts are not present in IFC. Depending on the modeler or the modeling phase, the application of certain terms also changes, and IFC does not always allow for this. One example is the fire compartment.

In an early design stage, fire compartments are typically (conceptually) designed as lists, where the list items are spaces. This method is quick, as a modeler in an early design stage could simply assign all spaces that need to be in a compartment to a relevant fire compartment, and then run a quick check on whether the design of these compartments makes sense. It is typically precise enough for early design stages and doesn’t bother the modeler with a heavy workload. Speaking in IFC terms, the fire compartment would be an *IfcZone*, and the spaces would be of type *IfcSpace*. However, compliance checking in later

design stages typically requires a more precise definition of the sizes of these compartments, which conceptually means that a designer would create a 3D element with geometric properties that exactly covers the fire compartment. IfcZones, however, cannot have a 3D geometric representation, thus, to model such compartment in IFC one could use IfcSpatialZone.

The example shows why the concept ‘fire compartment’ cannot, or should not, be mapped against a specific term in IFC, as it is used differently by fire experts in different stages of the design process. There is no clear distinction on when to use which IFC concept, and this (anecdotally) differs from country to country and from office to office. The choice is also software-dependent; for example, ArchiCAD natively does not have a button that enables modelers to create an IfcSpatialZone. It is therefore that the second part of REQ4 reads “however, IFC should not be inferred by default.”

FBO-Core therefore serves as an intermediary, that on a high level tries to group different fire safety terms, and enables mapping to IFC in a separate alignment module. It contains a couple of classes. The fbo:Element class groups all things that are physical elements, such as walls, doors, and floor slabs. It highly overlaps with IfcElement, but arguably extends it, and could best be described as a building element as defined in ISO 6707-1:2020(en) 3.3.5.4.

The counterpart of the fbo:Element is the fbo:Zone. The concept overlaps with how zone is defined in the BOT ontology (bot:Zone) and does not map to IfcZone, IfcSpatialZone, or IfcSpatialElement directly, but can be used to represent one of those. An fbo:Zone can contain, intersect, or be adjacent to an fbo:Element.

FBO-Core has an fbo:System class to represent systems. It enables both elements and zones to be part of a system, which could be defined as an ISO 12006-2:2015 3.1.3. construction system: “interacting construction objects organized to achieve one or more purposes”. Functional units are represented using fbo:FunctionalUnit and are subclasses of fbo:System. Such functional units have functions (fbo:UsageFunction). Zones in FBO are, different than zones in IFC, not necessarily considered systems, but can also have an fbo:UsageFunction.

FBO enables modeling distances, and does so using the fbo:Distance class, where :DoorX :hasDistance :DistanceZ and :DoorY :hasDistance :DistanceZ to define the distance :DistanceZ between :DoorX and :DoorY. As object properties in RDF graphs cannot contain properties themselves, there is not more efficient way to model distances in RDF graphs currently. Figure 10 shows the instantiation of the fbo:Distance class. The distance node in the RDF graph can then contain further information, such as the length and unit of this distance.

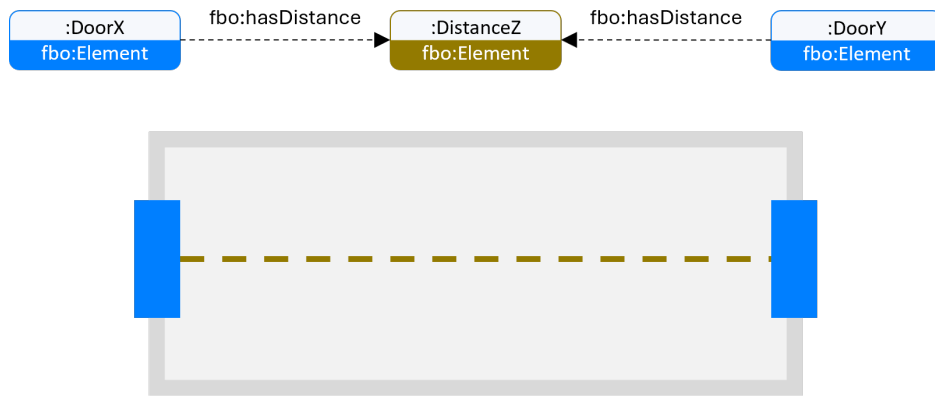


Figure 10 Instantiation of the *fbo:Distance* class

Evacuation routes are modeled similarly, inspired by the IndoorGML concept of paths, states, and transitions. Elements (*fbo:Element*) would typically be a transition, and spaces (subclass of *fbo:Zone*) would typically be a state. Paths are modeled directly between two elements, using the *fbo:startsPath* and *fbo:endsPath* to indicate the direction of the path. Evacuation routes can be modeled with the *fbo:EvacuationRoute* class which is a subclass of *fbo:Path* and essentially groups paths into a route. Figure 11 shows an instantiation of the path class.

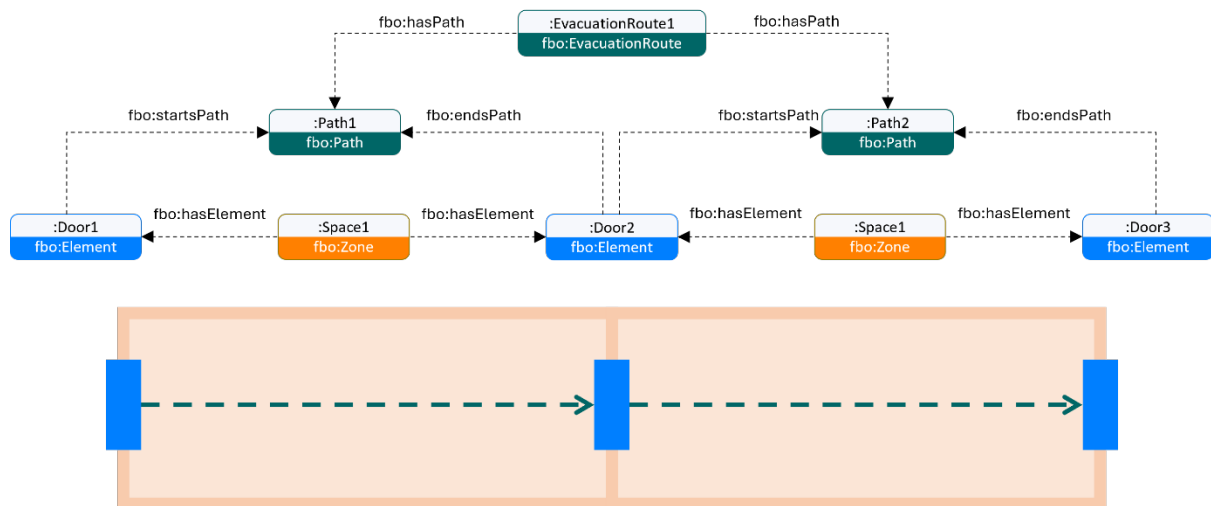


Figure 11 Instantiation of *fbo:Path*

6.3.1.2. FBO-Topology

The first module in FBO is the FBO-Topology module (Figure 12), that introduces a set of subclasses of *fbo:Zone*. It follows the logic of the BOT ontology in structuring topological entities, and contains a class for sites, buildings, storeys, spaces, and roads, as well as for spatial zones. These classes all have an incoming object property (*fbo:Space* has the incoming object property *fbo:hasSpace*) that are subproperties of *fbo:containsZone*. These object properties have no strict *rdfs:domain*, but instead use *schema:domainIncludes*. This still indicates how the object property should be used, without inferring the class of the subject of a triple. Zones can contain zones themselves, as well as be adjacent to other zones or intersect with other zones. The *fbo:containsZone* object property is transitive, meaning that (Listing 11):

```
if
fbo:Building fbo:hasStorey fbo:Storey
and
```

```
fbo:Storey fbo:hasSpace fbo:Space
then we infer that
fbo:Building fbo:containsZone fbo:Space
```

Listing 11 Inferencing with the transitive fbo:containsZone

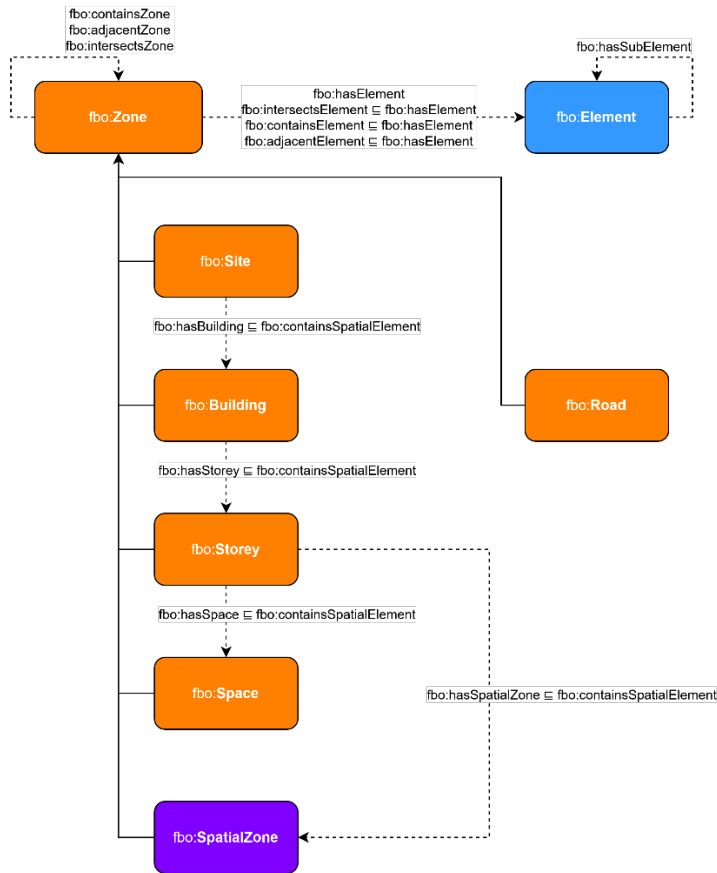


Figure 12 An overview of the FBO-Topology module

6.3.1.3. FBO-Element

A couple of modules, such as the FBO-Element module (Figure 13), are merely taxonomical trees with, in the case of FBO-Element, specific types of elements. These specific types of elements are modelled as subclasses of `fbo:Element`. The tree can grow multiple layers deep, as these subclasses can have subclasses themselves.

By modeling all elements in such a tree structure, the FBO ontology can incorporate new elements in the future by adding them to the tree. The FBO-Element module currently does not contain specific object properties, however, if necessary, they could be added here. One could think of developing an `fbo:containsBeam` object property as a subproperty of `fbo:containsElement` with `rdfs:range fbo:Beam`. Such properties would enable writing cleaner SPARQL queries.

Finally, the FBO-Element module contains the property `fbo:hasMaterial` and class `fbo:Material`, to enable indicating the material of an element.

