BFO-Based Ontology Enhancement to Promote Interoperability in BIM

Justine Flore Tchouanguem[1], Mohamed Hedi Karray[1], Bernard Kamsu Foguem[1], Camille Magniont[2], F. Henry Abanda[3*1], and Barry Smith[4]
[1]Université Fédérale de Toulouse Midi-Pyrénées, INP-ENIT, Tarbes, France
[2]LMDC, Université de Toulouse, INSA, UPS, France
[3]Oxford Brookes University, UK
[4]University at Buffalo, Buffalo NY, USA

Abstract. Building Information Modelling (BIM) is a process for managing construction project information in such a way as to provide a basis for enhanced decision-making and for collaboration in a construction supply chain. One impediment to the uptake of BIM is the limited interoperability of different BIM systems. To overcome this problem, a set of Industry Foundation Classes (IFC) has been proposed as a standard for the construction industry. Building on IFC, the ifcOWL ontology was developed in order to facilitate representation of building data in a consistent fashion across the Web by using the Web Ontology Language (OWL). This study presents a critical analysis of the ifcOWL ontology and of the associated interoperability issues. It shows how these issues can be resolved by using Basic Formal Ontology (ISO/IEC 21838-2) as top-level architecture. A set of competency questions is used as the basis for comparison of the original ifcOWL with the enhanced ontology, and the latter is used to align with a second ontology – the ontology for building intelligent environments (DOGONT) – in order to demonstrate the added value derived from BFO by showing how querying the enhanced ifcOWL yields useful additional information.

Keywords. Interoperability, ontology, BIM, ifcOWL, Basic Formal Ontology (BFO), top-level ontology

1. Introduction

1.1. Context

Building Information Modelling (BIM) is a process for managing construction information in order to provide a reliable basis for decision making during the life cycle of a building (buildingSMART - The Home of BIM', buildingSMART. , s.d.), (Kensek, 2015). A building information model should be in a format that allows data to be easily exchanged and shared across multiple computing environments. Industry Foundation Classes (IFC) is the leading open data format used throughout the Architecture, Engineering and Construction (AEC) domain. It is incorporated into many BIM software tools, following a strategy pioneered around the use of the STEP (ISO 10303) standard in software systems for computer-aided design. IFC, like STEP, is an open file format specification that is intended to be platform neutral.

In an ideal case, IFC data generated by any BIM authoring tool would be able to be used by any other BIM compliant software for applications such as feasibility studies; sunlight, energy and thermal simulations; cost estimation; materials management; and so on (((Kensek, 2015); (Abanda, Vidalakis, Oti, & M, 2015). This ideal case is achieved where all the terms are defined in IFC itself. Often, however, terms need to be employed which lack IFC definitions. The IFC files generated by one BIM software tool can, in such cases, typically not be automatically ingested and used by other such tools without loss of data or other problems.

This paper describes a strategy for enhancing information exchange in BIM systems by advancing interoperability of software tools following a strategy that is being field tested by the Industrial Ontologies Foundry (IOF) initiative. Ontology has a relatively poor record in the engineering domain because of the dominance of *ad hoc* ontology development, where each local engineering project would define its own ontology. The IOF, in contrast, is developing a single modular suite of engineering ontologies designed to be interoperable from the very start (Karray, et al., 2020). Interoperability itself is commonly defined along the following lines:

[*] Corresponding author: F. Henry Abanda or all other authors. Tel: +44(0)1865483475 E-mail : fabanda@brookes.ac.uk

interoperability =def. the ability of diverse systems, organizations and/or individuals to work together using each other's data, information, skills, expertise, equipment and so forth in order to achieve a common goal.

Similarly to the case of STEP, interoperability through IFC is underpinned by the provision of a common framework that can be used to achieve interoperability of data coming from different sources. The IOF approach rests on the use of ontologies, i.e., of controlled vocabularies in which terms and relational expressions are provided by logical definitions. The most important attempt to solve the problem of interoperability along these lines in BIM is ifcOWL, originally created by Beetz et al. (Beetz, Leeuwen, & Vries, 2009). ifcOWL is an ontology built by exporting IFC terms into an OWL (Web Ontology Language) framework (Beetz, Leeuwen, & Vries, 2009). Unfortunately, the existing ifcOWL has a number of weaknesses, turning above all on the fact that – in contrast to what is the case in IFC itself – the terms in ifcOWL lack definitions. Here, we propose EifcOWL (for 'Enhanced ifcOWL'), which seeks to improve OWL codification of the data involved in building processes through an improved vocabulary, in which BIM data are described in such a way that, with the help of definitions, both machines and humans will be better able to interpret the data exchanged.

1.2. Motivating Scenario

The achievement of interoperability in collaborative environments is an increasingly topical issue and, while much effort has been invested in this area, we still need to find ways to improve the exchange of information between experts and their associated information systems. This problem arises in a specific form in relation to the environmental assessment of a building. A typical building life cycle involves contributions from a number of diverse disciplines, including architecture, project management, environmental engineering, structural engineering, and thermal engineering, each in turn involving many different stakeholders or actors (Tchouanguem, Karray, Foguem, Magniont, & Abanda, 2019).

These various actors use different vocabularies, often employing non-standardized information formats and coding systems in a way that hampers the optimal synoptic exploitation of knowledge about the different aspects of a building. Although IFC and ifcOWL have contributed greatly to facilitating the smooth exchange of information between BIM systems, both have noticeable weaknesses. For instance, IFC and ifcOWL have a limited repertoire of natural language definitions of their terms and this impedes interoperability where human beings are involved. It is a major drawback, especially where non-experts in a given discipline wish to use the terms in that discipline, and in this respect, too, it is an obstacle to interoperability. Such problems are further exacerbated by the fact that the export-import capabilities of IFC and ifcOWL are less than effective, for example, in preventing loss of data when the corresponding files are imported or exported into other BIM systems.

There are differences, not only in the packaging of data about the buildings themselves, but also in the ways the databases of materials or products used in the construction of buildings are structured – differences that pose further information exchange challenges. Environmental databases for construction products are diverse and the indicators (e.g. embodied energy intensity) employed, for example in assessing their environmental impacts in building projects are, to a large degree, incompatible with conventional units of measurement. To clarify this remark, let us suppose the embodied energy of a wall is to be assessed. The embodied energy intensity unit is MJ/kg but, in practice, walls are measured in m^2. Therefore, the embodied energy intensity cannot be applied directly; we have to go through intermediate steps to convert from m^2 to kg.

How, then, can interoperability be advanced across these multifarious disciplines and domains? In the computer science and artificial intelligence disciplines, research points to the need for a machine processable language with rich semantics for representing the entities in disparate domains in a consistent fashion. This need is being addressed in many domains through ontological research (Torma & Zheng, 2021). Yet, partly due to the still faltering uptake of BIM, it remains difficult to fully exploit the linking capacities between BIM, ontologies, and databases of construction products.

This study focuses on the newly proposed ifcOWL approach to providing a Web Ontology Language (OWL) representation of the Industry Foundation Classes (IFC) schema. The methodology we employed in building ifcOWL has been well tested in multiple domains and highlights the advantages brought by the use of the top-level Basic Formal Ontology (BFO) (ISO/IED:21838-2) in the creation of domain ontologies at lower levels.

2. Ontologies

2.1. What is an ontology?

We follow Arp et al. (Arp, Smith, & Spear, 2015) in viewing an ontology as a kind of representational artefact whose purpose is to capture what is general in reality by representing universals, defined classes and the relations between them by using some combination of definitions, axioms, rules and constraints. The backbone of an ontology is its taxonomical hierarchy, which consists of the ontology's terms joined together by what are called *is_a* – for 'is a subclass of' – relations. This backbone is supplemented by other relations, such as *part_of*, holding between the entities represented by these terms.

The goal of ontology is to enable knowledge sharing and reuse by means of a definitive classification of entities in specific domains constructed on the basis of a controlled vocabulary with logical definitions of its terms. To achieve this goal, the content of an ontology should be validated by human experts in the corresponding domain (Arp, Smith, & Spear, 2015) (Ont18, s.d.) (Staab & R. Studer, 2009)– (Smith, Ontology (Science), 2008), specifically – for our purposes here – the domain of buildings and construction. Use of BFO facilitates ontology validation, since it is a small, simple and stable ontology, resting on principles that have been tested over some 20 years and are well understood by many who are active in the ontology field.

2.2. Basic Formal Ontology

Basic Formal Ontology (BFO) is a top-level, and therefore very general, ontology developed to support information retrieval, analysis and integration in diverse domains (Arp, Smith, & Spear, 2015). BFO has been widely accepted in a number of application domains and has recently been accepted to become ISO standard ISO/IEC 21838-2. BFO consists of two main categories of entities: continuants and occurrents. Continuants are entities that continue to exist through time while maintaining their identity – for instance a building or the design of a building. Occurrents are entities that occur, happen, unfold or evolve through time. A BFO occurrent can be either an entity that unfolds itself in time, such as a process of construction, or the instantaneous boundary of such an entity, for example, the beginning or ending of a process of construction.

BFO was developed initially to support integration of scientific data obtained through research. It has been used for this purpose in a growing number of ontology initiatives since 2005. BFO brings a number of benefits, including:

a) serving as a domain-neutral, common starting point for ontology building by those who work with specialist knowledge;
b) providing a common upper-level to support the interoperability of the multiple domain ontologies created with its terms;
c) promoting portability of expertise – a person who has been trained in its use in one area can easily apply the same method in other areas;
d) helping to ensure that ontologies built on its basis represent the universals in their respective domains in a consistent and coherently structured fashion;
e) supporting the work of scientists and engineers at multiple scales and levels of aggregation;
f) supporting the integration of data relating to such multiple levels.

Currently, 300 ontology initiatives are using BFO in order to exploit these benefits, including a number of collectively managed suites of ontologies organized in a hub-and-spokes structure in which BFO serves as the hub. Examples in the engineering domain are provided in (BFO21, 2021).

2.3. Compliance with BFO principles

IFC, like the STEP standard (Feeney., 2002) on which IFC is based, deals essentially with information artefacts. In STEP, every design object is called a 'product'. It is not a material product but rather an abstract entity, analogous to a geometrical figure or a subroutine in a computer program. In BFO, and in BFO-conformant domain ontologies, two different kinds of continuant entity are recognized: material entities, on the one hand (such as, again, a building), and information entities such as designs, pieces of software, algorithms, and so forth on the other (Smith & Ceusters, Aboutness: Towards Foundations for the Information Artifact Ontology, 2015).

An ontological supplement to STEP or IFC that uses BFO as its basis is thus able to capture both data about designs and data about the material entities (for example the buildings) in which a small proportion of such designs are materialized. It is clear that many aspects of the material world will impact the designs of material entities – for example the usage for which they have been designed. An ontology based on BFO would be in a position to represent these impacts by representing, within a single framework, both the design and the real, material environment in which it is to be realised. There is a wide range of domain ontologies already developed in BFO terms, including an environment ontology (Env19, s.d.), a maintenance Ontology (Karray, Ameri, Hodkiewicz, & Louge, 2019), a skill ontology (Ski19, s.d.), digital construction ontologies (Dig21, 2021) and others (BFO21, 2021).

2.4. How to build an ontology

The principal aim of any ontology is to support exchange of information on the basis of certain underlying semantic principles. To accomplish this, an ontology should have clear and easily accessible documentation (Karray M. , 2012). In addition, it is essential for each node of the ontology to be accompanied by a definition – ideally both in natural language and in an ontology language like OWL – and by lists of synonyms. Arp et al. suggested 25 principles that should be adopted in ontology development. Some of these principles are captured, together with those proposed by Smith, in (Smith, BSm06, 2006). Most of these principles are grounded in the fact that, since ontologies are built to be shared across different domains, it is beneficial if the ontologies used share a common upper layer of well-defined terms and accept common principles that have been thoroughly tested. This upper layer should be domain neutral, since its aim is to represent the most general categories of entities and the most general relations within and between them, categories and relations shared by all ontologies at lower levels.

3. Building-related ontologies

Reviews of existing ontologies in the building construction domain are provided in (Abanda, Tah, & Keivani, 2013), (Grzybek, Xu, Gulliver, & Fillingham, 2014) and (Pauwels, Zhang, & Lee, 2017). Rather than duplicating these efforts, we focused here on ontologies built on the basis of IFC. However, to enhance understanding, we first provide an overview of IFC itself.