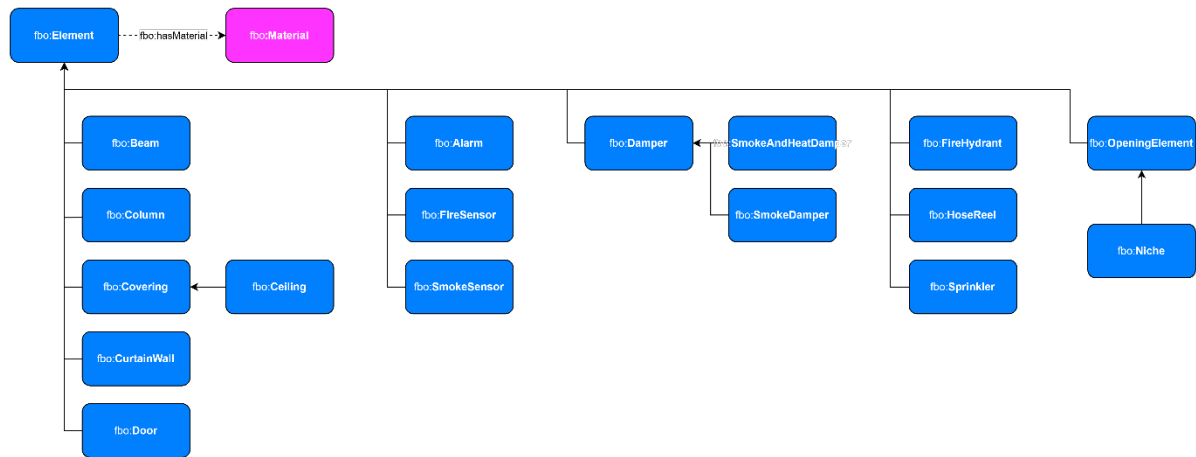


Figure 13 A snippet of the FBO-Element module

The FBO-Element module contains certain rules to help inferencing of specific types of elements. For example, an `fbo:Wall` with a property `fbo:isCompartmentationElement` `True` is assumed to be an `fbo:CompartmentBoundaryWall`. Listing 12 shows an example of how this reasoning is encoded in the ontology.

```
fbo:CompartmentBoundaryWall
...
owl:equivalentClass [
  rdf:type owl:Class ;
  owl:intersectionOf (
    fbo:Wall
    [
      rdf:type owl:Restriction ;
      owl:onProperty fbo:isCompartmentationElement ;
      owl:hasValue true
    ]
  )
] .
```

Listing 12 RDF snippet showing how walls that are compartmentation elements are inferred to be compartment boundary walls

6.3.1.4. FBO-Function

The FBO-Function module lists both subclass trees for functional units and usage functions, as these are often related. Figure 14 shows how both the functional unit class and the usage function class have their respective subclass trees, and how some of those are explicitly linked using `fbo:hasFunction`.

This module also integrates reasoning, with the goal of reducing the amount of manual work of a (BIM) modeler. The module for example contains rules that any `owl:Thing` that has a `fbo:ParkingFunction` should be classified as an `fbo:Parking`. Listing 13 shows how these rules are modelled.

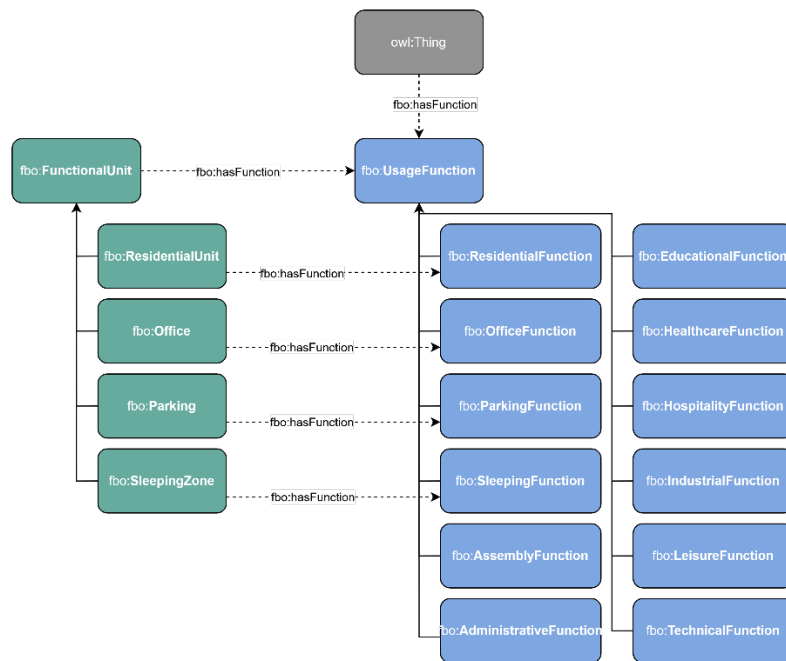


Figure 14 An overview of the FBO-Function module

```
fbo:Parking owl:equivalentClass [
  a owl:Restriction ;
  owl:onProperty fbo:hasFunction ;
  owl:someValuesFrom fbo:ParkingFunction
] .
```

Listing 13 Any owl:Thing that has an fbo:ParkingFunction is classified as an fbo:Parking

6.3.1.5. FBO-System

The FBO-System module expands upon the core fbo:System class by providing a dedicated taxonomic tree for specific building and fire safety systems. Figure 15 gives an overview of this module, illustrating how active and passive safety configurations are classified as direct subclasses of fbo:System.

These subclasses include essential fire protection configurations such as fbo:FireAlarmSystem, fbo:FireExtinguishingSystem, fbo:SmokeAlarmSystem, and fbo:WarningSystem, alongside structural or spatial aggregations like fbo:Facade and fbo:EscapeRouteSystem. Because FBO-Core establishes that both physical elements (fbo:Element) and spatial zones (fbo:Zone) can be linked to a system using the fbo:partOf object property, the FBO-System module allows modelers to semantically group disparate building components that collectively achieve a specific safety purpose.

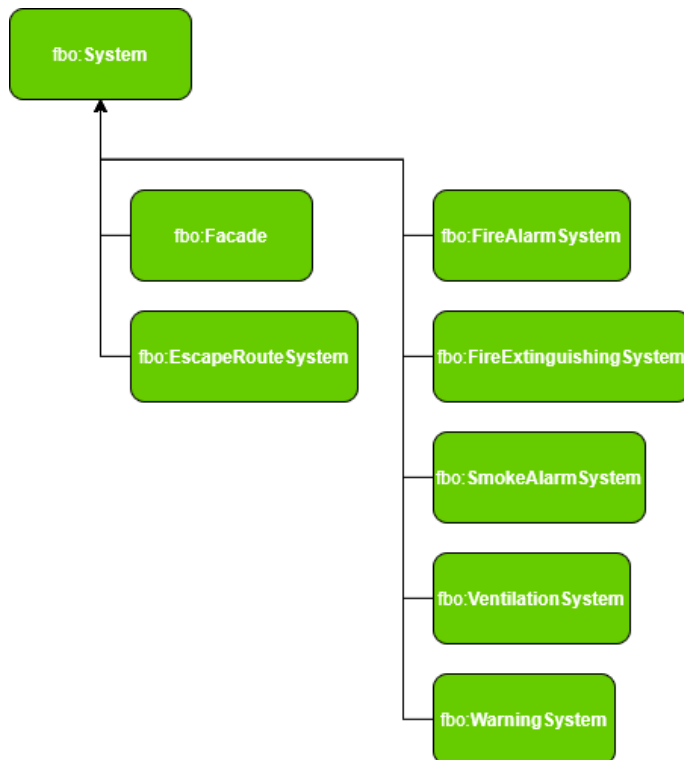


Figure 15 An overview of FBO-System

This structural aggregation is particularly useful for compliance checking, as regulations often impose requirements on the system as a whole rather than on individual isolated components. For example, a modeler can group various sensors, alarms, and control panels into a single `fbo:FireAlarmSystem`.

To demonstrate how this modularization benefits data retrieval, Listing 15 provides a SPARQL query that retrieves all physical elements belonging to any fire extinguishing system within the graph.

```

PREFIX fbo: <http://www.firebim.org/ontologies/fbo#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?system ?element
WHERE {
  ?system a ?systemType .
  ?systemType rdfs:subClassOf* fbo:FireExtinguishingSystem .
  ?element fbo:partOf ?system .
  ?element a ?elementType .
  ?elementType rdfs:subClassOf* fbo:Element .
}

```

Listing 14 SPARQL query retrieving all physical elements that are part of a fire extinguishing system

6.3.1.6. FBO-QuantityKind

The FBO-QuantityKind module is a direct mapping of terms from QUDT into the FBO prefix. It expresses physical quantities and their associated units of measurement, which are required for evaluating quantitative fire safety regulations. Figure 16 gives an overview of this module, highlighting the two primary root classes `fbo:QuantityKind` and `fbo:Unit`. This module can be used to express in which unit a specific property should be modeled for it to be checked against a digital

rule, and to automate the conversions of measurements from other units to those that are expected.



Figure 16 An overview of FBO-QuantityKind

6.3.1.7. FBO-Person

The FBO-Person module provides a compact taxonomic structure to represent human actors within the FireBIM ecosystem. Figure 17 illustrates the module, which introduces fbo:Person as the root class, and fbo:Expert and fbo:Occupant as subclasses. These subclasses can be used by the national ontologies to model specific types of persons, as a variety of occupants and other persons using the building are described in the different national regulations.

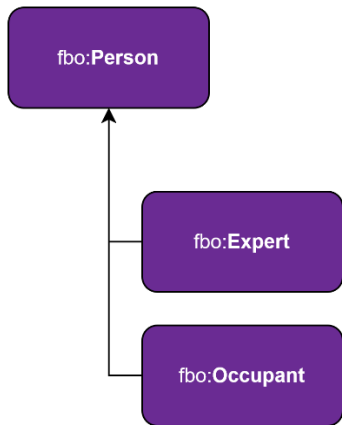


Figure 17 An overview of FBO-Person

6.3.1.8. FBO-Distance

FBO enables modeling distances, and does so using the `fbo:Distance` class, where `:DoorX :hasDistance :DistanceZ` and `:DoorY :hasDistance :DistanceZ` to define the distance `:DistanceZ` between `:DoorX` and `:DoorY`. As object properties in RDF graphs cannot contain properties themselves, there is not more efficient way to model distances in RDF graphs currently. Figure 19 shows the instantiation of the `fbo:Distance` class. The distance node in the RDF graph can then contain further information, such as the length and unit of this distance. The module further specifies the concept of `fbo:TravelDistance` as a subclass of `fbo:Distance` (see Figure 18).