3.1. Industry Foundation Classes

Born from an initiative of the IAI (International Alliance for Interoperability, later renamed 'buildingSMART'), the Industry Foundation Classes (IFC) comprise an object-oriented format based on the STEP standard (ISO 10303). IFC is also governed by the ISO 16739 (2016) standard and, like STEP (Feeney., 2002), employs the EXPRESS-G language (Wix, 2015), (Indbsmart18, s.d.) for its representation. The IFC data model allows users to uniformly represent building data according to the specifications of the IFC schema. IFC defines semantics, relations, and properties of data as follows:

- IFC semantics refers to the meaning (machine-readable unique identifier, object type or function) of the data for straight use by diverse construction project participants.
- IFC relations define how the data are linked by means of cross-references within an IFC data file
- IFC properties include geometric properties (for example dimensions of the object), physical properties (nature of materials used and manner of use), and qualitative data pertaining to the object (unit price, manufacturer, and so on). Note that, as explained above, these properties are not properties of a material entity, but rather properties associated with the design of a material entity.

In IFC, data are encoded in three formats:

- IFC: based on the STEP physical file structure defined by the ISO 10303-21 standard;
- ifcXML: the XML translation of IFC defined by the ISO 10303-28 standard;
- ifcZIP: a compressed archive format involving either of the previous two formats and potentially including additional content such as PDF files, images, and so forth (IFC18, s.d.).

The most recent version of IFC is the Industry Foundation Classes Version 4 – Addendum 2 [20, p. 4].

Multiple authors have applied the IFC EXPRESS schema to construct ontologies with the goal of improving interoperability in the built environment (Pauwels P. , 2014) (Terkaj & Pauwels, 2016) (Env19, s.d.). Unfortunately, this has led to the creation of a variety of partially non-interoperable ontologies, including ifcOWL (Abanda, Tah, & Keivani, 2013), on which we shall focus here, but also COBieOWL (Farias, Roxin, & Nicolle, 2015), ifcWoD (Farias, Roxin, & Nicolle, 2015), DOGONT (Bonino & Corno, 2008), and FOWLA (Farias, Roxin, & Nicolle, 2016). This growth in the number of BIM-based ontologies also stems from the fact that, as we shall see, most of them have only limited expectations as concerns what an ontology can achieve.

Table 1 Design principles for a good ontology

	Principle	**Description**
1	Use singular nouns or noun phrases	To ensure consistent noun-verb agreement.
2	Use lower case italic format for common nouns	To be in accordance with English language rules and for cross-ontology coordination.
3	Avoid acronyms	To avoid oversights due to the short life and context-dependent use of acronyms.
4	Associate each term in the ontology with a unique alphanumeric identifier	This identifier could uniquely locate a term in the hierarchy, for computer programming purposes or when a new version of the ontology is published, for example in a different language.
5	Ensure univocity of terms	For each term, the meaning should be invariant regardless of the context of use.
6	Ensure univocity of relational expressions	Each relational expression should always have the same meaning.
7	Avoid mass nouns	Mass nouns ('sugar', 'water') denote something that cannot be associated with a definite articular or with count. Use of mass nouns creates ambiguity as to whether we are talking about the item or the stuff of which it is made.
8	Distinguish the general from the particular	Terms in an ontology should represent what is general. Thus, an ontology is a T-Box (for 'terminology'). It should be combined with an A-Box (for 'assertions'), when reference to individuals is needed.
9	Provide all non-root terms with definitions	Provide a definition for each term to ease understanding and to facilitate reuse beyond the specific context in which it was defined.
10	Use Aristotelian definitions (also called definitions via genus and specific difference)	A definition of the term 'S' should take the form S =def. G that Ds where 'G' is the immediate parent term of 'S' and 'D' is that which differentiates instances of S from instances of G which are not instances of S.
11	Use essential features in defining terms	Essential features of a thing define what the thing really is. They are the constant elements in its structure.
12	Start with the most general terms in your domain	Define the terms from the top down in the ontology hierarchy.
13	Avoid circularity in defining terms	Do not define a term by using the term itself or a near synonym.
14	To ensure the intelligibility of definitions, use simpler terms than the term you are defining	Definitions should facilitate understanding of the term defined and thereby promote exchange of ontologies across disciplines and domains.

15	Do not create terms for universals via logical combination.	For example, avoid disjunctive or negative terms for universals and classes.
16	Definitions should be unpackable	Definitions should be substitutable for their defined terms without a change in meaning or truth value.
17	Structure every ontology around a backbone *is_a* hierarchy	Each ontology should have an *is_a* hierarchy having the structure of a directed, rooted tree.
18	Ensure *is_a* completeness	Every term in the hierarchy must be joined to the root of the tree by a path constituted by successive edges in the graph. Ensure ontological agreement between terms and their parents.
19	Ensure asserted single inheritance	Each non-root term has exactly one parent in the hierarchy.
20	The principle of openness	The ontology should be freely available for use without constraints.

3.2. ifcOWL

Since around 2000, a number of ontologies have been proposed in the domain of building construction. Starting from the first version of ifcOWL (Beetz, Leeuwen, & Vries, 2009), Pauwels et al proposed significant enhancements (Pauwels P. , 2014) (Pauwels, Zhang, & Lee, 2017) (Pauwels & Terkaj, 2017). Semantic web technologies were then brought to the AEC domain in (Pauwels & Van Deursen, IFC-to-RDF: adaptation, aggregation and enrichment, 2012), where an ontology for the IFC generated by converting the EXPRESS schema into an OWL ontology was proposed.

In arguing for the role of semantic web technologies as a solution to overcoming interoperability challenges in the building domain, Pauwels et al. (2017) took advantage of the production of the ifcOWL ontology by the buildingSmart Linked Data Working Group (LDWG) (Pauwels P. , 2016). ifcOWL is an ontology for the building domain whose main purpose is to support the conversion of IFC files into equivalent Resource Description Framework (RDF) files. It provides an OWL representation for the IFC schema and IFC data that have been made available in the form of a labelled oriented graph (RDF). ifcOWL is the most robust implementation of an IFC-based ontology. Its most recent version was created in 2016 and proposed for data organization and conversion in the construction industry and associated domains by the W3C Linked Building Data Community Group (Abanda, Tah, & Keivani, 2013) in the same year.

3.3. ifcWoD

The IFC Web of Data Ontology (ifcWoD) was introduced by de Farias et al. (Farias, Roxin, & Nicolle, 2015), with the rationale that the existing version of ifcOWL (IFC2X3.owl) does not fully exploit the capabilities of OWL. To rectify the resulting problems, de Farias et al. (Farias, Roxin, & Nicolle, 2015) translated certain attributes in the IFC schema into OWL object properties rather than classes and instances. The resulting ifcWoD ontology is useful for certain BIM purposes. Thus, it facilitates the writing of requests, optimizes their execution, and reduces the redundancy of data by increasing the capacity of what is derivable from reasoning engines. Unfortunately, however, these features are not sufficient to improve interoperability with other ontologies in the same field, since ifcWOD inherits the weaknesses of the version of ifcOWL that forms its basis. In order for ifcWoD to address interoperability issues, therefore, ifcOWL would itself need to be improved.

3.4. COBieOWL

COBieOWL (Farias, Roxin, & Nicolle, 2015) is an OWL ontology based on the Construction Operations Building Information Exchange (COBie) standard created in late 2006 under the auspices of the National Building Information Modelling Standard (NBIMS) (East E. W., 2007). COBie serves the publication of building information models that are focused on delivering asset data, as distinct from geometric information. COBie itself provides data in STEP or in other static formats such as those used by standard spreadsheets. It thus lacks logical formalism and semantic features. COBieOWL was developed to fill this gap.

Like IFC, COBie enhances BIM interoperability. However, its coverage domain is exclusively that of the COBie standard, which means that it is limited in the benefits it can bring to the wider AEC domain. Furthermore, COBie was made only for contractors, builders, designers and facility managers (East E. W., 2007), rather than for all those involved in the construction project. Lastly, the majority of BIM software in the market is still unable to read and/or generate COBie files.

3.5. FOWLA

The Federated Architecture for OWL Ontology (FOWLA) was proposed by de Farias et al. (Farias, Roxin, & Nicolle, 2016) to improve the interoperability of BIM at the level of data. FOWLA is a rule-based federated architecture aiming to leverage semantic web technologies to support interoperability and to resolve data structure heterogeneity issues between different AEC/FM (Facility Management) ontologies. FOWLA brings two main advantages. First, it allows users to write queries by combining terms from ifcOWL, COBieOWL and ifcWoD. Second, it builds on a rule engine that allows new alignment rules to be inferred. For example, transitivity is used to automatically deduce that COBieOWL is aligned with ifcWoD, which is in turn aligned with ifcOWL. Unfortunately, leveraging heterogeneity at vocabulary level, i.e., between the terms used in these different ontologies, is not sufficient to allow semantic interoperability between actors in the AEC industry. In practice, stakeholders often use the same term but associate with it completely different semantics. (Farias, Roxin, & Nicolle, 2016), which is still the only paper on FOWLA, lacks details as to how this issue will be rectified.

3.6. DOGONT

The Domotic OSGi Gateway ONTology (DOGONT) is an ontology created by Dario Bonino in 2008 (Bonino & Corno, 2008) to support domotic (which is to say: home automation) environments. Home automation is the action of centralizing the control of a building's systems: ventilation, heating, lighting, etc., to improve occupant comfort and efficient operation of building systems. It can also reduce the energy consumption and operating costs, while improving life cycle utilities. The result is a so-called "smart home". DOGONT consists of just four classes: BuildingThing, BuildingEnvironment, State, and Functionality (see Table 2).

Table 2 DOGONT classes (Wix, 2015)

Class	**Role**
BuildingThing	models available things, either controllable or not
BuildingEnvironment	models the places where things are located
State	models the stable configurations that controllable things can assume
Functionality	models what controllable things can do

However, even though DOGONT was very well developed, with few errors (OOPS! - OntOlogy Pitfall Scanner! - Results, s.d.), (fundacionctic / parrot — Bitbucket, s.d.), it has hardly been used in any dataset. Furthermore, DOGONT focuses on the operational phase of construction and does not integrate with other phases of the building life cycle such as design, maintenance or demolition. In addition, DOGONT (like almost all ontologies) suffers from the lack of association with a top-level ontology defining high-level terms (such as 'State') in a way that promotes interoperability with other ontologies using the same or related high-level terms.

3.7. Building Topology Ontology (BOT)

The challenge of managing the complex structure of IFC EXPRESS schema has been a major obstacle to success in developing building ontologies. In response, the W3C Linked Building Data (LBD) community group (W3C, 2018) (Rasmussen, Pauwels, Hviid, & Karlshoj, 2017) (TERKAJ, SCHNEIDER, & PAUWELS, 2017) (Building Data on the Web Working Group Charter, s.d.) – (buildingSMART, 2018) has developed an ontology for the AEC domain, called the Building Topology Ontology (BOT). The W3C LBD Community Group defines the BOT as a minimal OWL DL ontology for describing the core topological concepts of a building. This means that BOT includes representations of the relationships between the subcomponents of a building. Its authors have also provided a set of best practices for the treatment of building data on the web and for use of the BTO, which is aligned with ifcOWL and DOGONT (W3C, 2018), as well as with ontologies for the geospatial and sensor domains. Upcoming ontologies from this initiative (covering products, geometry, and the properties of building elements) will also pursue alignment with ifcOWL.

3.8. The importance of IfcOWL

In view of the main goal of IfcOWL, which is to enhance interoperability in BIM, it can be concluded that the efforts described above, although significant, are still insufficient to ensure such interoperability. Due to the popularity of ifcOWL and the broad use of IFC in BIM, it is logical to view ifcOWL as the most appropriate ontology to build on for the future, given that IFC has the ability to cover the entire building life cycle. ifcOWL has also been improved significantly over time (Terkaj & Pauwels, 2016), (IFC4 Documentation, s.d.) and thus has good prospects of being reused in the future, for example through alignment with other ontologies via BOT (Rasmussen, Pauwels, Hviid, & Karlshoj, 2017), (Building Topology Ontology, s.d.). It is thus imperative to critically examine ifcOWL in order to identify gaps before considering any further improvements.

4. Criticism and restructuring of ifcOWL

4.1. Criticismes

Since the introduction of ifcOWL by Beetz et al. in 2009 (Beetz, Leeuwen, & Vries, 2009), a number of criticisms and enhancements have been proposed ((Pauwels, Krijnen, Terkaj, & Beetz, 2017) and (Beetz, Leeuwen, & Vries, 2009)). First, ifcOWL does not comply with many of the principles presented in Table 1. Many classes in the IFC schema do indeed represent entities in the building domain, but almost all classes are under the root class “Thing”. Too many intermediate classes are missing, and this prevents ifcOWL from fulfilling a critical function of an ontology: that of providing a definitive, exhaustive and easily navigable classification of entities in its domain. Furthermore, none of the nodes has a description or a definition. Providing natural language definitions for each term is essential if an ontology is to support coherent (re)use across multiple communities in such a way that the results will be able to support both human understanding and computational reasoning across data deriving from different domains and different sources. Moreover, although ifcOWL includes an *is_a* hierarchy, it includes no partonomy (thus no use of mereological relations such as *part_of*), and therefore also no partonomy relationship, for example, between IfcBuildingStorey and IfcBuilding.