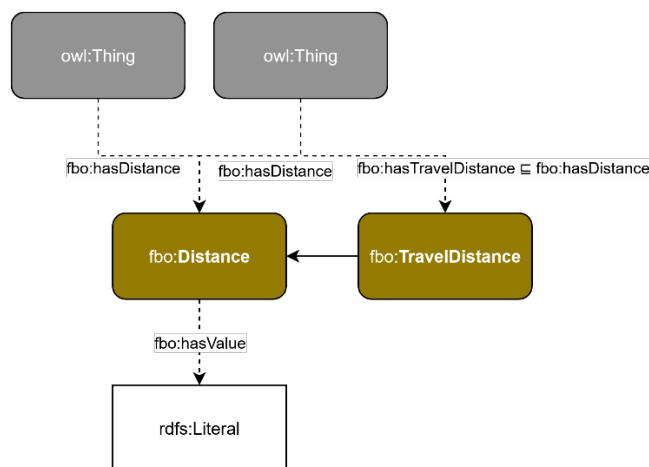


Figure 18 An overview of FBO-Distance

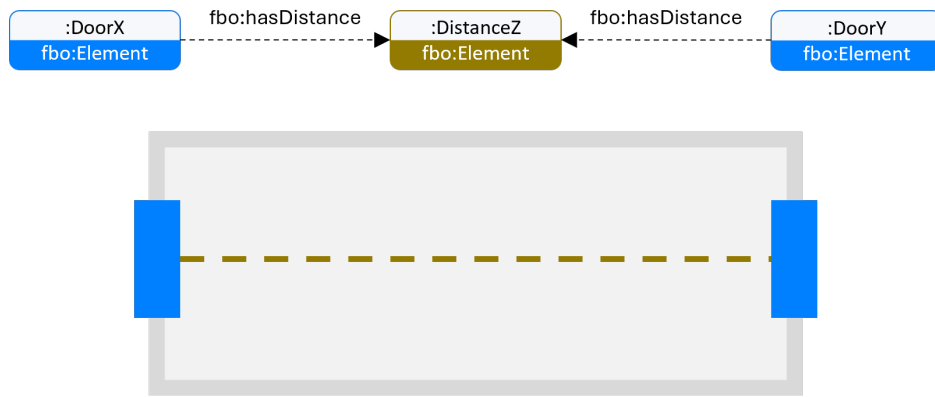


Figure 19 Instantiation of the *fbo:Distance* class

6.3.1.9. FBO-FireSafety

The FBO-FireSafety module is mostly a set of properties that can be used to add fire safety related information to objects in the model. Figure 20 shows a snippet of this module. Some properties are linked to a specific class of the ontology, for example, *fbo:isCollectiveSpace* is linked to *fbo:Space*, as we assume that this property should only be used for that specific class (or subclasses of it). Some properties are linked to specific elements (*fbo:isEmergencyStaircase* is linked to *fbo:Staircase*), whereas other properties are linked to a higher abstraction (*fbo:isMovable* is linked to *fbo:Element*). This is because multiple types of elements can be movable, and inferring a specific type might cause issues with reasoning in future cases. Some properties are linked to *owl:Thing* (and thus don't have an *rdfs:domain* specified), however, they typically contain the softer *schema:domainIncludes* to indicate potential subjects for a triple. For example, *fbo:hasAutomaticFireExtinguishingSystem* *schema:domainIncludes* *fbo:Space*, so that the ontology suggests that this property could be used for spaces. A reasoner would in that case, however, not infer the subject to be of type *fbo:Space* as other types of zones might also have an automatic fire extinguishing system.

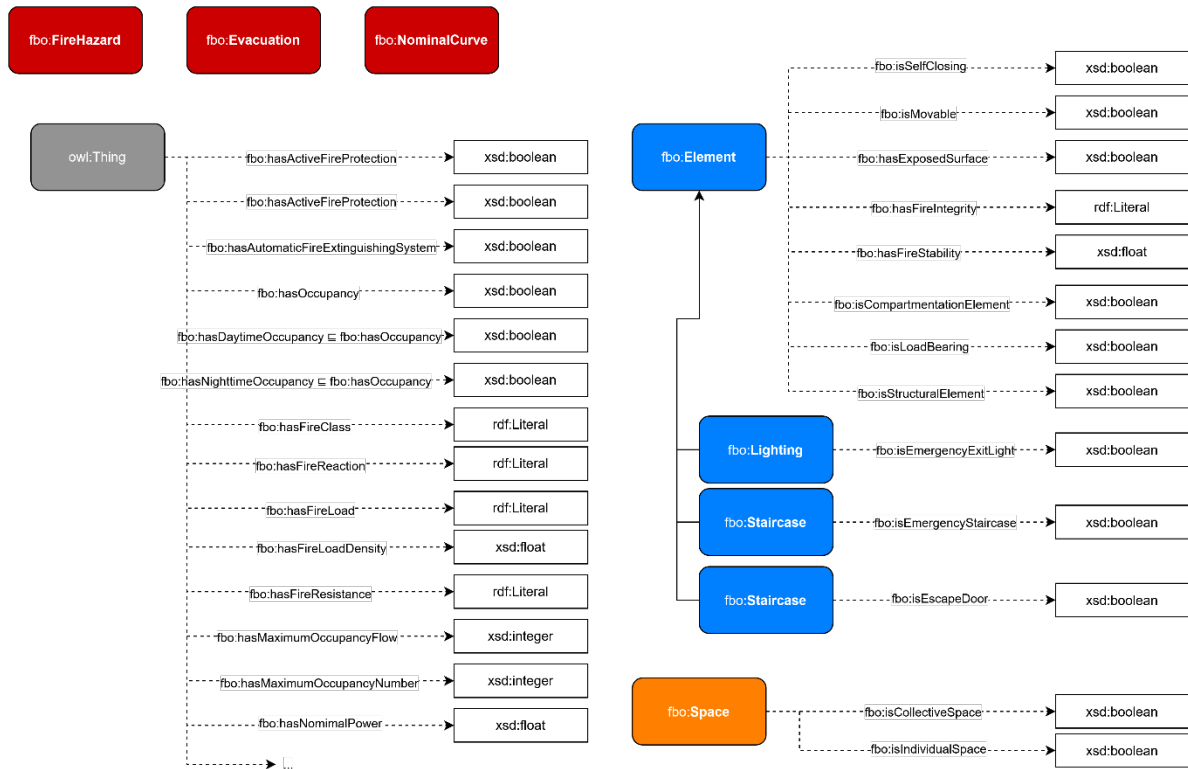


Figure 20 A snippet of the FBO-FireSafety module

6.3.1.10. FBO-Path

Evacuation routes are modeled similarly, inspired by the IndoorGML concept of paths, states, and transitions. Elements (`fbo:Element`) would typically be a transition, and spaces (subclass of `fbo:Zone`) would typically be a state. Paths are modeled directly between two elements, using the `fbo:startsPath` and `fbo:endsPath` to indicate the direction of the path. Evacuation routes can be modeled with the `fbo:EvacuationRoute` class which is a subclass of `fbo:Path` and essentially groups paths into a route. Figure 21 shows the FBO-Path module, and Figure 22 shows an instantiation of the path class.

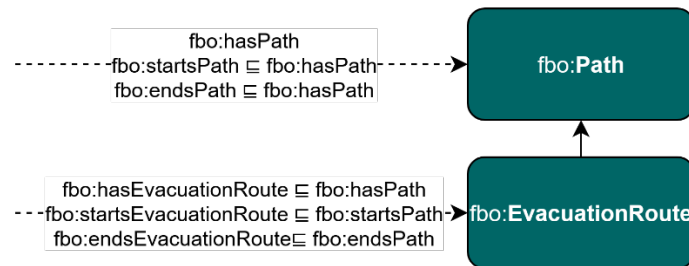


Figure 21 An overview of FBO-Path

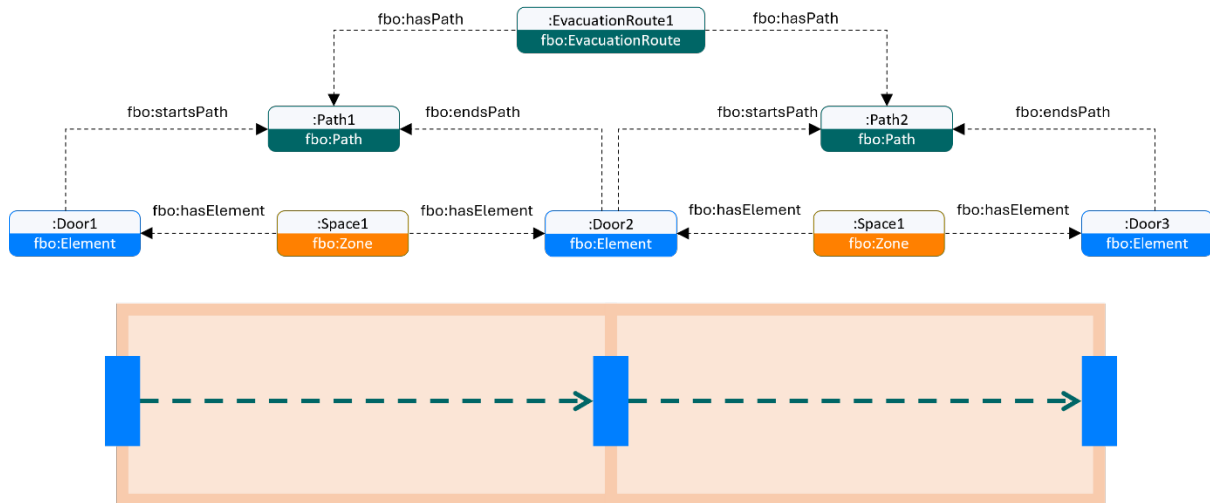


Figure 22 Instantiation of *fbo:Path*

6.3.1.11. FBO-Time

FBO-Time provides a small taxonomy for representing temporal constraints in fire safety engineering. Figure 23 illustrates *fbo:Time* as the root class, and *fbo:EvacuationTime* and *fbo:WalkingTime* as subclasses. These subclasses are the *rdfs:range* of *fbo:hasEvacuationTime* and *fbo:hasWalkingTime*. An *rdfs:domain* is not specified for these object properties, so that they can be used flexibly.



Figure 23 An overview of FBO-Time

6.3.1.12. FBO-Geometry

The FBO-Geometry module provides a framework to assign geometric properties, or quantities, as well as explicit geometries to physical elements and topological zones. Figure 24 provides an overview of the module. Given diverse requirements of fire safety engineering, ranging from simple parametric compliance checks to complex 3D evacuation simulations, the module supports multiple levels of geometric representation without strictly enforcing a single standard. It is modeled based on the W3C LBD OMG ontology by Wagner et al. (2019).

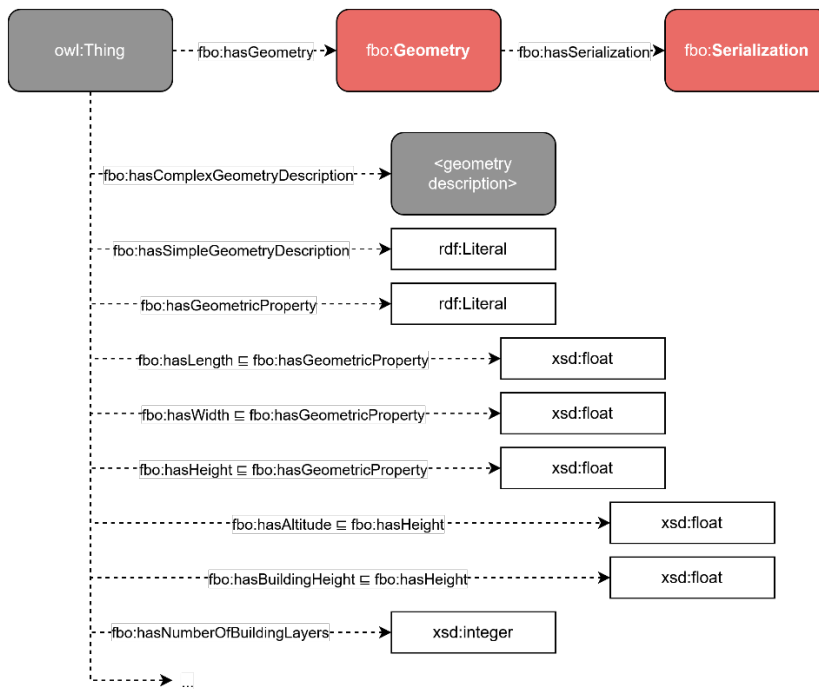


Figure 24 A snippet of the FBO-Geometry module

Any object in the graph can be linked to an `fbo:Geometry` node via the `fbo:hasGeometry` object property. This intermediate node can be defined by a specific serialization format using `fbo:Serialization`. External or highly detailed geometries, such as IFC files or topological graphs, can be linked using `fbo:hasComplexGeometryDescription`. Alternatively, more lightweight geometries can also be directly stored as a literal using `fbo:hasSimpleGeometryDescription`. This could for example be used to store Well-Known Text (WKT) strings, to support GeoSPARQL features.

Beyond explicit 3D representations, the module facilitates storing 1D or 2D parameters. The module introduces `fbo:hasGeometricProperty` as a top-level datatype property. To enable more granular querying, the property is extended with more specific datatype properties, such as `fbo:hasLength`, `fbo:hasHeight`, and `fbo:hasWidth`. These on their own can also have subproperties, to reach even more specific geometric properties (such as `fbo:hasBuildingHeight`). The national ontologies could use this hierarchical property creation to create very specific geometric properties that represent exactly the calculation method that is specified in the local regulations, while the hierarchy itself enables mapping those terms to other terms, potentially from other countries.

Additionally, discrete structural quantifications that impact fire safety, such as `fbo:hasNumberOfBuildingLayers`, are mapped separately to `xsd:integer`.

6.3.1.13. FBO-Zone

The FBO-Zone module is mostly a taxonomical tree with specific types of zones, which are modelled as subclasses of `fbo:Zone`. Figure 25 gives an overview of this module. It extends the core concept of a zone into distinct branches depending on how fire safety experts and modelers typically use them in practice.

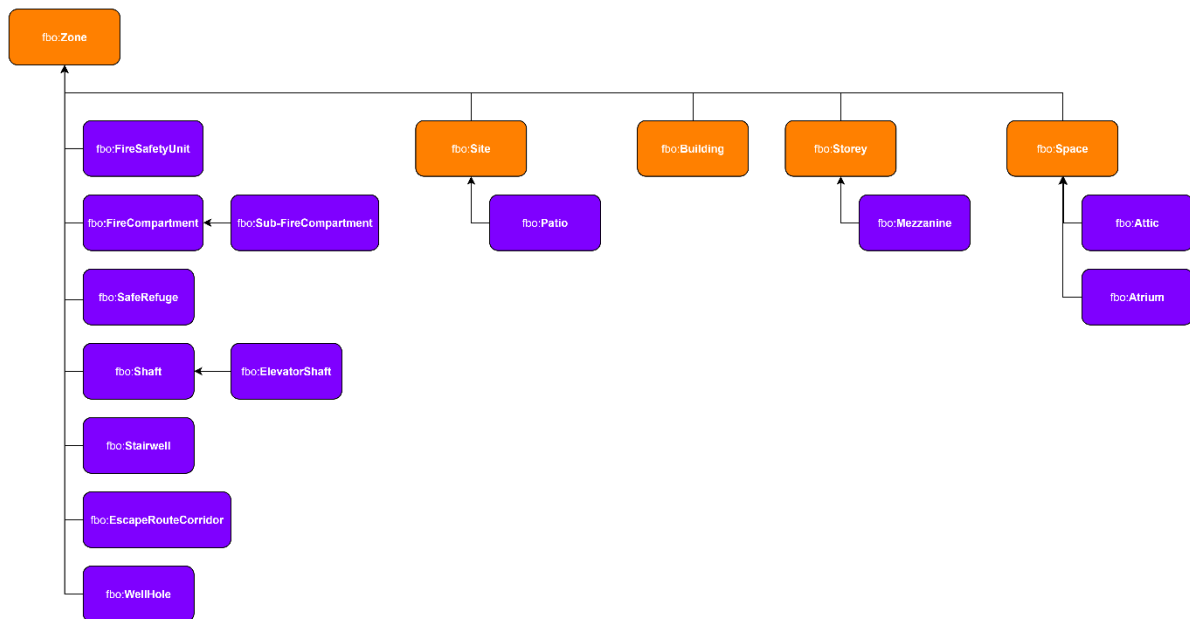


Figure 25 An overview of the FBO-Zone module

A significant part of the module consists of pure fire safety and evacuation zones, such as `fbo:FireSafetyUnit`, `fbo:FireCompartment`, `fbo:SafeRefuge`, and `fbo:EscapeRouteCorridor`. In an early design stage, these concepts are typically used as non-physical aggregations to group smaller spaces and assign regulatory constraints, like maximum fire load densities. The module also contains an `fbo:Sub-FireCompartment` class to enable modeling of nested compartments.

A second branch deals with vertical circulation and voids, which present unique risks for fire and smoke spread. This includes `fbo:Stairwell`, `fbo:WellHole`, and `fbo:Shaft` (with `fbo:ElevatorShaft` as a specific subclass).

The ontology finally extends the standard topological entities that were introduced in the FBO-Topology module. It must be noted that these extensions only contain classes that are specific topological types, e.g. an atrium is a specific type of space that has a dedicated topological representation. Extensions based on the function of these zones, such as ‘Kitchen’ or ‘Office’ are not part of this module, as functions are modeled in FBO-Function.

6.3.1. Vertical modularization

This European FBO ontology should be seen primarily as an umbrella ontology, under which other national ontologies could be subclassed. As this project aims for (legal) compliance checking, legal definitions need to be national (or even local) legal definitions that are closer to the national ontologies. It is therefore that the European FBO ontology covers terms that are semantically ‘relatively weak’, as most semantics are placed in the national extensions of FBO.

6.3.1.1. Terminological note on “harmonization”

When we speak of “harmonization” across the FBO framework (and generally in the FireBIM project), it is important to clarify what that means in the context of (legal) compliance checking. Harmonization in this context does not mean standardizing or overwriting local definitions of terms, with the goal of reaching European consensus on these definitions. Fire safety definitions are different across European countries. Sometimes, one could argue that certain standardization is possible, or even needed, for example, when using physical quantities. However, often times, definitions stem from

decades of developments in local fire safety knowledge, which is highly contextualized, making it impossible to enforce a single, universal definition across Europe. Even when considering physical quantities, this project taught us that standardization is challenging, as relatively simple concepts such as the height of a building are contextual (to architectural styles, to urban design, to firefighter practices, ...).

Instead, harmonization is achieved through structural alignment and interoperability of local terminology. The overarching FBO acts as a unifying umbrella that bridges terms from national fire safety terminology. Harmonization domain concepts in that case means mapping disparate local terms to shared, high-level umbrella concepts. This classification strategy is the core of our harmonization effort. By establishing which local terms map to which umbrella terms, fire safety experts dictate how terms are cross-referenced, which creates a framework that enables mapping and comparison of terms across borders. It allows understanding of similarities and differences of fire safety terms across borders, without losing local legal nuances.

6.3.2. FireBIM Matrix

An intensive knowledge acquisition phase led to the development of the FireBIM Matrix, which is a Notion board with a collection of relevant fire safety terms from all participating countries. The terms contain a rich set of metadata, such as their definitions, the source of these definitions, eventual calculation methods, links to other terms, links to legal documents, and so forth. The FireBIM Matrix is thoroughly described in Deliverable 4.1. Figure 26 shows a snippet of the collected terms in this matrix.

Asa TM - Harmonized Term	Country	TM - Name (English)	TM - Name (native language)	TM - Designators	TC - Adopted definition	TC - Explicit Definition (Native Language)	TC - Explicit Definition (English)	TC - Implicit Definition (Native Language)
AccessibleSector	Netherlands	accessible sector	toegankelijkheidssector	NA	Explicit Definition + Implicit Definition	voor personen met een functiebeperking zelfstandig bruikbaar en toegankelijk gedeelte van een gebouw	part of a building independently usable and accessible for persons with functional disabilities	Een woongebouw heeft een toegankelijkheidssector die hoger ligt dan 12 heeft van meer dan 15000' die voor zorg heeft een toegankelijk
AccessibleSectorElevatorLobby	Netherlands	elevator lobby accessible sector	voortooral lift toegankelijkheidssector	NA	Implicit Definition	NA	NA	Een extra beschermde vluchtroute mogen grenzen. Het voortooral kan ontstaan en kan komen bij
ActiveFireProtection	Europe	Active Fire Protection	Active Fire Protection	NA	Explicit definition	method(s) used to reduce or prevent the spread and effects of fire (3.2.38) heat or smoke (3.389) by virtue of detection and/or suppression of the fire and which require a certain amount of motion and/or response to be activated EXAMPLE: The application of agents (e.g. halon gas or water spray) to the fire or the control of ventilation and/or smoke. Note 2 to entry: Compare with the terms <i>passive fire protection</i> (3.320) and <i>suppression system</i> (3.418).	method(s) used to reduce or prevent the spread and effects of fire (3.2.38) heat or smoke (3.389) by virtue of detection and/or suppression of the fire and which require a certain amount of motion and/or response to be activated EXAMPLE: The application of agents (e.g. halon gas or water spray) to the fire or the control of ventilation and/or smoke. Note 2 to entry: Compare with the terms <i>passive fire protection</i> (3.320) and <i>suppression system</i> (3.418).	NA
ActiveFireProtection	Denmark	Active fire protection	Aktiv brandsikringsanlæg	NA	Implicit definition	NA	NA	Aktiv brandsikring omfatter alt om, og bekæmpe brand i en by
AdditionallyProtectedEscapeRoute	Netherlands	Additionally protected escape route	Extra beschermde vluchtroute	NA	Explicit Definition	Buiten een brandcompartiment gelegen gedeelte van een beschermde vluchtroute	Part of a protected escape route located outside a fire compartment	NA
AdjacentBuilding	Portugal	Adjacent Building	Edifício adjacente	NA	NA	NA	NA	NA
AdjacentRoom	Portugal	Adjacent Room	Espaço adjacente	NA	NA	NA	NA	NA
AdjacentStoreysArea	Portugal	Adjacent Storeys Area	Área do piso adjacente	NA	NA	NA	NA	NA
AdjoiningTerrain	Netherlands	adjoining terrain	aansluitend terrein	NA	Explicit Definition	aan een bouwwerk grenzend onbebouwd gedeelte van een bouwterrein of perceel of openbaar toegankelijk gebied;	empty terrain adjoining the plot or area available to the public	NA
AggregateRoomType	Portugal	Aggregate Room	Agregação de locais	NA	Expert definition	NA	NA	NA
AmmunitionExplosivesDepot	Portugal	Ammunition and Explosives Depot	Paiód de munições ou de explosivos	NA	NA	NA	NA	NA
Archive	Portugal	Archive	Arquivo	NA	NA	NA	NA	NA
AreaBuildingFunctionalFireUnit	Denmark	Functional Fire Unit Area	Bygningafnitsareal	NA	Implicit definition	NA	NA	Summen af bygningsektioner
AreaCompartmentCompliance	Portugal	AreaCompartmentCompliance	Conformidade da área de compartimentação	NA	Expert definition	NA	NA	NA
AreaCompartmentException	Portugal	Area Compartmentation Exception	Área de compartimentação - excepção	NA	Expert definition	NA	NA	NA
AreaDuplex	Belgium	Area of duplex compartment	Oppervlakte van een duplex	NA	Implicit definition	NA	NA	de gecumuleerde bruto-oppervlakte van de twee verdiepingen

Figure 26 A snippet of the collection of terms in the FireBIM Matrix

A variety of methods were tried to translate this matrix into an ontology, using both manual and (semi-)automatic procedures. The subsections below show an example of the applied methods in the Belgian context, which are illustrative for the other countries.