Such problems can be rectified to some degree by drawing from the definitions provided in the BuildingSmart documentation (IFC18, s.d.), where we find, for example:

IFC:IfcBuilding =def. construction work that has the provision of shelter for its occupants or contents as one of its main purposes and is normally designed to stand permanently in one place.

Furthermore, when a storey is specified in the IFC building model, then it is associated in every case with a building – it is part of that building by virtue of its spatial location. The order of spatial structural elements comprising a building project as conceived by IFC goes from high to low: IfcBuilding, IfcBuildingStorey, IfcSpace. Clearly, therefore, a *part_of* relation can easily be included in ifcOWL and should be included to advance ease of use of the ontology.

ifcOWL was not developed from scratch but generated automatically from an IFC data file using the API *EXPRESStoOWL* tool (Pauwels P. , EXPRESStoOWL is a set of reusable Java components that allows to parse EXPRESS files and convert them into OWL ontologies in the context of the Industry Foundation Classes (IFC), 2018). Our goal in this section, however, is not to discuss how ifcOWL was developed, but rather to evaluate its structure and the extent to which it fulfils the interoperability requirements of BIM.
We focus first on the requirements listed in Table 1 above, where we see that:

- ifcOWL complies with principle 1, since it uses only singular nouns.
- ifcOWL includes acronyms, but only those derived from the International System of Units (SI) and from IFC itself. Thus, it comes close to satisfying principles 2 and 3.
- No identifiers are provided for the terms in ifcOWL, thus it is not compliant with principle 4. Identifiers are, nevertheless, indispensable, for instance in tracking terms over time as their definitions and use evolve through successive versions of the ontology.

- ifcOWL is compliant with principles 5-8.
- Since ifcOWL has no ontologically based foundational definitions for any of its terms, it is not compliant with principle 9, and thus its compliance with principles 10–14 and 16 cannot be evaluated.
- No negative terms for universals or classes (principle 15) exist in ifcOWL.
- ifcOWL is not structured around a single backbone *is_a* hierarchy. Although all terms are joined to the root of the tree by a path constituted by successive edges in the graph, the root is not a genuine ontology node but simply the Thing that is hardwired into OWL. Some terms, such as "BINARY", "IfcApplication", "IfcGridAxis" are then isolated from the rest of the hierarchy, since they are linked only via OWL: Thing.
- ifcOWL is not compliant with principle 19 as it embodies cases of multiple inheritance. For example, the class IfcProduct has two parents: IfcProductSelect and IfcObject; IfcProcess has the two parents: IfcObject and IfcProcessSelect; and so forth.
- The principle of openness (20) is respected since the IFC ontology is available to all potential users (licensed by CC 3.0).

4.2. Restructuring ifcOWL

Thus, at best, ifcOWL fulfils only some of the requirements the satisfaction of which is identified by Arp et al. (Arp, Smith, & Spear, 2015) as indicative of best practices from the point of view of ontology-based data system interoperability. In what follows, we aim not to rebuild ifcOWL from scratch, but simply to improve it by adding definitions and by restructuring in such a way that these principles are indeed satisfied. Thus, our goal is to re-engineer ifcOWL on the basis of BFO. The choice of BFO as top-level ontology will help to enhance interoperability with already existing ontologies in neighbouring domains such as the Relation Ontology (Smith, et al., 2005) and the Environment Ontology (EnvO) (Buttigieg, et al., 2013), as well as with the ontologies being developed within the framework of the Industrial Ontologies Foundry (Wallace, Kiritsis, Smith, & Will, 2018).

The version 0.3 of digital Construction Ontologies (Dig21, 2021) has been refactored in compliance with BFO specified in the ISO/IEC 21838-2 standard and this has resulted in a uniform naming of instantiated classes and related properties. Exploration of these construction ontologies will produce exciting new information, which will be used as the basis for future research. At the same time, it should be noted that BFO is not the only candidate ontology that can serve this purpose. The Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) (Gangemi, Guarino, Masolo, & Schneider, 2002), for example, has established itself as a useful top-level ontology in a number of relevant domains, such as hydrology (Brodaric, 2012). However, where BFO has been re-used in a sustainable fashion in many ontology initiatives following a common set of best practice principles, reuse of DOLCE has been more haphazard, in part because DOLCE has not provided its users with the sorts of services provided by BFO (Smith, BSm19, 2019).

5. Enhanced ifcOWL (EifcOWL)

5.1. Enhancing the ifcOWL ontology

Restructuring of ifcOWL using BFO was conducted on the basis of the prior work reported in (Beetz, Leeuwen, & Vries, 2009), (Terkaj & Pauwels, 2016), (Sojić & Terkaj, 2015), (Pauwels, Krijnen, Terkaj, & Beetz, 2017), (Pauwels & Terkaj, A Method to generate a Modular ifcOWL Ontology, 2017) and (Roxin & Pauwels, 2016). We believe that an improvement of the underlying ifcOWL structure, and of its ability to promote interoperability, is essential to provide more semantic enrichment and accuracy of the ontologies. The use of BFO as top-level ontology provides a tested starting point for definitions. **Error! Reference source not found.** summarizes the methodology we used in developing the enhanced ifcOWL ontology and Table 3 summarizes the key categories of BFO employed in this enhancement.

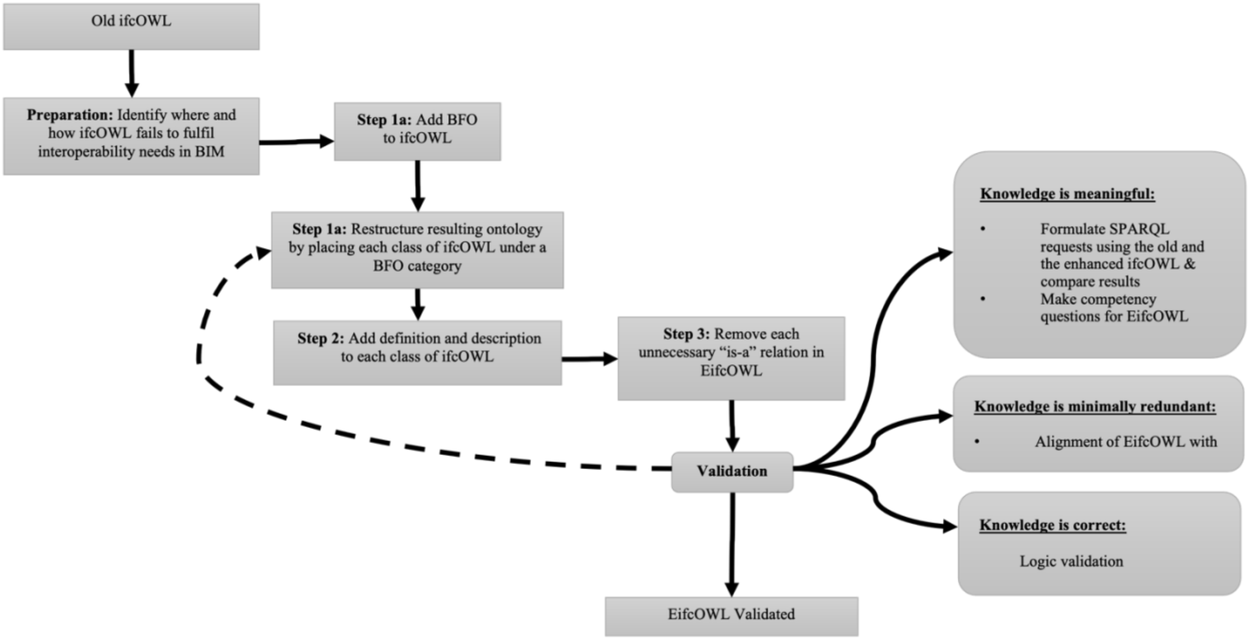

Old ifcOWL
Preparation: Identify where and how ifcOWL fails to fulfil interoperability needs in BIM
Step 1a: Add BFO to ifcOWL
Step 1a: Restructure resulting ontology by placing each class of ifcOWL under a BFO category
Step 2: Add definition and description to each class of ifcOWL
Step 3: Remove each unnecessary "is-a" relation in EifcOWL
Validation
EifcOWL Validated
Knowledge is meaningful:
• Formulate SPARQL requests using the old and the enhanced ifcOWL & compare results
• Make competency questions for EifcOWL
Knowledge is minimally redundant:
• Alignment of EifcOWL with
Knowledge is correct:
Logic validation

Figure 1. The proposed methodology

Table 3 BFO categories (Arp, Smith, & Spear, 2015)

Category	**Definition**
Entity	Anything that exists
Continuant	Entity that continues or persists over time while maintaining its identity and has no temporal parts. It is a dependent or independent object
Occurrent	Entity that occurs, happens, unfolds or develops in time: events or processes or happenings
Independent continuant	A continuant entity that is the bearer of qualities. It can maintain its identity and existence through gain and loss of parts\dispositions\roles, and through changes in their qualities
Generically dependent continuant	An entity that is dependent on one or more other, independent, continuants that can serve as its bearer. It is similar to complex continuant patterns of the sort created through the process of evolution.
Specifically dependent continuant	An entity that depends on one or more specific independent continuants for its existence. It exhibits existential dependence and has two subcategories: quality and realizable entity
Process	Occurrent entity that exists in time by occurring or happening, has temporal parts, and always depends on at least one material entity. It can be partitioned into temporal parts in different ways and at different levels of granularity
Process boundary	The instantaneous temporal boundary of a process. It is the limiting or smallest temporal process part
Quality	An entity that depends on or inheres in an entity at all and is fully exhibited or manifested or realized in that entity

Material entity	An independent continuant that has some portions of matter as part, is spatially extended in three dimensions, and continues to exist through some intervals of time. It has three subcategories: object, fiat object part, and object aggregate
Immaterial entity	An independent continuant that contains no material entities as part. It has three subcategories: site, spatial region and continuant flat boundary
Site	An immaterial entity in which objects are or can be contained. Examples are: cavities, holes, conduits.

5.1.1. Step1: Addition of definitions to ifcOWL

For the purpose of adding definitions to ifcOWL, the definitions of each term in IFC provided by buildingSmart at [20, p. 4] were taken as the starting point. These definitions were then used to annotate the classes of the ifcOWL ontology in creating EifcOWL, using the Aristotelian form recommended in Arp et al. (Arp, Smith, & Spear, 2015) (see also Table 1, item 10), namely :

S=def. G that Ds,

where ‘G’ (for genus) is the immediate parent term of ‘S’ (for species) in the ontology for which the definition is being created. ‘D’ stands for differentia, which is to say ‘D’ tells us what it is about certain Gs that confers the status of Ss on them. An example is provided in **Error! Reference source not found.**.