6.3.2.1. Manual construction from the matrix

The Belgian national ontology accompanying FBO 0.5 was constructed manually from the Belgian entries in the FireBIM Matrix in Notion. The entries were considered one at a time and placed in the FBO hierarchy. This involves deciding whether an entry describes a class, a property or a relationship, and identifying an appropriate parent in the European ontology. The national definitions and their source references provide the basis for these decisions.

The structure of the matrix does not always make these modelling choices explicit. Some entries combine a definition with a condition from a regulation, while others need further clarification before they can be represented consistently in OWL. Manual construction therefore remained necessary alongside the automated work described below.

6.3.2.2. Semi-automated generation from Notion

Buildwise developed `NotionToOntology.py` to automate the transfer of the matrix content into RDF. The script reads the Notion databases containing the terminology and supporting information, including countries, source documents, relations and IFC mappings. It resolves the links between records and generates both a combined Turtle graph and separate files for the European vocabulary, the national vocabularies and the IFC mappings. The Belgian vocabulary is written to `fbo-be.ttl`. The same process produces the Danish, Lithuanian, Dutch and Portuguese files.

The conversion follows explicit rules based on the fields in the matrix. Country information determines the namespace of a term. The technical type field distinguishes classes from properties, while value-type and unit information help distinguish datatype properties from object properties. Parent relationships can be translated into `rdfs:subClassOf` or `rdfs:subPropertyOf` statements. Other links in the matrix are retained through properties such as `fbo:isLinkedTo` and `fbo:hasSubitem`.

Where available, the generated records include English and national-language labels and definitions, references to source documents, and links back to the corresponding Notion entries. The script also copies the last-edit date and the fire-expert validation field. This allows the source entry and its recorded status to be consulted when a term needs clarification. IFC mappings are exported separately to `fbo-ifc.ttl`, where mapping records can describe the IFC class, predefined type or property-set information associated with a term.

This removes much of the repeated work of copying records and recreating their links. The overall process is nevertheless semi-automated, because the quality of the generated ontology depends on the content and structure of the matrix. The script cannot resolve an ambiguous definition or determine whether a modelling choice is appropriate for the intended check. Missing type information can also cause an entry to be treated as a class by default. The generated vocabulary therefore still needs review. Corrections are made in Notion or in the conversion logic, after which the files can be regenerated.

6.3.3. A method for future extendibility and alignment of the national ontologies

The evolving nature of the matrix, and therefore the need for an ontology that is easily updatable, led to the decision to use semi-automated approaches for the current versions of the national ontologies. The Notion API was used to export the table, and a Python script then serialized the table into a graph-based format. Since the table does not indicate whether something is a class or a property, an LLM does a first prediction of these types and adds the prediction to the graph. The result of this procedure is an intermediate ontology for each country.

A second LLM agent (using Gemini 3.1 Pro) is then given a multi-stage prompt. It is requested to:

1. Load the manually created FBO ontology, including all the models, and read it.
2. Load the intermediate ontology of a country, that was generated using the FireBIM Matrix.
3. Based on the terms in the manual FBO ontology, correct the `owl:Class`, `owl:ObjectProperty` and `owl:DatatypeProperty` triples.
4. Map all the terms from the intermediate ontology of a country to the FBO ontology of that country, using either `rdfs:subClassOf`/`rdfs:subPropertyOf` or an equivalence statement.

The LLM agent presented intermediate results after each step, enabling humans to correct intermediate results. The method enables quick mapping between the manual input of the fire safety experts, and the manually created FBO ontology that serves as a stable structure. By doing this, we enable fire safety experts in the future to enrich the ontology, by simply adding new rows to the Notion board (or a similar tabular representation).

6.3.4. National ontologies

The screenshot shows the FireBIM website's 'National modules' section. It features a grid of five cards, each representing a national fire-safety ontology (FBO) for a different country. Each card includes the country name, a brief description, version information, and download links.

Country	URI	Description	Versions	Classes	Properties	Individuals	File Size	Download Link
Belgium (FBO-BE)	https://firebim.org/ontologies/fbo/be#	Belgian fire-safety vocabulary with legal definitions, regulation references and links to the core FBO concepts and to IFC.	2026.09.04 (DRAFT, LATEST) issued 2026-09-04	21	42	9	59 kB	.ttl browse
Denmark (FBO-DK)	https://firebim.org/ontologies/fbo/dk#	Danish fire-safety vocabulary (BR18 and guidelines) with Danish and English labels and definitions, regulation references and links to the core FBO concepts and to IFC.	2026.09.04 (DRAFT, LATEST) issued 2026-09-04	17	90	21	138 kB	.ttl browse
Lithuania (FBO-LT)	https://firebim.org/ontologies/fbo/lt#	Lithuanian fire-safety vocabulary with legal definitions, regulation references and links to the core FBO concepts and to IFC.	2026.09.04 (DRAFT, LATEST) issued 2026-09-04	31	31	9	84 kB	.ttl browse
Netherlands (FBO-NL)	https://firebim.org/ontologies/fbo/nl#	Dutch fire-safety vocabulary (Bbl) with legal definitions, regulation references and links to the core FBO concepts and to IFC.	2026.09.04 (DRAFT, LATEST) issued 2026-09-04	29	80	5	119 kB	.ttl browse
Portugal (FBO-PT)	https://firebim.org/ontologies/fbo/pt#	Portuguese fire-safety vocabulary with legal definitions, regulation references and links to the core FBO concepts and to IFC.	2026.09.04 (DRAFT, LATEST) issued 2026-09-04	44	93	7	178 kB	.ttl browse

Figure 27 An overview of the national FBO modules

The national FBO modules (Figure 27) are published on the FireBIM website, and are version controlled. Older (draft) versions are kept alive, so that software that is built on top of the older versions will not break.

The metadata of the national ontologies clearly shows differences in the size of these ontologies, some being larger than others, while some ontologies also have a larger share of properties. This stems from the differences in how national fire regulations are modelled, as these national ontologies directly represent the terms that are being used in these fire regulations. The

differences in modelling can be found in both the differences in the fire safety regulations themselves, as well as in how these regulations are expressed in a specific language.

6.3.4.1. Application of the national ontologies

The Belgian compartmentation demonstrator described in D5.2 uses FBO terms to represent the information required by a selected check. In the IFC-based workflow, configurable mappings tell the parser where to find the relevant objects and properties in the dataset. Several ways of representing compartments were tested, including groups of IfcSpace objects, IfcZone, IfcSpatialZone and classified proxy objects. The parser uses the selected mappings to create a building graph containing concepts such as fbo-be:Building and fbo-be:FireCompartment, together with the properties required by the digital rules. (D5.2, Section 3.1.1, pp. 8-10.)

The ontology defines the vocabulary used in this graph; the parser creates instances that describe a particular building. Some properties can be read from the model, while others need to be calculated. For a compartment assembled from rooms, for example, the treatment of internal walls and compartment boundaries affects the calculation of its area or volume. These calculations are part of the application that produces the building graph.

D-studio's Revit implementation produces the graph directly from the authoring environment. The user groups rooms into compartments and selects the applicable checks. The plug-in then creates the FBO instances, adds the required properties and calculated geometry, and submits the graph to the backend. A compartment can consequently be represented in FBO even when there is no separate compartment object in the Revit model. (D5.2, Section 3.1.1, pp. 11-14.)

This application work also led to additional terms for information needed during processing. The RDF examples in D5.2 use the fbo-core-program: namespace for solid geometry, references to the elements making up a compartment, and an indication of which tool calculated a value. Examples include SolidGeometry, CompositionReference and isCalculatedBy. These terms describe how the building graph is constructed and how derived information is recorded. (D5.2, pp. 12-13.)

Application extensions can therefore identify information that needs a place in the shared ontology. A reusable geometry concept may fit within FBO-Geometry, whereas a term describing a software operation may remain in a separate application module. National definitions remain in FBO-BE. Incorporating an extension requires checking its meaning and its relation to the existing classes and properties; the terms used in the demonstrator should not be assumed to have already been included in FBO 0.5.

6.4. Answering the competency questions

In the subsections below, we'll answer the competency questions that the ontology aims to enable answering. The subsections take the Danish and the Dutch ontology as an example. Since all ontologies reuse the European FBO structure, and are modeled similarly, one could simply replace the national prefixes for the national prefixes of other countries, and the queries would still be valid.

6.4.1. What is the definition of a fire safety term in a national regulation?

All national fire safety terms have a definition that is described in the ontology, that can be queried using regular `rdfs:comment` (and `rdfs:label` for the name) properties. Listing 16 shows the SPARQL query that returns definitions of the Danish term `fbo-dk:FireCompartment`, and Table 9 shows the results. These are given in two languages, as the concept has definitions in both Danish and English.

```
PREFIX fbo-dk: <http://www.firebim.org/ontologies/fbo-dk#>
```

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?definition
WHERE {
    fbo-dk:FireCompartment rdfs:comment ?definition
}

```

Listing 15 SPARQL query answering CQ1

Table 9 Results of CQ1

definition
"A detached (in terms of fire spread) building or a defined area of a building that is constructed with fire compartment separations from other rooms and buildings. A fire compartment consists of one or more Subcompartments."@en
"En brandmæssigt fritliggende bygning eller et afgrænset område af en bygning, som er udført med brandsektionsadskillelser mod andre rum og bygninger. En brandsektion består af en eller flere brandceller."@da

FBO models definitions specifically as `fbo:hasExplicitDefinition`, `fbo:hasImplicitDefinition`, and `fbo:hasExpertDefinition`, as some terms are not explicitly defined in the regulatory documents. These three properties are all subproperties of `rdfs:comment`. Would one want to query more specific definitions, e.g. only those that are explicit, then these properties could be used. A filter can be applied to select definitions in a specific language only. Finally, as all national terms are modelled as subclasses and subproperties of the European FBO, one could also use European terms to find a term from a specific country. Listing 17 shows how these ideas are materialized in a SPARQL query. First, a national term is queried by using the European term `fbo:hasFireCompartmentArea`. Then, a triple is created to find the definition using a variable `?definitionProperty`, where `?definitionProperty` is one of the three previously mentioned properties. Finally, this definition is filtered to be in English. Table 10 shows the results of this query. It shows two properties, namely the area of the fire compartment as well as the area of the subcompartment in Denmark, and for each of these properties an explicit definition and an implicit definition was found.

```

PREFIX fbo: <http://www.firebim.org/ontologies/fbo#>
PREFIX fbo-dk: <http://www.firebim.org/ontologies/fbo-dk#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?nationalTerm ?definitionProperty ?definition WHERE {
    ?nationalTerm rdfs:subPropertyOf* fbo:hasFireCompartmentArea.