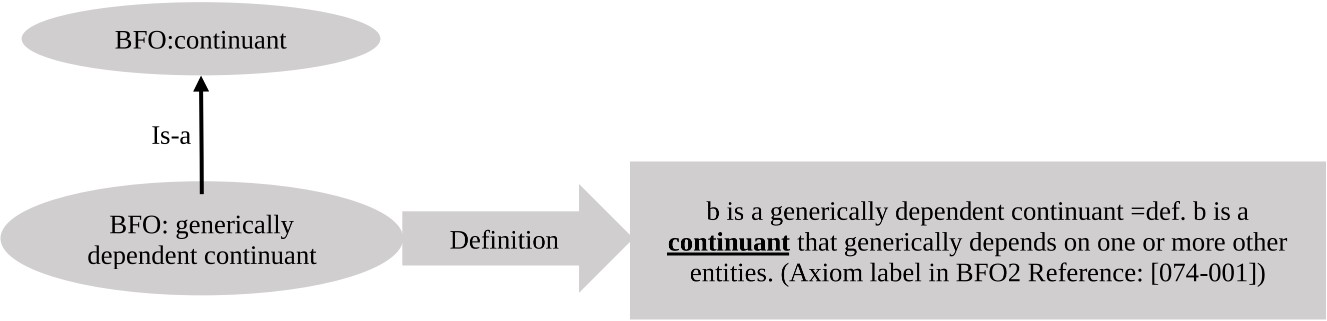
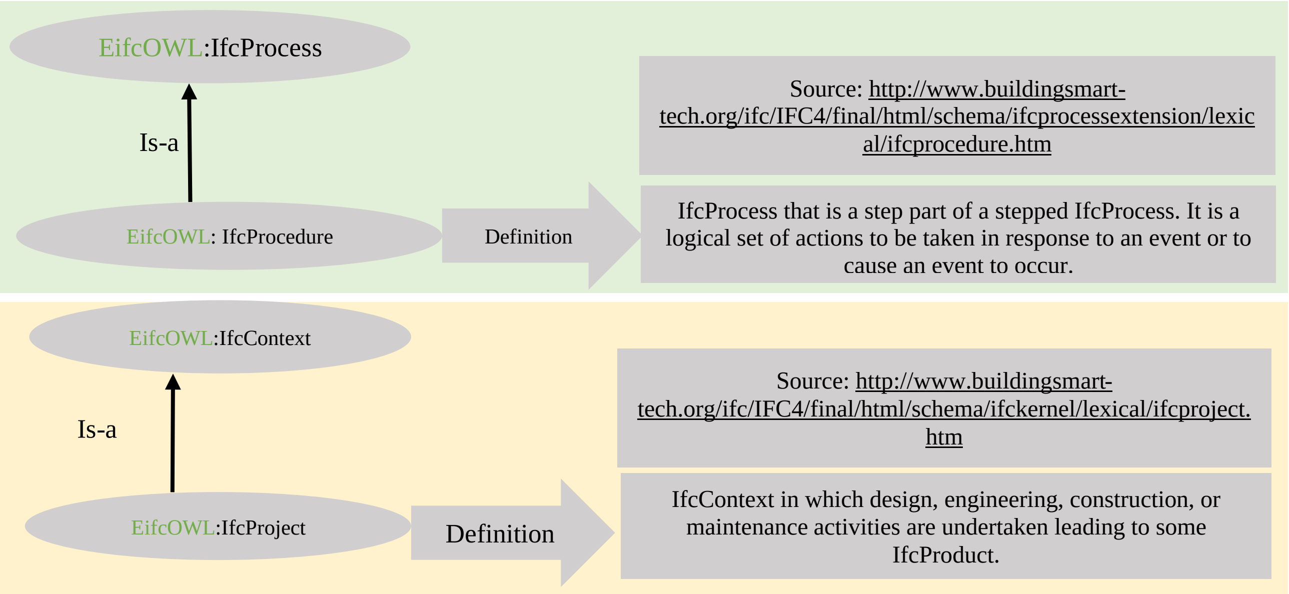

BFO:continuant
Is-a
BFO: generically dependent continuant
Definition
b is a generically dependent continuant =def. b is a **continuant** that generically depends on one or more other entities. (Axiom label in BFO2 Reference: [074-001])

Figure 2 An example of an Aristotelian definition in the BFO ontology

In **Error! Reference source not found.**, *generically dependent continuant* is defined with regard to its immediate BFO parent category of *continuant*, thus following the Aristotelian form. Examples of Aristotelian definitions in EifcOWL are provided in **Error! Reference source not found.** for *IfcProject* and *IfcProcedure*, with parent terms *IfcContext* and *IfcProcess*, respectively. *IfcProcedure* is used to capture information about stepped processes such as calibration, start/stop procedures for equipment items, designated actions to take in the event of an emergency, and so forth. The main use of IfcProject in an exchange structure is to provide the root instance and the context for all other included information items.

IfcProcess is itself defined as:
Eifcowl:IfcProcess =def. process represented by one individual activity or event that is ordered on the basis of their role in building construction phases (e.g. design, acquisition, construction or maintenance of products), which has sequence relationships with other processes, which transforms input into output, and may connect to other processes through input-output relationships. An IfcProcess can be an activity (or task), or an event.



Source: http://www.buildingsmart-tech.org/ifc/IFC4/final/html/schema/ifcprocessextension/lexical/ifcprocedure.htm

IfcProcess that is a step part of a stepped IfcProcess. It is a logical set of actions to be taken in response to an event or to cause an event to occur.

Source: http://www.buildingsmart-tech.org/ifc/IFC4/final/html/schema/ifckernel/lexical/ifcproject.htm

IfcContext in which design, engineering, construction, or maintenance activities are undertaken leading to some IfcProduct.

Figure 3 Examples of definitions following the Aristotelian form in EifcOWL

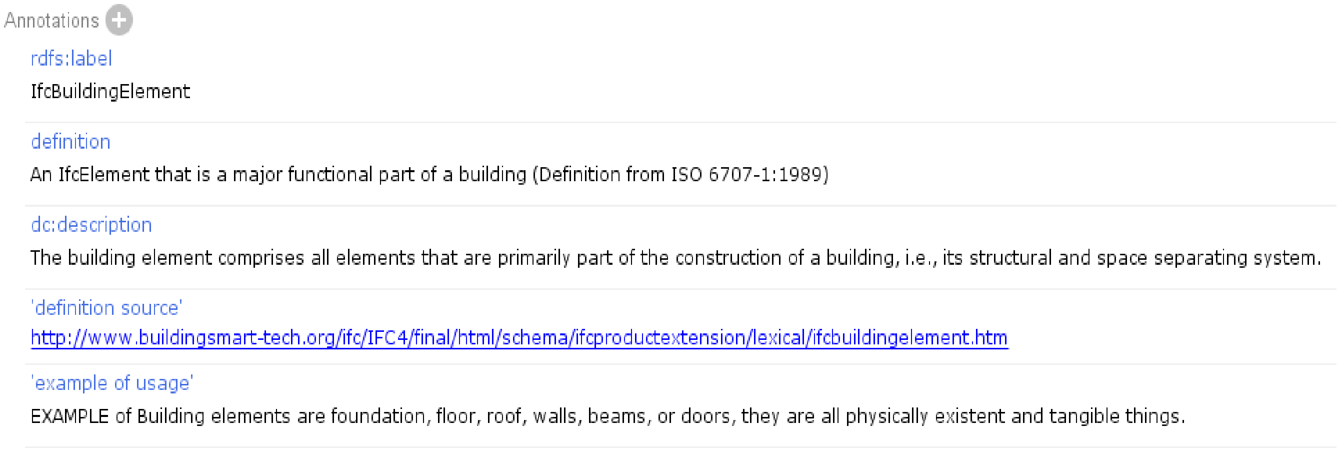
These natural language definitions contain all the elements necessary for the understanding of IFC terms by human beings. The BuildingSmart documentation also contains descriptions and examples of usage for approximately 950 classes of the IFC standard. The examples of usage were added, as displayed in Figure 4, to clarify the meaning of terms in EifcOWL.

5.1.2. Steps 2 & 3: Restructuring ifcOWL using the BFO top-level ontology

In order to link categories of BFO with EifcOWL concepts, we use the starting definitions for each term in its respective ontology. For example, to classify *IfcBuilding* under BFO *object*, we need to use the definitions:

BFO: object =def. material entity that is maximally causally unified (Arp, Smith, & Spear, 2015)

IFC: IfcBuilding =def. construction work that has the provision of shelter for its occupants or contents as one of its main purposes and is normally designed to stand permanently in one place (Indbsmart18, s.d.).



rdfs:label
IfcBuildingElement

definition
An IfcElement that is a major functional part of a building (Definition from ISO 6707-1:1989)

dc:description
The building element comprises all elements that are primarily part of the construction of a building, i.e., its structural and space separating system.

'definition source'
http://www.buildingsmart-tech.org/ifc/IFC4/final/html/schema/ifcproductextension/lexical/ifcbuildingelement.htm

'example of usage'
EXAMPLE of Building elements are foundation, floor, roof, walls, beams, or doors, they are all physically existent and tangible things.

Figure 4 Descriptions and examples of usage in EIFCOWL

EifcOWL: IfcBuilding =def. object under construction or already completed object that has the provision of shelter for its occupants or contents as one of its main purposes and that is a construction or a refurbishment work that is designed to stand permanently in one place.

The restructuring of the concepts reflects the thinking process adopted, which was as follows:
A *continuant* is an entity that persists, endures, or continues to exist through time while maintaining its identity. In contrast, an *occurrent* is an entity that occurs, happens, unfolds or develops in time: events or processes or happenings. Thus, *IfcBuilding*, which represents a structure that provides shelter for its occupants or contents and stands in one place, will be a continuant. In addition, the building is used to provide a basic element within the spatial structure hierarchy for the components of a building project (together with site, storey, and space). The building is (if specified) associated to a site and has a spatial structure that serves as the primary project breakdown and is required to be hierarchical. Whereas *BFO: Immaterial entity* is defined as an independent continuant that contains no material entities as part, *BFO: Material entity* is defined as an independent continuant that has some portions of matter as part, is spatially extended in three dimensions, and continues to exist through some intervals of time. We conclude that *IfcBuilding* is a material entity. Following the same reasoning process, IfcBuilding was classified as a *BFO:object* (a maximally causally unified material entity).

The restructuring of EifcOWL then follows the flowchart in **Error! Reference source not found.**.

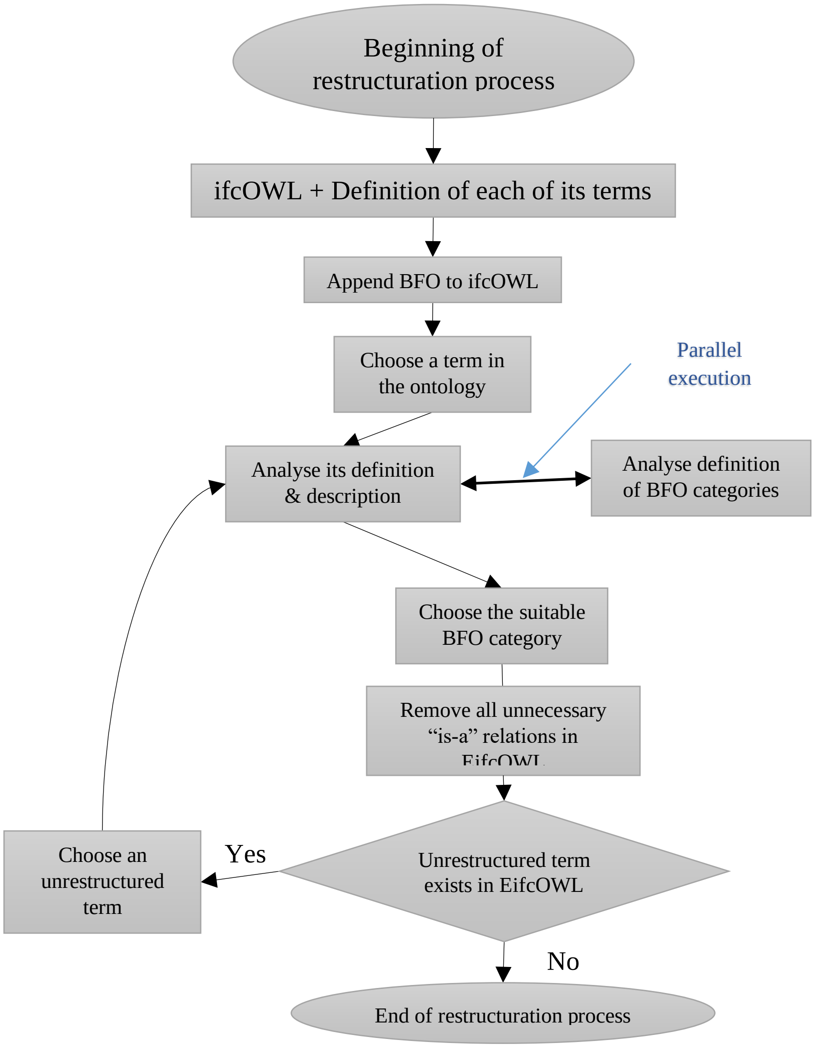

Beginning of restructuration process
ifcOWL + Definition of each of its terms
Append BFO to ifcOWL
Choose a term in the ontology
Parallel execution
Analyse its definition & description
Analyse definition of BFO categories
Choose the suitable BFO category
Remove all unnecessary "is-a" relations in EifcOWL
Unrestructured term exists in EifcOWL
Yes
Choose an unrestructured term
No
End of restructuration process

Figure 5 The restructuration process of ifcOWL

The restructuration process starts after each term of ifcOWL has been associated with the appropriate definition. The latter is then analysed in relation to the BFO categories, a suitable BFO category is chosen, and all unnecessary "is-a" relations are removed from EifcOWL. The process is repeated until there are no more unrestructured terms in EifcOWL.

The top-down aspect of this strategy involves using the BFO category definitions as the starting point for working out, for each ifcOWL class, which category provides the most appropriate fit. The bottom-up aspect consists in reading and understanding the definitions of these classes in ifcOWL. As examples of classes with non-trivial BFO-conformant definitions, consider *IfcProperty* and *IfcPropertyDefinition*, which are classified under BFO: *quality* and BFO: *generically dependent continuant*, respectively.

BFO: Quality is defined as follows:

BFO: Quality =def. specifically dependent continuant that, in contrast to roles and dispositions, does not require any further process in order to be realized

IfcProperty is defined in (IFC18, s.d.) as follows:
IFC: IfcProperty =def. an abstract generalization for all types of properties that can be associated with IFC objects through the property set.