    ?nationalTerm ?definitionProperty ?definition .

    FILTER(?definitionProperty IN (
        fbo:hasExplicitDefinition,
        fbo:hasImplicitDefinition,
        fbo:hasExpertDefinition
    ))

    FILTER (LANG(?definition) = "en")
}

```

Listing 16 SPARQL query answering CQ1 with more detail

Table 10 Results of CQ1 with more detail

nationalTerm	fbo:hasExplicitDefinition	fbo:hasImplicitDefinition
fbo- dk:AreaFireCompartment	“The sum of the gross areas of all floors, including basements and usable roof spaces as well as enclosed balconies, enclosed porches, connecting corridors and similar. In connection with fire conditions, no deductions may be made, cf. BR18 § 455(3)(1), (4) and (5) and (4), which may only be used when calculating the building percentage.”@en	“Extract from BR18 §445: The total area of the fire compartment is calculated as follows: - by adding the gross areas of all floors, including inserted floors, basements and usable attics as well as balcony openings, conservatories, connecting corridors and similar. - Rooms that pass through several floors are only included in the floor in which the floor is located. - However, both external and internal stairs, stairwells, balconies, installation shafts and elevator shafts are counted for each floor. The floor area is not included: - Open balconies (completely open above the railing on at least 2 sides) - Open roof terraces. - Bed lofts with an area of up to 4.5 m². - Projection of external stairs and balconies on the ground. - External fire escape stairs (escape stairs). - External insulation of a maximum of 25 cm. The floor area is measured in a plane determined by the top of the finished floor to the outside of the limiting external walls with the following deviations: 1) In usable attics, the area in a horizontal plane 1.50 m above the finished floor is included within the intersection of the plane with the outside of the roof covering. 2) For open floors, gates, airlocks and the like, the area is counted to the line indicated by the outer walls of the building. 3) For common walls between rooms that are to be included in each floor area, measure to the middle of the wall; however, for common walls

		located above the boundary of the property line, measure to the boundary property line.”@en
fbo-dk:AreaSubCompartment	“The floor area of a room's floor surface delimited by the room's surrounding walls. When calculating the area, no deduction is made for the area of any columns and other building elements in the room.”@en	“The area of a Subcompartment is the sum of the floor areas of the rooms.”@en

6.4.2. How is a quantity calculated in a national regulation?

Calculation methods of quantities, for example the Dutch term fbo-nl:UsableAreaFireCompartment, are stored in the definition of the term. They can therefore be accessed similarly to the query in CQ1. Listing 18 shows the query, and Table 11 shows the calculation method for this quantity.

```
PREFIX fbo: <http://www.firebim.org/ontologies/fbo#>
PREFIX fbo-nl: <http://www.firebim.org/ontologies/fbo-nl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?definition WHERE {
  fbo-nl:UsableAreaFireCompartment rdfs:comment ?definition

  FILTER (LANG(?definition) = "en")
}
```

Listing 17 SPARQL query answering CQ2

Table 11 Results of CQ2

definition
“The usable area of a firecompartment is measured at floor level, between the rising partitions that enclose the room or group of rooms in question. The following are not taken into account when determining the GO: — the area of parts of floors above which the net height is less than 1.5 m, with the exception of floors under stairs, ramps, etc.; — a lift shaft; — a stairwell, ring hole or void, if its area is greater than or equal to 4 m ² ; — a free-standing building structure (other than a staircase) if its horizontal cross-section is greater than or equal to 0.5 m ² ; — a service shaft, if its horizontal cross section is greater than or equal to 0.5 m ² ; — a load-bearing internal wall.”@en

6.4.3. What is the preferred IFC4.3 method to model a fire safety term in a national regulation?

We use a custom object property to link classes and properties to a mapping in IFC4.3. This will be further elaborated in Section 6.5 Alignment. Listing 19 shows a SPARQL query using that object property to query the IFC mapping to a specific property, and Table 12 shows the results of that query.

```
PREFIX fbo: <http://www.firebim.org/ontologies/fbo#>
PREFIX fbo-dk: <http://www.firebim.org/ontologies/fbo-dk#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
SELECT ?ifcMapping WHERE {
  fbo-dk:BuildingLayers fbo:hasIFC4.3Equivalent ?ifcMapping .
}
```

Listing 18 SPARQL query answering CQ3

Table 12 Results of CQ3

definition
<https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/NumberOfStoreys>

6.4.4. What is the ISO equivalent of a term in a national regulation?

Similar to mapping a term to IFC, we created an object property to map a term to a term in an ISO standard. There is no specific standard that the ontology tries to target, however, some obvious ones that were used are ISO 13943:2023(en) Fire Safety, and ISO 6707-1:2020(en) Buildings and civil engineering works. The SPARQL query in Listing 20 shows how to return the ISO term, and Table 13 shows the result for a specific property, fbo-dk:FireResistance.

```
PREFIX fbo: <http://www.firebim.org/ontologies/fbo#>
PREFIX fbo-dk: <http://www.firebim.org/ontologies/fbo-dk#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?isoEquivalent
WHERE {
  fbo-dk:FireResistance fbo:hasISOEquivalent ?isoEquivalent .
}
```

Listing 19 SPARQL query answering CQ4

Table 13 Results of CQ4

definition
<https://www.iso.org/obp/ui/#iso:std:iso:13943:ed-4:v1:en:term:3.165>

6.4.5. Does country A have terms that are similar to term X in country B?

Since the national ontologies are mapped to the European FBO ontology, this European ontology serves as an umbrella ontology, and can therefore be used to map terms between countries. Imagine the scenario where a Dutch architect is designing a building in Denmark, and wants to understand whether Denmark has a term that is similar to what in the Netherlands is known as ‘brandcompartiment’ (‘fire compartment’). Listing 21 shows how this is modeled in SPARQL. First, we take the Dutch fbo-nl:FireCompartment, and find the ?europeanTerm to which it maps. Then, we find the other subclasses of that term, specifically in the Danish namespace. We use rdfs:subClassOf* with an asterix, so that the subclasses of subclasses are also returned. Finally, we filter for English and Danish labels. Table 14 shows the results. It shows that in Denmark, there are two terms related to the Dutch term ‘brandcompartiment’, namely the Danish ‘brandsektion’ and ‘brandcelle’. A definition is also given.

```
PREFIX fbo: <http://www.firebim.org/ontologies/fbo#>
PREFIX fbo-nl: <http://www.firebim.org/ontologies/fbo-nl#>
PREFIX fbo-dk: <http://www.firebim.org/ontologies/fbo#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?europeanTerm ?danishTerm ?danishDefinition WHERE {
  # Trace the Dutch term to its European core concept
```

```

fbo-nl:FireCompartment rdfs:subClassOf ?europeanTerm .

# Find any Danish terms tied to that same core concept
?danishTerm rdfs:subClassOf* ?europeanTerm ;
    rdfs:label ?danishLabel ;
    rdfs:comment ?danishDefinition .

# Ensure the core concept belongs to the European base ontology
FILTER (STRSTARTS(STR(?europeanTerm),
"http://www.firebim.org/ontologies/fbo#"))

# Ensure the Danish concept belongs to the Danish ontology
FILTER (STRSTARTS(STR(?danishTerm),
"http://www.firebim.org/ontologies/fbo-dk#"))

    FILTER (LANG(?danishLabel) = "da"
    FILTER (LANG(?danishDefinition) = "en"

}

```

Listing 20 SPARQL query answering CQ5

Table 14 Results of CQ5

europeanTerm	danishTerm	danishLabel	danishDefinition
fbo:FireCompartment	fbo-dk:FireCompartment	“Brandsektion”@da	"A detached (in terms of fire spread) building or a defined area of a building that is constructed with fire compartment separations from other rooms and buildings. A fire compartment consists of one or more Subcompartments." @en ;
fbo:FireCompartment	fbo-dk:FireCompartmentSubcompartment	“Brandcelle”@da	“One or more rooms from which the fire does not spread to other subcompartments during the time required for evacuation and rescue efforts in adjacent subcompartments.” @en

6.4.6. What is the unit of a fire safety property?

The ontology can also be used to declare using which unit a specific national property should be modeled. Our own `fbo:Unit` with a set of subclasses is used for this, which is mapped to QUDT. Listing 22 shows how to query the unit of such a property, and Table 15 shows the unit in which `fbo-dk:FireLoadDensity` should be modeled.