EifcOWL: IfcProperty =def. quality that is an abstract generalization for all types of properties that can be associated with IFC objects through the property set mechanism

Each IfcProperty has attributes such as name, description, partofPSet, etc.

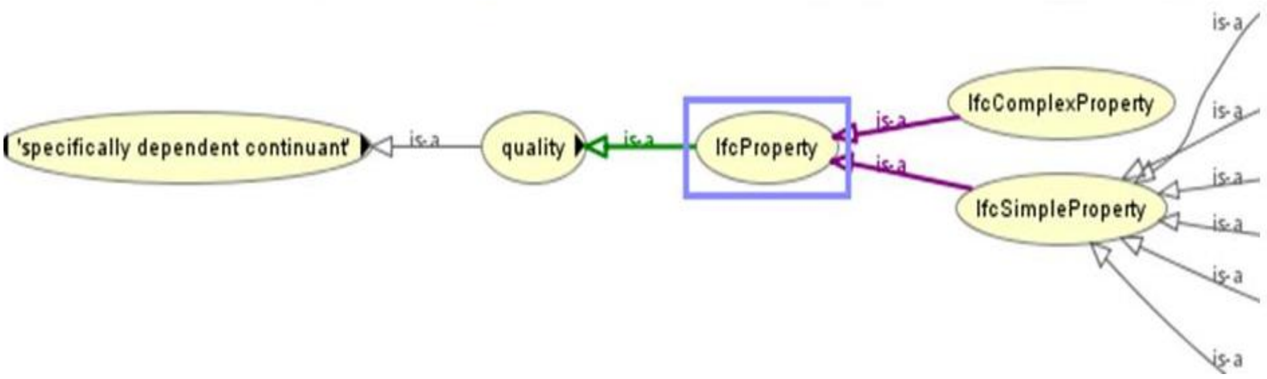


Figure 6 Categorization of 'IfcProperty' classes

BFO:Quality is a specifically dependent continuant, which means that, like all specifically dependent continuants, it depends on one or more specific independent continuants for its existence. BFO: Quality is instantiated only if there is an object that is its bearer, a feature that is also shared with IfcProperty, which exists only if a particular IFC object exists. The latter has two subclasses: IfcComplexProperty and IfcSimpleProperty. Based on their definitions, both of these can be classified under BFO:Quality.

In addition, information provided in (IFC18, s.d.) allows us to understand the other entities on which each *IfcProperty* depends, or which are dependent on it (see **Error! Reference source not found.**).

The digital artifact encodes data, behaviours, attributes, or properties that software applications use to retrieve and display it. The properties attributable to an IFC modelling object can be predefined/generated automatically by the design software or customized by users. So, many properties are attributes assigned to digital artifacts.

Occurrent classes in EifcOWL include: *IfcProcess*, *IfcTimeperiod*, *IfcTimeSeriesValue*, and *IfcDuration*. The structure of these classes is presented in **Error! Reference source not found.**. In fact, IfcTimePeriod is defined (in IFC) as a time period given by a start time and an end time.

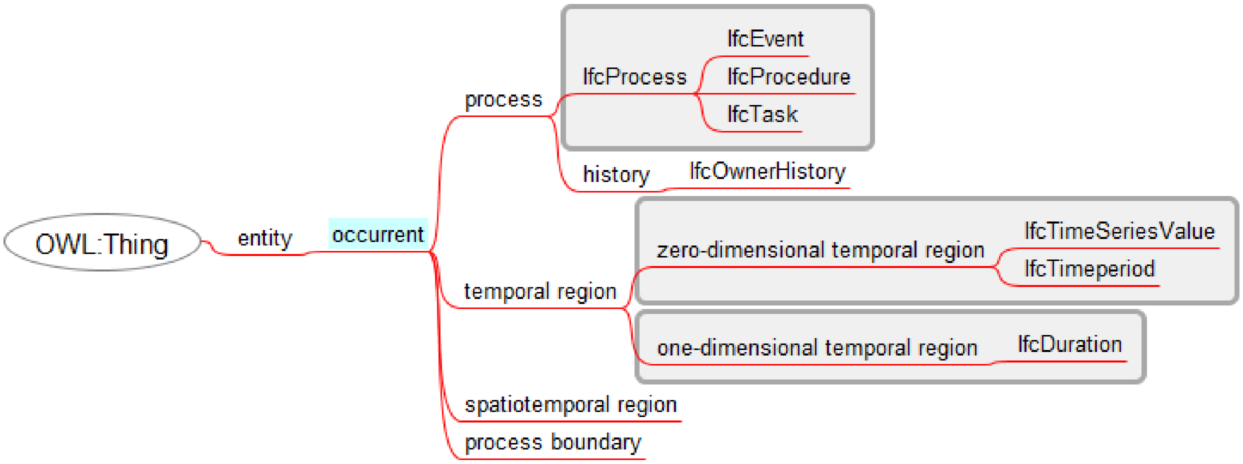


Figure 7 Examples of occurrent subcategories in EifcOWL

IfcEvent, IfcProcedure, IfcTask, IfcOwnerHistory, IfcTimeSeriesValue, IfcTimeperiod and IfcDuration were all classified as "occurrent" subcategories.

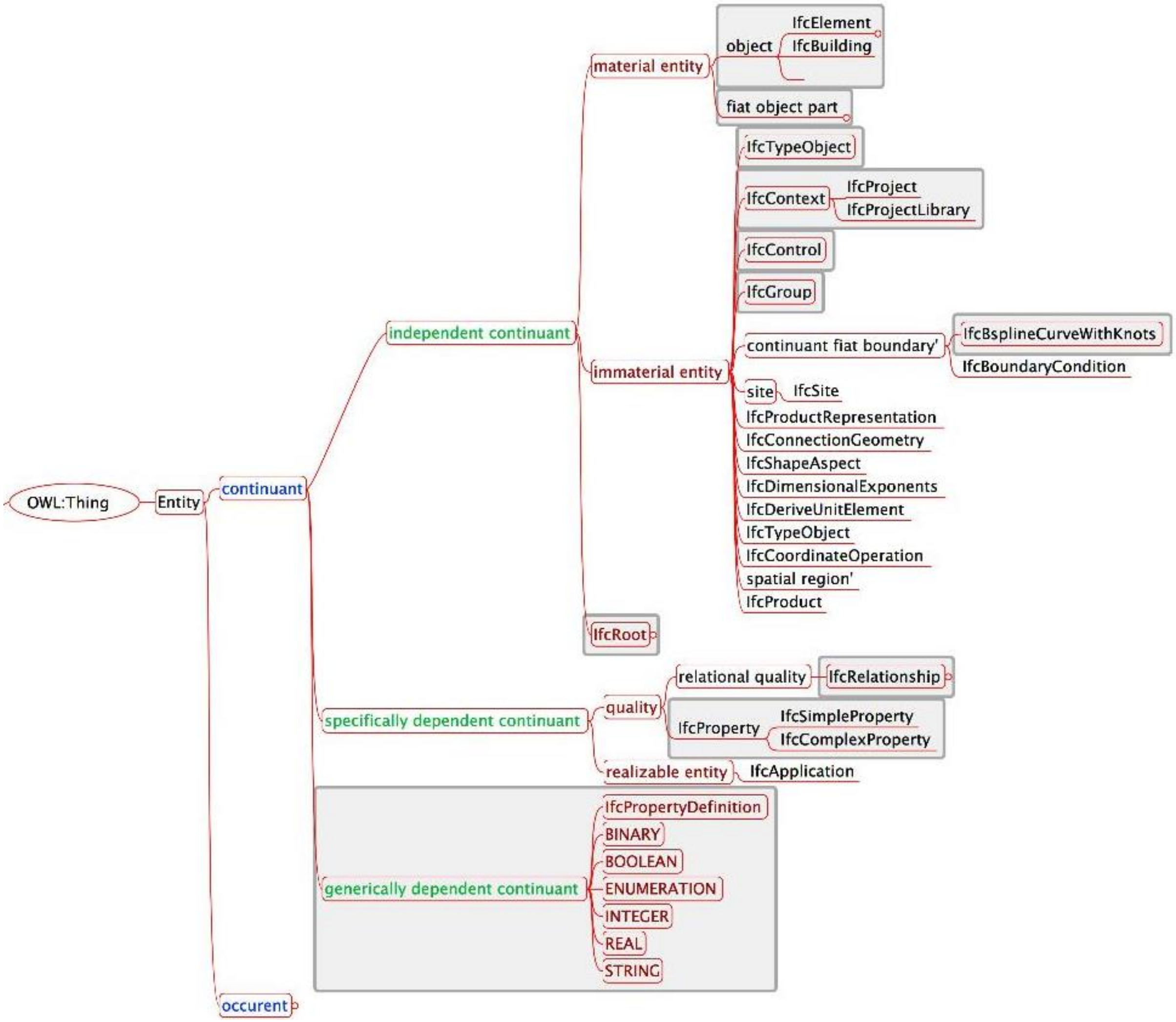


Figure 8 Synopsis of Continuant subcategories

IfcTimeSeriesValue regroups a list of values that comprise the time series. At least one value must be supplied. As time series, they are entities in which an occurrent entity can be located without extent, and only along the temporal dimension. Considering that Zero-dimensional temporal regions (designated parent of IfcTimeSeriesValue in BFO) are "the temporal regions that process boundaries are located in" (Arp, et al.), we suggest, they were the best parent for IfcTimeSeriesValue[2].

We acknowledge the difficulty to classify some of the 12 thousand of IfcOwl classes. Many key terms in ifcOWL, such as IfcBuilding, IfcSite, IfcProduct and IfcRelationship, are classified as subcategories of "continuant" in BFO.

- IfcBsplineCurveWithKnots is, according to ISO/CD 10303-42:1992, the type of b-spline curve for which the knot values are explicitly given. Considering that, we classified it under BFO:Continuant fiat boundary.
- IfcProduct: BFO:Continuant fiat boundary is defined as an immaterial entity that does not include a spatial region as part. IfcProduct is defined as an abstract representation of any object that relates to a geometric or spatial context. It occurs at a specific location in space if it has a geometric representation assigned. Since IfcProduct is not a boundary but is abstract, we decided to classify it as a BFO:immaterial entity.

A synopsis of the structure of EifcOWL regarding BFO: continuant classes. At the end of the restructuring process, a persistent URL was created and the ontology was posted online at http://mindoc.enit.fr/voc/enhancedifcowl.

5.1.3. Comparisons between ifcOWL and EifcOWL

In ontologies that allow multiple inheritance, a class can inherit from more than one class. That is to say, a single class can have more than one superclass parent. Multiple inheritance is supported by OWL, but it can generate additional complexity and, if not carefully handled, it can lead to errors. Conflict resolution in multiple inheritance is managed using prioritizations based on practical priority factors. These factors include the inheritance of more specific class properties and the restriction of the scope of potentially conflicting properties in a local context. In the course of this first phase of the restructuring, it was noticed that EifcOWL still did not comply with the single inheritance principle. To attempt to solve this problem, defined classes were used, since they have necessary and sufficient conditions (equivalent classes) (Masmoudi, Karray, Lamine, Zghal, & Archimede, 2020), which provide a useful means of determining class membership. In ontology, languages standardized for the Semantic Web such as the Web Ontology Language (OWL) and the various Description Logics (DLs), a defined class is a class for which necessary and sufficient conditions have been specified (as in an "if and only if" statement). Any class in which all the individuals satisfy such a definition can be inferred to be a subclass of the corresponding defined class. We can therefore use the defined class as a query to "gather" subclasses that satisfy its conditions.

Examples of such defined classes in EifcOWL are: IfcProcess, IfcFillAreaStyleTiles, IfcProjectionElement. The last is defined as follows:

IfcProjectionElement =def. a specialization of the general feature element to represent projections applied to building elements.

Differences between ifcOWL and EifcOWL are shown in Table 4. First, all classes (35) and axioms (104) from the BFO-OWL ontology were included in the new ontology. However, some relationships are not used in EifcOWL, and the corresponding axioms are therefore not included. On the other hand, some relations holding between EifcOWL classes and BFO categories were added, with corresponding addition of new axioms. Starting from zero definitions in ifcOWL, EifcOWL contains more than 580 definitions.

5.1.4. Semantic enrichment with Enhanced ifcOWL

Figures 7, 8 and 9 highlight further differences between ifcOWL and EifcOWL. **Error! Reference source not found.** presents part of the structure of the *IfcRoot* class and its subclasses as they were structured in the original ifcOWL. **Error! Reference source not found.** and 8 show how those classes are redistributed after the BFO alignment.

[2] See https://standards.buildingsmart.org/IFC/DEV/IFC4_3/RC2/HTML/schema/ifcdatetimeresource/lexical/ifctimeseriesvalue.htm for more on IfcTimeSeriesValue.

Based on the structure of ifcOWL as displayed in **Error! Reference source not found.**, an Ifcrelationship is_a IfcRoot and an IfcProcess is_a IfcObjectDefinition. An IfcGroup or IfcControl is_a IfcObjectDefinition, and so on. This shows structural failings of ifcOWL because the axioms are semantically illogical.

Table 4 Comparison between original and enhanced ifcOWL

Metrics		**Old ifcOWL**	**Enhanced ifcOWL**	**Including BFO**
Class count		1294	1329	35
Class definition Count		0	>555+34 = 589	34
SubClassOf axioms		5035	5060	104
Axioms	Added to		>100	1684
	Deleted from	>50		
	Total	20529	24911	

For example, if we consider a building with two sorts of doors, for example interior doors and exterior doors, then it is important to distinguish them by specifying their respective roles: interior or exterior. This information, which is available in most BIM software, cannot be modelled in the original ifcOWL ontology. Thanks to the additional high-level categories included in EifcOWL, we can model information of this sort very easily.

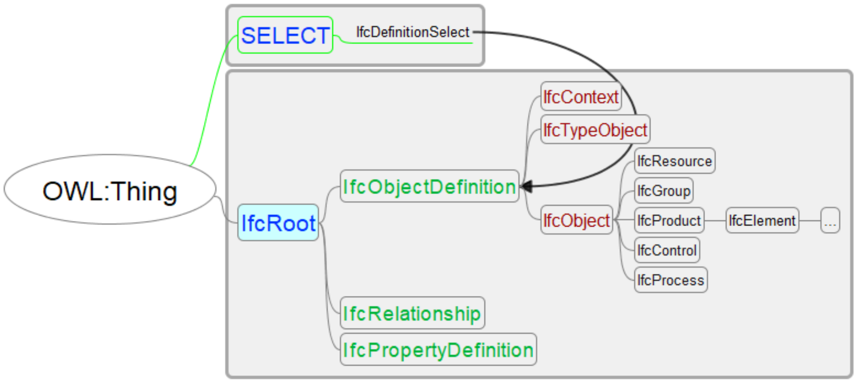


Figure 9 Old ifcOWL structure - IfcRoot and subclasses

6. Evaluation of the enhanced ifcOWL ontology

As proposed by Ian Horrocks (Horrocks, Patel-Schneider, & Harmelen, 2003), a high-quality ontology should be: meaningful, correct, minimally redundant, and richly axiomatized. The evaluation step aims to ensure that the enhanced ifcOWL meets as many of these characteristics as possible. To evaluate EifcOWL, three techniques were used. Firstly, the ontology was checked for its consistency. Secondly, the ontology was aligned with DOGONT, a well-established small ontology that we use to illustrate one use case for EifcOWL. Lastly, the enhanced ontology was queried to establish whether the results were in conformity with our pre-ontological knowledge of the building domain. We also used a case study to demonstrate the value of EifcOWL as the last task of the evaluation.

6.1. Ontology consistency checking

The ontology evaluation was conducted with the Hermit Reasoner in Protégé. The logic validation consisted in verifying the completeness and the correctness of the EifcOWL ontology. For each inconsistency found by the Hermit reasoner, the "Inconsistent ontology explanation" window popped up to indicate an error message. The ontology was rechecked, and the Hermit Reasoner was used to identify inconsistencies. The process was iterated until no error message was found or generated in the "Inconsistent ontology explanation" window. Once this was achieved, the ontology was judged to be consistent.

6.2. *Ontology alignment*

In addition to the amount of information that can be leveraged from EifcOWL using the SPARQL language, the ontology is also useful if an alignment with other building-related ontologies is desired. Ehrig (Ehrig, 2006) defines ontology alignment as the act of finding, for each term in the first ontology, the corresponding term in the second ontology. Ontology alignment will be partial if there is no corresponding second term.

This subsection aims to demonstrate that alignment is facilitated by EifcOWL thanks to the addition of definitions and the use of BFO categories. DOGONT has considerable similarities with other building-related ontologies, including ifcOWL. For instance, the BuildingThing class in DOGONT corresponds to the IfcBuildingElement class. DOGONT is, however, very small and it can thus be useful to the AEC domain only if it can be complemented by other comparable ontologies. With regard to their names and available definitions, DOGONT classes were first binned under BFO categories to facilitate the alignment. EifcOWL was then aligned with the results obtained. The result of the alignment process is presented in **Error! Reference source not found.**, where entities from DOGONT are in bold. The result shows that “device”, “controllable”, “uncontrollable”, “IfcBuilding” and “IfcPort” are all children of BFO: material entity.

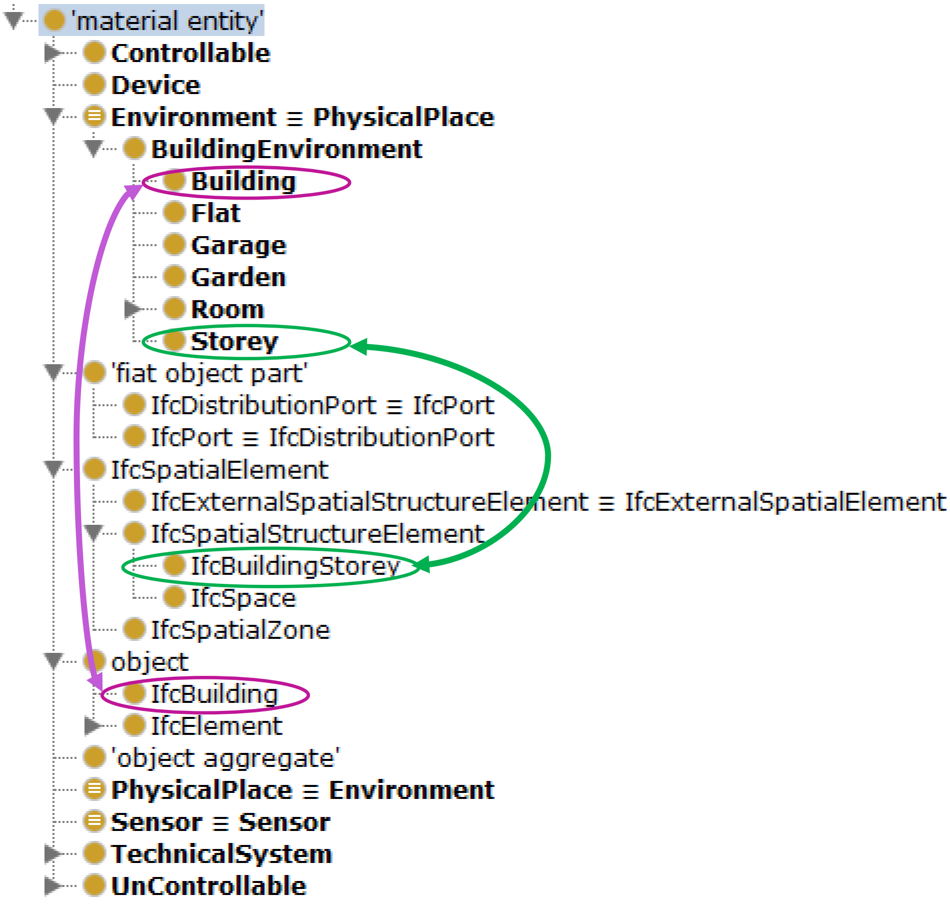
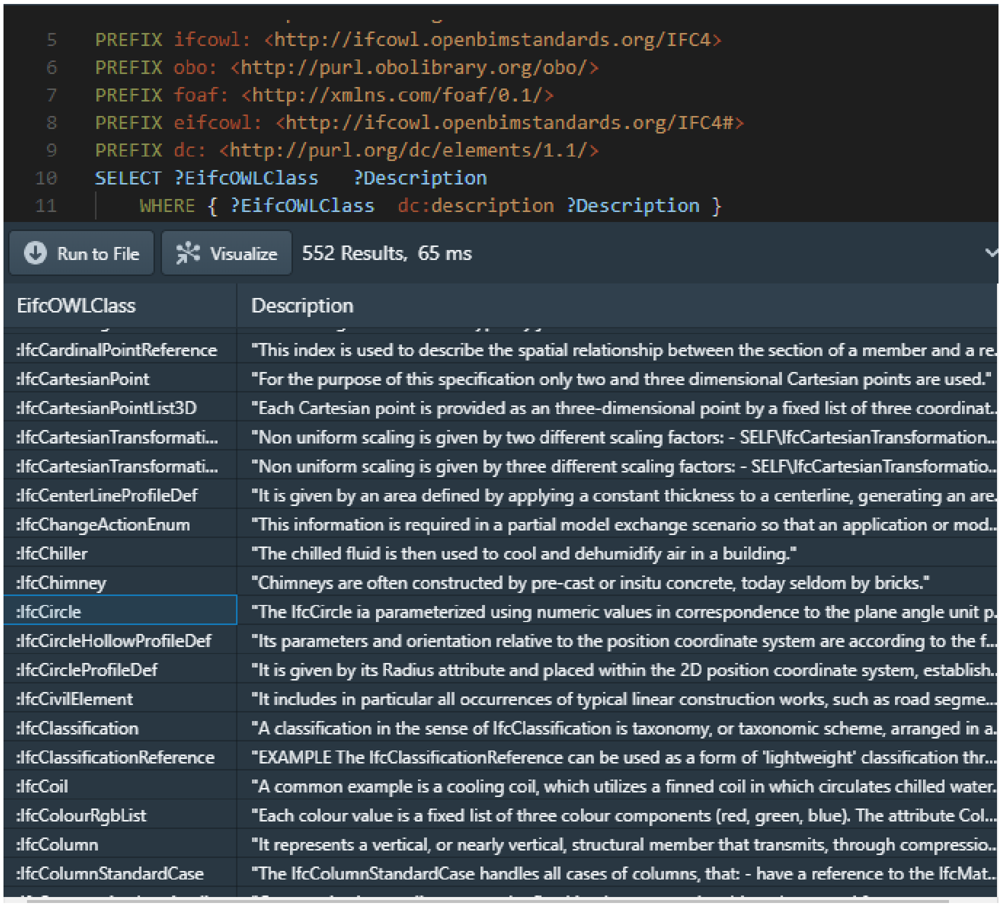


Figure 10 Alignment of material entity terms in ifcOWL with DOGONT

Qualities and functions from EifcOWL and DOGONT were also aligned. Their correspondences can help experts to find useful information about a particular category of elements. As an example, some elements that can be controlled in a building, such as ports (for example distribution ports for water or electricity) can be obtained through the alignment of EifcOWL and DOGONT ontologies as in **Error! Reference source not found.**. The proposed semantic enrichment is oriented towards the practical needs of experts by software applications. From this perspective, BIM tools can address the semantic level of the interoperability challenge with opportunities to pave the way for the eventual integration of information from distinct decentralized sources over a construction lifecycle.

6.3. Querying the enhanced ifcOWL ontology

Since ifcOWL can make IFC data available in RDF graphs, SPARQL can be used for querying IFC data. However, there are no descriptions provided in the existing version of ifcOWL, so ifcOWL needs more annotations to enrich its semantics. In EifcOWL, in contrast, there are many class descriptions, and an example of what results when EifcOWL is queried with SPARQL is shown in Figure 11. This figure presents both the request and the result. The latter substantiates the result of the alignment. Furthermore, the set of material entities (Figure 12) or the set of processes can also be extracted from the new ontology, demonstrating, once again, its great semantic richness.