```
PREFIX fbo-dk: <http://www.firebim.org/ontologies/fbo#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?property ?propLabel ?unit ?unitLabel WHERE {
  fbo-dk:FireLoadDensity fbo:hasApplicableUnit ?unit .
  ?unit rdfs:label ?unitLabel .
}
```

Listing 21 SPARQL query answering CQ6

Table 15 Results of CQ6

unit	unitLabel
fbo:MegaJ-PER-M2	"MegaJ-PER-M2"@en

6.5. Alignment

6.5.1. Terminological note on “reuse”

There is general consensus amongst semantic web experts, and also in other communities that deal with the development of schemas, that reusing ontologies is a best practice. Not entirely clear is what reusing means in practice. Multiple ideas exist on the reuse of ontologies (see Moreira et al., 2024), and in this project, we do not directly apply classes from other ontologies, but mirror them and place them in the FBO namespace. Alignment modules then take care of a mapping to the original source, often by using `rdfs:subClassOf` or `rdfs:subPropertyOf` properties. Similar approaches were used by earlier ontology development projects in the AECO industry, including SAREF4BLDG, that has an alignment module to BOT (Schneider et al., 2017). The core reason to apply this specific method of reuse is threefold:

1. All terms in this project are available using the same prefix, making it easier to write SPARQL queries or to build software.
2. Software that is built using those ontologies is dependent on FBO, and not on a range of ontologies maintained by external organizations.
3. It enables the project to enrich these external concepts, such as those borrowed from QUDT, with custom domain-specific metadata (such as national identification codes, multilingual labels, etc.) under local governance, while avoiding the overhead of importing entire external schemas.

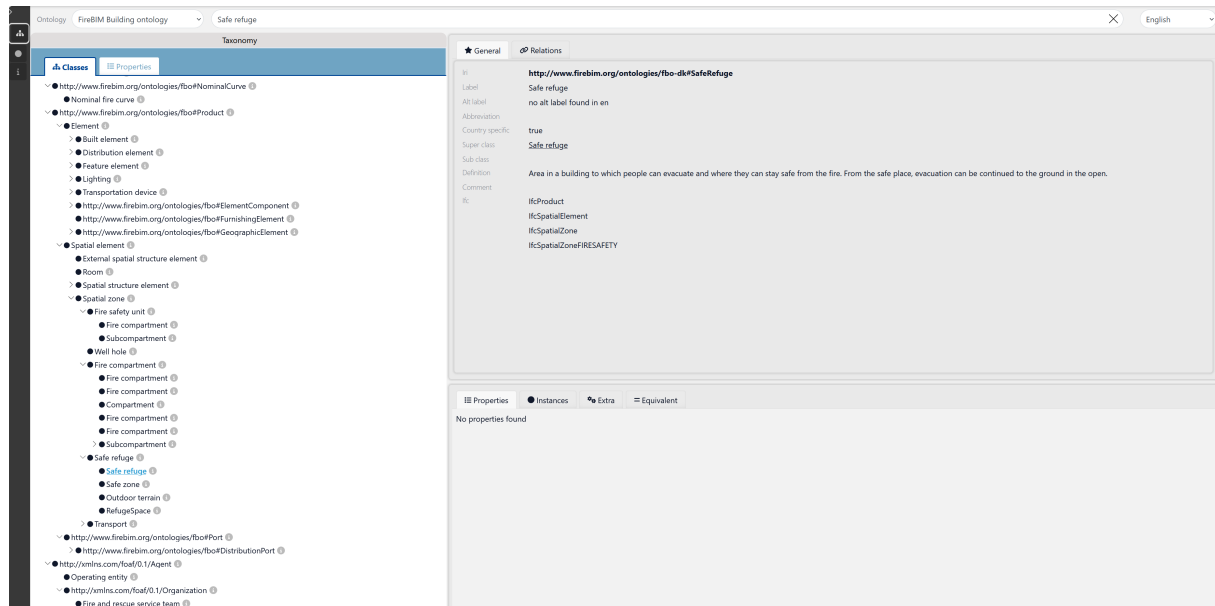
6.5.2. Alignment to FOAF and QUDT

FBO is aligned to various other ontologies, being FOAF, QUDT, and via a softer alignment to IFC. To make the alignment to FOAF and QUDT, alignment modules were created. The FOAF alignment aligns an `fbo:Person` to a `foaf:Person`, using a subclass relationship, and the QUDT alignment aligns all classes in the FBO-QuantityKind module to classes in QUDT, if available.

6.6. User interfaces

While this section is not explicitly part of this deliverable, during the years, FireBIM has produced a variety of tools that lower the barrier to using and maintaining the ontologies.

First, below is the Wistor platform, which has the ontologies loaded and enables searching terms from these ontologies.



Ideas from the Wistor platform were later used to create an overview of the ontologies on the firebim website, which similarly enables end users to scroll through the terms in the ontologies and visualize the information in a human-readable user interface.

FBO-DK – Denmark

Danish fire-safety vocabulary (BR18 and guidelines) with Danish and English labels and definitions, regulation references and links to the core FBO concepts and to IFC.

fbo-dk: <<https://firebim.org/ontologies/fbo/dk#>>

Version IRI: <https://firebim.org/ontologies/fbo/dk/2026-09-04#>

17 classes · 90 properties · 21 individuals · 2120 triples

Version 2026.09.04 · draft · latest

DRAFT issued 2026-09-04 [fbo-dk.ttl](#)

compartment

AllClassesObject propertiesDatatype propertiesIndividuals

28 terms

AreaFireCompartmentCompartment-Area - Floor area

AreaSubCompartmentSubcompartment area

AutomaticFireVentilationsystemAutomatic Fire Ventilation...

BoundaryFireCompartmentFire compartment - boundary

BoundarySubCompartmentRoom - Subcompartment - bo...

BuildingFireCompartmentDaytimeoccupancyDaytime ...

BuildingFunctionalFireUnitFunctional Fire Unit

CompartmentSleepingSleeping section

DoorExitExit door

DoorIndependentExitdoorsIndependent exit doors

ExitExit

FireCompartmentFire Compartment

FireCompartmentSubcompartmentCompartment - Sub...

FireHazardSpecialSpecial fire hazard

FireSafetyUnitFire safety unit

FreeOpenRampsBuilding compartments with free open ra...

FunctionTypeFunction Type of use

IndustrialClass5Industrial and warehouse class 5

NumberOccupantsSpaceCompartmentNumber of Occ...

NumberOccupantsSubcompartmentsNumber of Occup...

RoomfunctionEscaperouteCorridorEvacuation route corr...

RoomfunctionEscapestaircaseEmergency escape stairc...

SeparationFireFire separation

SystemIndependentExitIndependent exits

WalltypeCompartmentFirewallCompartment Firewall

WalltypeSubcompartmentFirewallSubcompartment Fire...

WidthFreeDoorsFree width Emergency escape door

WidthFreeEvacuationroutesFree width Evacuation routes

ClassObject propertyDatatype propertyIndividual

CLASS

Fire Compartment

fbo-dk:FireCompartment

<https://firebim.org/ontologies/fbo/dk#FireCompartment>

LABELS

BrandsektionDA-DK

Fire CompartmentEN

DEFINITION

En brandmæssigt fritliggende bygning eller et afgrænset område af en bygning, som er udført med brandsektionsadskillelser mod andre rum og bygninger. En brandsektion består af en eller flere brandceller.

DA-DK

A detached (in terms of fire spread) building or a defined area of a building that is constructed with fire compartment separations from other rooms and buildings. A fire compartment consists of one or more Subcompartments.

EN

SUBCLASS OF

fbo:FireCompartment - Compartment (fbo/core)

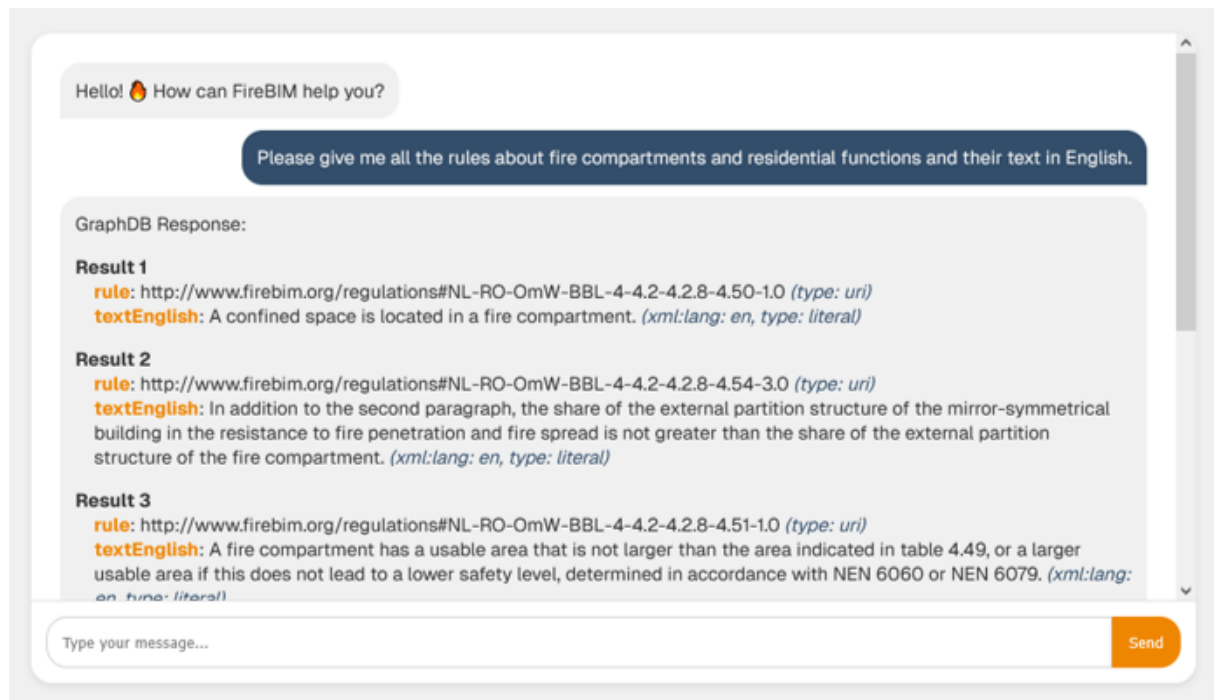
LINKED TO (UMBRELLA CONCEPT)

fbo:FireCompartment - Compartment (fbo/core)

ANNOTATIONS & OTHER PROPERTIES

fbo:hasAdoptedDefinitionType	Explicit definition
fbo:hasArticleID	BR18-VK5-D01
fbo:hasCountry	fbo:Denmark - Denmark (fbo/core)
fbo:hasCountryCode	DK
fbo:hasDocumentReference	fbo-dk:BR18VK5D01 - BR18-VK5-D01
fbo:hasDomainReference	fbo:X11CompartmentSize - 1.1 Compartment Size (fbo/core) fbo:X12FireCompartmentBoundary - 1.2 Fire Compartment Boundary (fbo/core) fbo:X1COMPARTMENTATION - 1. COMPARTIMENTATION (fbo/core)
fbo:hasEquivalentTerm	Fire section, Fire division
fbo:hasFireExpertValidation	true**xsd:boolean
fbo:hasLastEdition	2026-04-22T11:48:00+00:00**xsd:dateTime
fbo:hasNotionID	130
fbo:hasNotionURL	https://www.notion.so/FireCompartment-151f8487e6e78099a790f6834c5080d
qudt:hasApplicableUnit	[-]

Finally, a chatbot interface was built. The FireIBM Chatbot is an interactive, web-based tool that bridges the gap between the ontologies, which are often complex to understand, query, or navigate for non-semantic web experts, and end-user accessibility. By integrating domain ontologies with a GraphRAG (Graph Retrieval-Augmented Generation) architecture, the tool enables stakeholders to query both building codes and terms from the ontology using natural language. The system translates natural language queries into SPARQL queries, allowing non-expert users to query the ontology without manual SPARQL creation. The chatbot interacts with the multiple ontologies, enabling cross-country querying that is also answered by competency question 5. Unlike most LLM-based chatbots, the LLM behind this chatbot solely generates the SPARQL queries. The result that it posts in the chatbot interface is then the response of a graph database. The generated SPARQL query that was used to return these results can also be seen, enabling domain experts to verify the query logic.



7. FireBIM Regulation Quality Ontology

A final artifact in the FireBIM Ontology Stack is the FireBIM Regulation Quality Ontology, or FRQ. This small ontology is designed to enable human validation of the digital rules that will be created in this project. The hypothesis is that for different countries, and even within a country but for different types or complexities of rules, different methods will be applied to create these digital rules. Some of those will be created by humans, some by deterministic algorithms, and some by (at least partially) using artificial intelligence. To enable end-users to trust the library of digital rules, this FRQ ontology enables storing human expert evaluations with those digital rulesets.

Figure 28 shows an overview of the ontology, which consists of only one class and a set of properties. An instance of `frq:DocumentSubdivision` connects to an evaluation record via the `frq:hasEvaluation` property, which instantiates an `frq:Evaluation` node. This node holds a set of evaluation metrics, in a few categories. The `frq:hasFireExpertValidation` indicates whether a certified fire safety expert has reviewed and approved the digital interpretation of the rule, while `frq:hasArgumentation` captures qualitative expert commentary, edge cases, or legal justifications behind that evaluation. Then, some properties evaluate the technical and semantic correctness of the digital rule. The `frq:hasSyntaxCorrectness` evaluates whether the digital rule does not have syntactical errors, `frq:hasTermMappingCorrectness` evaluates whether the terms in the natural language representation of the rule were correctly mapped to one of the terms in the FBO ontology, and `frq:hasManualCorrectionEffort` evaluates the effort made by human experts to fix those errors. Then, the ontology contains properties to evaluate the different formats of the rule, with `frq:hasHumanReadableFormat` and `frq:hasMachineReadableFormat` evaluating whether these formats exist, and `frq:hasHumanReadableFormatGenerationMethod` and `frq:hasMachineReadableGenerationMethod` documenting the pipelines used to generate those formats. Finally, `frq:hasFireBIMEvaluation` serves as an aggregated evaluation metric or status indicator derived from the other validations, and could serve as a final verdict that states how reliable we as a consortium think the digital rule is.

Extending this ontology with more properties is relatively easy, and future extensions could be made based on more rigid evaluation criteria. In order for these to materialize, we propose that the end results of FireBIM should be used for a period of time, after which a group of fire safety experts and semantic web experts could evaluate whether this set of evaluation criteria is sufficient, or whether the ontology needs an extension.

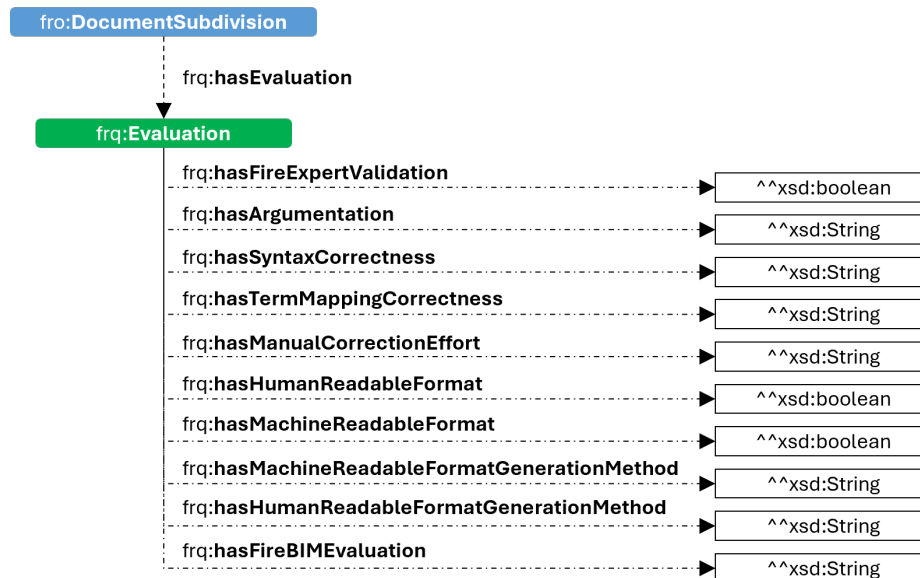


Figure 28 The FireBIM Regulation Quality Ontology

While the **frq:hasFireBIMEvaluation** could be manually filled, this is also something that could be automated once a more rigid evaluation metric is in place. Listing 23 shows an example SPARQL Construct query that generates a **frq:hasFireBIMEvaluation** score based on three other criteria, namely whether the rule has been validated by a fire expert, whether the syntax is correct, and whether the term mapping is correct. If all three are true, the query generates a “Validated” literal, whereas otherwise a “Requires Expert Review” is generated.