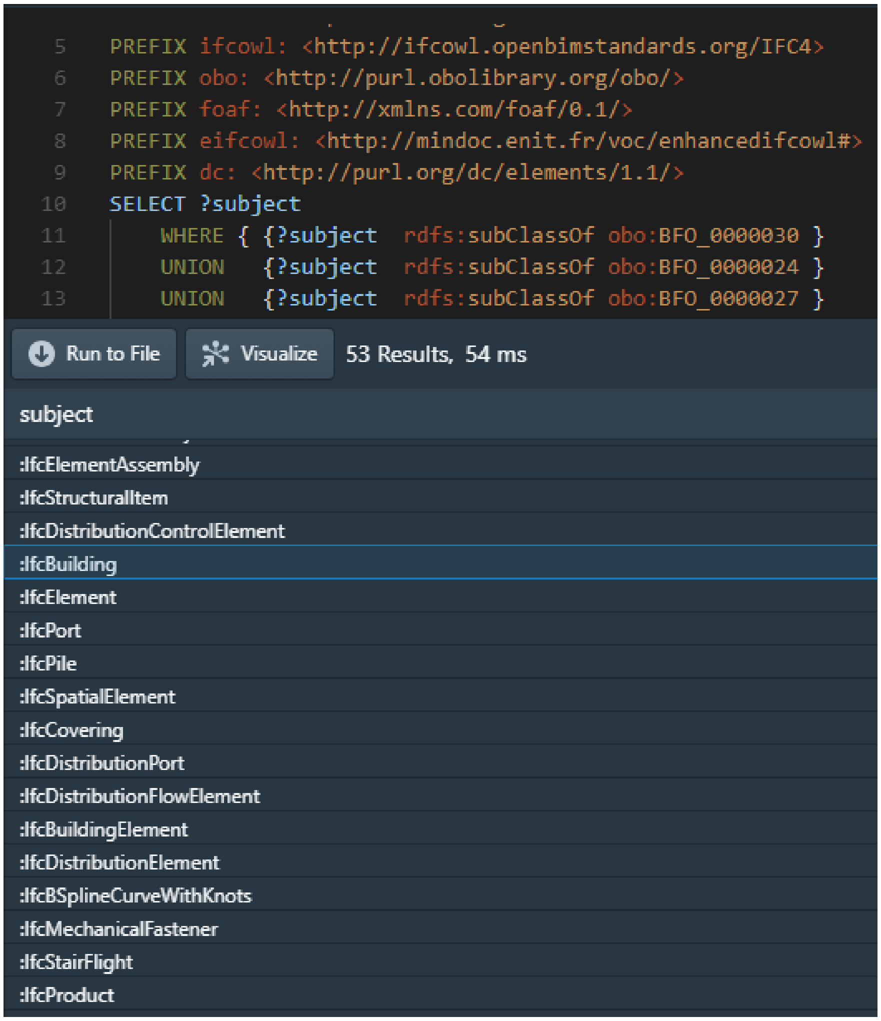
```
PREFIX ifcowl: <http://ifcowl.openbimstandards.org/IFC4>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX eifcowl: <http://ifcowl.openbimstandards.org/IFC4#>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?EifcOWLClass   ?Description
    WHERE { ?EifcOWLClass  dc:description ?Description }
```

552 Results, 65 ms

EifcOWLClass	Description
:IfcCardinalPointReference	"This index is used to describe the spatial relationship between the section of a member and a re.
:IfcCartesianPoint	"For the purpose of this specification only two and three dimensional Cartesian points are used."
:IfcCartesianPointList3D	"Each Cartesian point is provided as an three-dimensional point by a fixed list of three coordinat..
:IfcCartesianTransformati...	"Non uniform scaling is given by two different scaling factors: - SELF\IfcCartesianTransformation...
:IfcCartesianTransformati...	"Non uniform scaling is given by three different scaling factors: - SELF\IfcCartesianTransformatio...
:IfcCenterLineProfileDef	"It is given by an area defined by applying a constant thickness to a centerline, generating an are.
:IfcChangeActionEnum	"This information is required in a partial model exchange scenario so that an application or mod...
:IfcChiller	"The chilled fluid is then used to cool and dehumidify air in a building."
:IfcChimney	"Chimneys are often constructed by pre-cast or insitu concrete, today seldom by bricks."
:IfcCircle	"The IfcCircle ia parameterized using numeric values in correspondence to the plane angle unit p.
:IfcCircleHollowProfileDef	"Its parameters and orientation relative to the position coordinate system are according to the f...
:IfcCircleProfileDef	"It is given by its Radius attribute and placed within the 2D position coordinate system, establish..
:IfcCivilElement	"It includes in particular all occurrences of typical linear construction works, such as road segme...
:IfcClassification	"A classification in the sense of IfcClassification is taxonomy, or taxonomic scheme, arranged in a.
:IfcClassificationReference	"EXAMPLE The IfcClassificationReference can be used as a form of 'lightweight' classification thr...
:IfcCoil	"A common example is a cooling coil, which utilizes a finned coil in which circulates chilled water.
:IfcColourRgbList	"Each colour value is a fixed list of three colour components (red, green, blue). The attribute Col...
:IfcColumn	"It represents a vertical, or nearly vertical, structural member that transmits, through compressio..
:IfcColumnStandardCase	"The IfcColumnStandardCase handles all cases of columns, that: - have a reference to the IfcMat..

Figure 11 Requested descriptions in enhanced ifcOWL. Descriptions are requested through a SPARQL request and are useful both for a human expert and for software.

To show the level of enhancements of EifcOWL over IfcOWL, a comparison between the two ontologies was conducted with the Protégé comparator API. The result (Table 5) clearly shows not only the addition of descriptions to each ifcOWL class, but also the addition of axioms, particularly those concerning subclasses. The enhancements enable many more details to be captured about an IfcBuilding, as can be seen in Table 5, which highlights elements that have been added, changed or renamed from the old to the new version: definition, descriptions, relations, and so on. On the other hand, EifcOWL's changes or additions (e.g. IfcBuilding SubClassOf IfcSpatialStructuralElement) to the superclasses of terms brings the risk of non-compliance with previous versions of the IfcOWL ontology.



```
PREFIX ifcowl: <http://ifcowl.openbimstandards.org/IFC4>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX eifcowl: <http://mindoc.enit.fr/voc/enhancedifcowl#>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?subject
    WHERE { {?subject  rdfs:subClassOf obo:BFO_0000030 }
    UNION   {?subject  rdfs:subClassOf obo:BFO_0000024 }
    UNION   {?subject  rdfs:subClassOf obo:BFO_0000027 }
```

53 Results, 54 ms

```
subject
:IfcElementAssembly
:IfcStructuralItem
:IfcDistributionControlElement
:IfcBuilding
:IfcElement
:IfcPort
:IfcPile
:IfcSpatialElement
:IfcCovering
:IfcDistributionPort
:IfcDistributionFlowElement
:IfcBuildingElement
:IfcDistributionElement
:IfcBSplineCurveWithKnots
:IfcMechanicalFastener
:IfcStairFlight
:IfcProduct
```

Figure 12 Some material entities requested from EifcOWL, i.e. from all EifcOWL classes that are classified under the BFO: material entity category at that time.

Table 5 Ontology differences – examples related to IfcBuilding

Description	Baseline	New Axiom
Superclass changed	IfcBuilding SubClassOf IfcSpatialStructuralElement	IfcBuilding SubClassOf object
Added		IfcBuilding example of usage "The IfcBuilding is used to build the spatial structure of a building (that serves as the primary project breakdown and is required to be hierarchical)."
Added		IfcBuilding definition "object under construction that has the provision of shelter for its occupants or contents as one of its main purposes and it is a

		construction work that is normally designed to stand permanently in one place.”
Added		IfcBuilding definition source http://www.buildingsmart-tech.org/ifc/IFC4/final/html/schema/ifcproductextension/lexical/ifcbuilding.htm
Added		IfcBuilding description “A building represents a structure that provides shelter for its occupants or contents and stands in one place. The building is also used to provide a basic element within the spatial structure hierarchy for the components of a building project (together with site, storey, and space). A building is (if specified) associated to a site. A building may span several connected or disconnected buildings. Therefore, building complex provides for a collection of buildings included in a site. A building can also be decomposed into (vertical) parts, where each part defines a building section.
Added		IfcBuilding label “IfcBuilding”

6.4. Evaluation using a case study

A small-scale experiment was conducted with the goal of demonstrating the value of EifcOWL. The case study concerns a small building (plan) created using Autodesk Revit Software.

We started with a basic building called "BX" (for Building X), as displayed in Figure 13. BX has four walls, one window and one door.

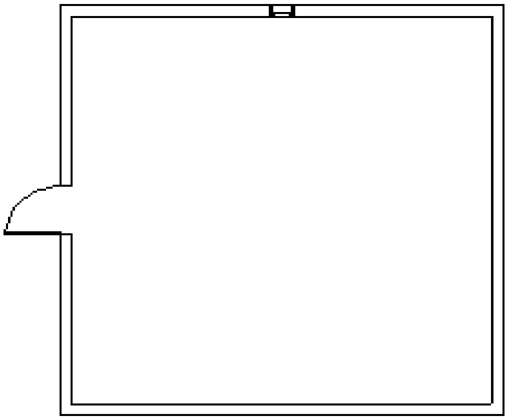


***Figure 13**. The Use case building: Building "X" (BX)*

In the early design phase, we have the geometric dimensions of building elements but otherwise none of the details required to perform a Life Cycle Assessment (LCA) of this building. The available information will evolve throughout the life cycle of BX. Our aim is to minimise the environmental impact of BX throughout its lifecycle. To achieve this goal, many obstacles will have to be overcome. First, in the early design phases, we will need to query various databases of Environmental Product Declarations (EPD) in order to perform an LCA of BX before the execution phase. But the heterogeneity of such databases is a barrier that needs to be overcome. Secondly, we will need to perform the environmental impact assessment (EPA) of BX. However, here, the variety of BIM-based file formats usually results in loss of data due to interoperability issues.

In our case study, following the instantiation of some of the building elements using Autodesk Revit software, we assign a construction product in the integrated database to each building element in the workspace. Linked Building Data (LBD) of the sample is then generated using a dedicated plugin (Tchouanguem Djuedja, et al., 2021) developed in Autodesk. Given that certain experts in domotics would like to automate certain operations in the building, such as opening and closing of windows and doors or lighting, we begin by focusing on those elements that are relevant for such automation, which in principle include all material entities in the building. However, there is no possibility, either for humans or for computer programs, to directly query all material entities of a building. Using ifcOWL, the instantiation produces the output displayed in Figure 14.

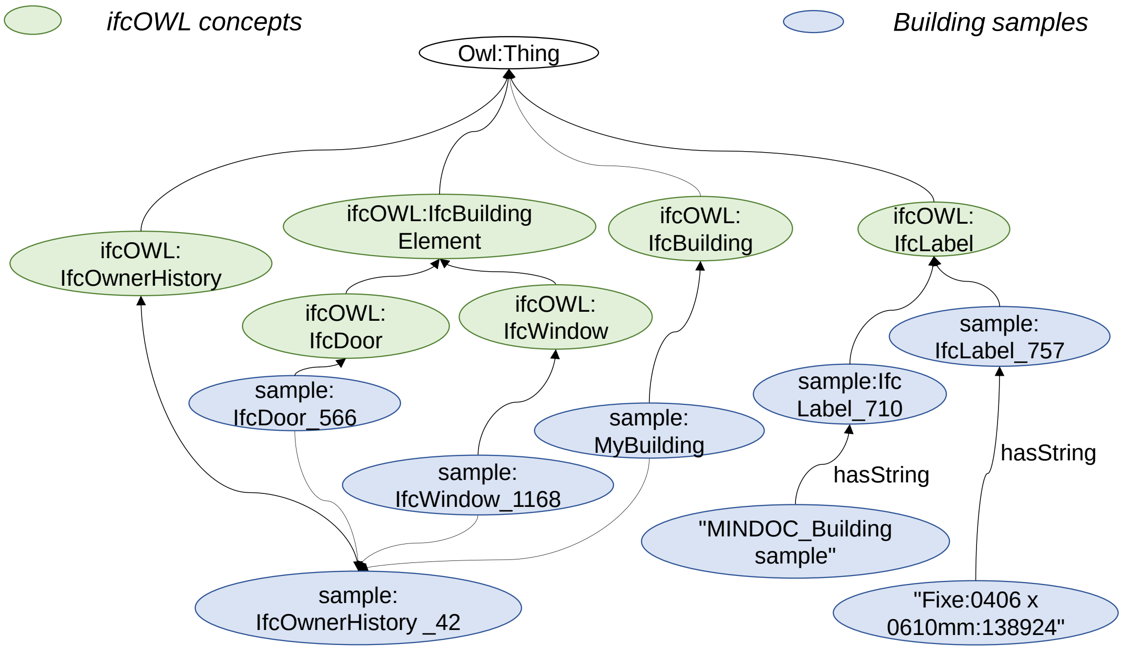


Figure 14. *Instantiation of building BX using the ifcOWL ontology.*

In ifcOWL, the information displayed about building elements such as doors or windows is limited to geometric characteristics. Otherwise, only the globalId, the name, the objectPlacement, the objectType, and labels such as overallHeight, overallWidth, etc., are available. By taking advantage of BFO, function and disposition can be added to the model. Knowing the function of a particular item can be helpful, for instance in designing the automation features of the building or in choosing the appropriate material to associate to the model. For our sample building, we assigned the function *Exterior-door* to a door. In a situation where we need to choose the appropriate material for that door, its specific function will be critical to the process.

To illustrate an additional benefit provided by EifcOWL, it was used as part of a rule base developed to compare various alternative construction solutions (Tchouanguem Djuedja, et al., 2021). This allowed us to verify that the building products chosen by the expert complied with the associated rules. (In the ideal case it would allow us to filter these products beforehand, in order to offer experts only those alternatives which, in a given construction solution, comply with the applicable rules – in the case of (Tchouanguem Djuedja, et al., 2021), rules about the environmental performance of buildings.)