```

PREFIX frq: <http://www.firebim.org/ontologies/frq#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

CONSTRUCT {
    ?eval frq:hasFireBIMEvaluation ?score .
}
WHERE {
    ?eval a frq:Evaluation ;
        frq:hasFireExpertValidation ?expertVal ;
        frq:hasSyntaxCorrectness ?syntax ;
        frq:hasTermMappingCorrectness ?mapping .

    BIND(
        IF(?expertVal = true && ?syntax = "Correct" && ?mapping = "Correct",
            "Validated",
            "Requires Expert Review")
        AS ?score
    )
}
  
```

Listing 22 A SPARQL Construct query that generates a **frq:hasFireBIMEvaluation** based on other evaluation metrics

8. Conclusion

This deliverable presented the FireBIM Ontology Stack, consisting of three interconnected ontologies: the FireBIM Regulation Ontology (FRO), the FireBIM Building Ontology (FBO), and the FireBIM Regulation Quality Ontology (FRQ). Together, these ontologies provide a semantic bridge between natural language fire safety regulations and digital building information models. By employing a modular architecture, the ontology stack harmonizes high-level fire safety concepts across Europe while maintaining the highly specific legal definitions required for national compliance checking. The applied methodology ensures that the ontology stack can be extended in the future to include additional countries and new (fire safety-related) knowledge domains.

8.1. Handover

The primary purpose of the FireBIM Ontology Stack is to serve as a central terminology library for the subsequent work packages in the FireBIM project. The transition of this work to other parts of the project takes place in several directions:

8.1.1. Work Package 2 (Digitization of regulations)

The structural components defined in FRO and the domain terminology in FBO are direct inputs for Deliverable 2.3 (Generic framework for digitization of fire safety regulations). They provide the necessary semantic structure to convert natural language building codes into machine-readable formats, which will eventually culminate in the digital rulesets produced in Deliverable 2.5.

8.1.2. Work Package 3 (Reference architecture)

The ontology stack provides the core semantic data model for the FireBIM reference architecture. It will be implemented to structure both rules and linked building data in the FireBIM platform, enabling data federation, querying, reasoning, and compliance checking across different building models and regulations.

8.1.3. Work Package 4 and 5 (Applications and demonstrators)

The national modules of FBO are a prerequisite for executing the demonstrators in WP5. The demonstrators will utilize the ontology to map building information (such as IFC models) to the specific national terms required to perform automated, SHACL-based rule checking. Furthermore, feedback derived from these demonstrators and the end-user requirements in WP4 could be used to iteratively update the ontology stack.

8.1.4. Work Package 6 (Standardization)

The content of the ontologies, and the alignments modeled for the FBO ontology, directly feed into the work on standardisation and certification of WP6. The methodology of linking national legal terminology to a harmonized European umbrella ontology serves as a reference case for future cross-border standardization efforts in the AEC domain.

9. References

1. Donkers, A., Yang, D., de Vries, B., & Baken, N. (2022). Semantic web technologies for indoor environmental quality: A review and ontology design. *Buildings*, 12(10), 1522.

2. Guyo, E. D., Hartmann, T., & Snyders, S. (2023). An ontology to represent firefighters data requirements during building fire emergencies. *Advanced Engineering Informatics*, 56, 101992.
3. Wang, R. G., Wu, P. Y., Liu, C. Y., Tan, J. C., Chuang, M. L., & Chou, C. C. (2022). Route planning for fire rescue operations in long-term care facilities using ontology and building information models. *Buildings*, 12(7), 1060.
4. Deng, H., Fan, Z., Wei, X., & Deng, Y. (2025). An ontology-based approach to dynamic indoor fire emergency evacuation path planning with BIM integration. *Journal of Building Engineering*, 106, 112562.
5. Da, M., Zhong, T., & Huang, J. (2023). Knowledge graph construction to facilitate indoor fire emergency evacuation. *ISPRS International Journal of Geo-Information*, 12(10), 403.
6. Neto, J., Morais, A. J., Gonçalves, R., & Coelho, A. L. (2022). Context-based multi-agent recommender system, supported on IoT, for guiding the occupants of a building in case of a fire. *Electronics*, 11(21), 3466.
7. Huang, C. Y., Chiang, Y. H., & Tsai, F. (2022). An ontology integrating the open standards of city models and Internet of things for smart-city applications. *IEEE Internet of Things Journal*, 9(20), 20444-20457.
8. Chou, C. C., Jeng, A. P., Chu, C. P., Chang, C. H., & Wang, R. G. (2018). Generation and visualization of earthquake drill scripts for first responders using ontology and serious game platforms. *Advanced Engineering Informatics*, 38, 538-554.
9. Fitkau, I., & Hartmann, T. (2024). An ontology-based approach of automatic compliance checking for structural fire safety requirements. *Advanced engineering informatics*, 59, 102314.
10. Chen, Y., & Jiang, H. (2024). Optimizing automated compliance checking with ontology-enhanced natural language processing: Case in the fire safety domain. *Journal of Environmental Management*, 371, 123320.
11. Li, S., Wang, J., & Xu, Z. (2025). Automated compliance checking for BIM models based on Chinese-NLP and knowledge graph: an integrative conceptual framework. *Engineering, Construction and Architectural Management*, 32(6), 3832-3856.
12. Hassan, F. U., & Le, T. (2023). Ontology-based approach to risk-factor identification to support the management of provisions in bridge design. *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction*, 15(1), 04522031.
13. Beetz, J., Van Leeuwen, J., & De Vries, B. (2009). IfcOWL: A case of transforming EXPRESS schemas into ontologies. *Ai Edam*, 23(1), 89-101.
14. Pauwels, P., & Terkaj, W. (2016). EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology. *Automation in construction*, 63, 100-133.
15. Rasmussen, M. H., Lefrançois, M., Schneider, G. F., & Pauwels, P. (2020). BOT: The building topology ontology of the W3C linked building data group. *Semantic Web*, 12(1), 143-161.
16. Holten Rasmussen, M., Lefrançois, M., Bonduel, M., Anker Hviid, C., & Karlshøj, J. (2018, August). OPM: An ontology for describing properties that evolve over time. In *CEUR Workshop Proceedings* (Vol. 2159, pp. 24-33). CEUR Workshop Proceedings.
17. Wagner, A., Bonduel, M., Pauwels, P., & Rüppel, U. (2019, July). Relating geometry descriptions to its derivatives on the web. In *Proceedings of the 2019 European Conference for Computing in Construction* (pp. 304-313). University College Dublin.
18. Bonduel, M., Wagner, A., Pauwels, P., Vergauwen, M., & Klein, R. (2019). Including widespread geometry formats in semantic graphs using RDF literals. In *2019 European Conference on Computing in Construction* (pp. 341-350). European Council on Computing in Construction.

19. Janowicz, K., Haller, A., Cox, S. J., Le Phuoc, D., & Lefrançois, M. (2019). SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *Journal of Web Semantics*, 56, 1-10.
20. Daniele, L., den Hartog, F., & Roes, J. (2015, July). Created in close interaction with the industry: the smart appliances reference (SAREF) ontology. In *International Workshop Formal Ontologies Meet Industries* (pp. 100-112). Cham: Springer International Publishing.
21. Werbrouck, J., Hagedorn, P., Donkers, A., & Fauth, J. (2026). Semantic Web Technologies for Building Permitting. *Digitalisation and Innovation of Building Permit Processes for a Sustainable Built Environment*.
22. Dridi, A., Patlakas, P., Lefrançois, M., Beach, T., Petr, H., & Vakaj, E. (2024). Aec3po: A knowledge model for machine executable construction regulations. *Available at SSRN 4784123*.
23. Corry, E., Pauwels, P., Hu, S., Keane, M., & O'Donnell, J. (2015). A performance assessment ontology for the environmental and energy management of buildings. *Automation in Construction*, 57, 249-259.
24. Moreira, J., Donkers, A. J., Pauwels, P., Bektas, E., & van Ee, T. (2024). Onto4reuse: towards an ontology reuse framework for knowledge-intensive software engineering. In *JOWO 2024: The Joint Ontology Workshops: Proceedings of the Joint Ontology Workshops (JOWO)-Episode X: The Tukker Zomer of Ontology, and satellite events co-located with the 14th International Conference on Formal Ontology in Information Systems (FOIS 2024)*. CEUR-WS. org.
25. Schneider, G. F. (2019, June). Automated ontology matching in the architecture, engineering and construction domain-a case study. In *LDAC* (pp. 35-49).