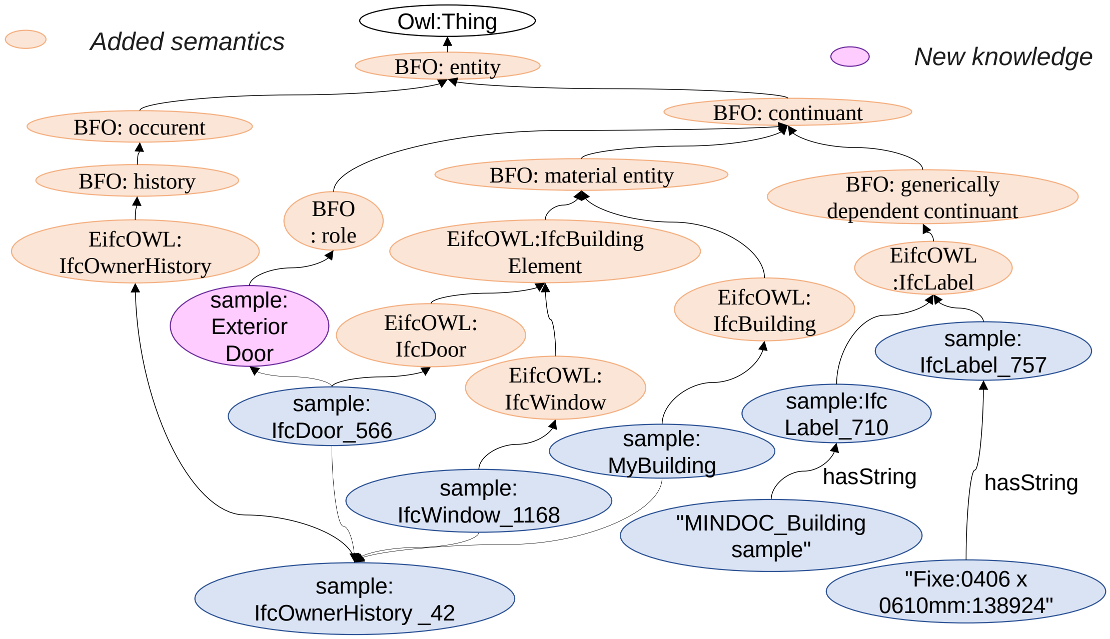


Figure 15. *Instantiation of building BX using the Enhanced ifcOWL ontology*

7. Conclusion

This study is intended to serve as starting point for the exploration and understanding of some issues raised by the ifcOWL ontology. In the case study, ontologies are generated semi-automatically from environmental databases. We have implemented an approach that allows the integration of data while taking their semantics into account, thus demonstrating the value of additional categorization and axioms of EifcOWL compared to ifcOWL. In this way, we have verified that it is possible to combine Semantic Web technologies integrating ontologies to generate better data integration to address interoperability issues in the building industry. We hope that the result will help to foster interoperability between ontologies used in construction, especially in the BIM domain. During the life cycle of a building, the diversity of involved stakeholders attaches different meaning and different purposes to a given object as it evolves through time. We believe that the realism that is incorporated into BFO about such objects as they preserve their identity through time provides a solid basis for improved information exchange in the building domain – and this explains, in turn, our choice of a BFO-based top-level architecture. However, classifying concepts of existing ontologies while try to stick to their definitions and at the same time fit to BFO structure is really a tedious and sensitive task especially when dealing with huge ontologies like the case of ifcOWL (including around 12 thousand classes).

8. Acknowledgement

This work was carried out within the scope of the MINDOC project funded by the Occitanie region of Southern France.

9. References

Abanda, F. H., Tah, J. H., & Keivani, R. (2013). Trends in built environment semantic Web applications: Where are we today? *Expert Syst. Appl., 40*(14), 5563–5577.

Abanda, F., Vidalakis, C., Oti, A., & M, J. H. (2015, Dec). A critical analysis of Building Information Modelling systems used in construction projects. *Adv. Eng. Softw., 90*, 183–201.

Arp, R., Smith, B., & Spear, A. D. (2015). *Building Ontologies with Basic Formal Ontology.* MIT Press.

Beetz, J., Leeuwen, J. v., & Vries, B. d. (2009, Feb). IfcOWL: A case of transforming EXPRESS schemas into ontologies. *Artif. Intell. Eng. Des. Anal. Manuf., 23*(01), 89.

BFO-Based Engineering Ontologies. (2021). Retrieved June 15, 2021, from ncorwiki.buffalo.edu: http://ncorwiki.buffalo.edu/index.php/BFO-Based_Engineering_Ontologies

Bonino, D., & Corno, F. (2008). DogOnt - Ontology Modeling for Intelligent Domotic Environments. In S. S. A. Sheth (Ed.), *in The Semantic Web - ISWC 2008. 5318*, pp. , 790–803. Berlin: Eds. Berlin, Heidelberg: Springer Berlin Heidelberg.

Brodaric, T. H. (2012). *The Void in Hydro Ontology.* Retrieved from https://pdfs.semanticscholar.org/dac5/cdf0e6c3b502b4a9dc79b6708ef45092628e.pdf

Building Data on the Web Working Group Charter. (n.d.). Retrieved May 05, 2018, from w3c-lbd-cg.github.io: https://w3c-lbd-cg.github.io/lbd/charter/

Building Topology Ontology. (n.d.). Retrieved Oct 17, 2018, from w3c-lbd-cg.github.io: https://w3c-lbd-cg.github.io/bot/#zones

buildingSMART - The Home of BIM', buildingSMART. . (n.d.). Retrieved MAY 2018, from buildingSMART: https://www.buildingsmart.org/.

buildingSMART. (2018). *Linked Data Working Group*.

Buttigieg, P. L., Morrison, N., Smith, B., Mungall, C. J., Lewis, S. E., & Consortium, t. E. (2013, Dec). The environment ontology: contextualising biological and biomedical entities. *J. Biomed. Semant., 4*(1), 43.

Digital Construction Ontologies. Version 0.3 (BFO ISO compliant). (2021, April 20.). Retrieved from digitalconstruction.github.io: https://digitalconstruction.github.io/v/0.3/index.html

East E. W. (2007). *Construction Operations Building Information Exchange (COBIE).* Defense Technical Information Center, Fort Belvoir, VA,, ENGINEER RESEARCH AND DEVELOPMENT CENTER CHAMPAIGN IL CONSTRUCTION ENGINEERING RESEARCH LAB.

Ehrig, M. (2006). *Ontology alignment: bridging the semantic gap* (Vol. 4). Springer Science & Business Media.

'Index of /ifc/IFC4/final/html/schema'. (n.d.). Retrieved May 04, 2018, from www.buildingsmart-tech.org: http://www.buildingsmart-tech.org/ifc/IFC4/final/html/schema/

Environment Ontology. (n.d.). Retrieved October 28, 2019, from www.environmentontology.org: http://www.environmentontology.org/home

Farias, M. T., Roxin, A., & Nicolle, C. (2015). COBieOWL, an OWL Ontology Based on COBie Standard. *in On the Move to Meaningful Internet Systems: OTM 2015 Conferences*, (pp. 361–377.).

Farias, T. M., Roxin, A., & Nicolle, C. (2015, Nov). IfcWoD, Semantically Adapting IFC Model Relations into OWL Properties. *ArXiv151103897 Cs*.

Farias, T. M., Roxin, A., & Nicolle, C. (2016). A Federated Approach for Interoperating AEC/FM Ontologies. *LDAC2016 - 4th Linked Data in Architecture and Construction Workshop.* Madrid.

Feeney., A. B. (2002, June). The STEP Modular Architecture. *Journal of Computing and Information Science in Engineering, 2*(2), 132–135.

fundacionctic / parrot — Bitbucket. (n.d.). Retrieved May 05, 2018, from bitbucket.org: https://bitbucket.org/fundacionctic/parrot/wiki/browse/

Gangemi, A., Guarino, N., Masolo, C., & Schneider, A. O. (2002). Sweetening Ontologies with DOLCE. (A. G.-P. Benjamins, Ed.) *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web, 2473*, 166-181.

Grzybek, H., Xu, S., Gulliver, S., & Fillingham, V. (2014, Nov). Considering the Feasibility of Semantic Model Design in the Built-Environment. *Buildings, 4*(4), 849–879.

Horrocks, I., Patel-Schneider, P. F., & Harmelen, F. v. (2003). From SHIQ and RDF to OWL: The Making of a Web Ontology Language. *Journal of web semantics, 1*(1), 31.

IFC Overview summary — Welcome to buildingSMART-Tech.org' . (n.d.). Retrieved May 04, 2018, from www.buildingsmart-tech.org: http://www.buildingsmart-tech.org/specifications/ifc-overview.

IFC4 Documentation. (n.d.). Retrieved May 05, 2018, from www.buildingsmart-tech.org: http://www.buildingsmart-tech.org/ifc/IFC4/Add1/html/

Karray, M. (2012). *Contribution à la spécification et à l'élaboration d'une plateforme de maintenance orientée connaissances.* Besançon: Doctoral dissertation, Université de Franche-Comté.

Karray, M. H., Ameri, F., Hodkiewicz, M., & Louge, T. (2019, Jan). ROMAIN: Towards a BFO compliant reference ontology for industrial maintenance. *Appl. Ontol., 14*(02), 155–177.

Karray, M., Otte, N., Rai, R., Ameri, F., Kulvatunyou, B., Smith, B., . . . al., e. (2020). The Industrial Ontologies Foundry (IOF) perspectives. *I-ESA Workshops.* Tarbes.

Kensek, K. (2015). *Manuel BIM: Théorie et applications.* Eyrolles.

Masmoudi, M., Karray, M. H., Lamine, S. B., Zghal, H. B., & Archimede, B. (2020). MEMOn: modular environmental monitoring ontology to link heterogeneous Earth observed data. *Environmental Modelling & Software, 124, 104581.*

Ontology (Computer Science) - definition in Encyclopedia of Database Systems. (n.d.). Retrieved May 04, 2018, from tomgruber.org: http://tomgruber.org/writing/ontology-definition-2007.htm.

OOPS! - OntOlogy Pitfall Scanner! - Results. (n.d.). Retrieved May 05, 2018, from http://oops.linkeddata.es/response.jsp

Pauwels, P. (2014). *ifcOWL ontology file added for IFC4_ADD1.* W3C community & business groups, Linked Building Data Community Group.

Pauwels, P. (2016). From Linked Building Data to Building Data on the Web. *TPAC2016.*

Pauwels, P. (2018). *EXPRESStoOWL is a set of reusable Java components that allows to parse EXPRESS files and convert them into OWL ontologies in the context of the Industry Foundation Classes (IFC).* pipauwel/EXPRESS.

Pauwels, P., & Terkaj, W. (2017). A Method to generate a Modular ifcOWL Ontology. *8th Int. Workshop Form. Ontol. Meet Ind*, (p. 13).

Pauwels, P., & Van Deursen, D. (2012). IFC-to-RDF: adaptation, aggregation and enrichment. *First Int. Workshop Linked Data Archit. Constr.*, (p. 65).

Pauwels, P., Krijnen, T., Terkaj, W., & Beetz, J. (2017). Enhancing the ifcOWL ontology with an alternative representation for geometric data. *Autom. Constr., 80*, 77–94,.

Pauwels, P., Van Deursen, D., Verstraeten, R., De Roo, J., De Meyer, R., Van de Walle, R., & Van Campenhout, J. (2011, Aug). A semantic rule checking environment for building performance checking'. *Autom. Constr., 20*(5), 506–518.

Pauwels, P., Zhang, S., & Lee, Y.-C. (2017, Jan). Semantic web technologies in AEC industry: A literature overview. *Autom. Constr., 73*, 145–165.

Rasmussen, M. H., Pauwels, P., Hviid, C. A., & Karlshoj, J. (2017). Proposing a central aec ontology that allows for domain specific extensions. *Conference on Computing in Construction, 1*, pp. 237–244.

Roxin, A., & Pauwels, P. (2016). SimpleBIM: From full ifcOWL graphs to simplified building graphs. *11th Eur. Conf. Prod. Process Model.* (pp. Pp 11-18, p. 8). ECPPM.

Skill Modelling for Digital Factories | SpringerLink. (n.d.). Retrieved Oct 28, 2019, from https://link.springer.com/chapter/10.1007/978-3-319-99707-0_40

Smith, B. (2006). Against Idiosyncrasy in Ontology Development. *Front. Artif. Intell. Appl.*, 12.

Smith, B. (2008, July 16). Ontology (Science). *Nat. Preced.*

Smith, B. (2019). Ontology as Product-Service System: A Study of GO, BFO and DOLCE. *Proceedings of the 10th International Conference on Biomedical Ontology (ICBO).* Buffalo, NY: University at Buffalo.

Smith, B., & Ceusters, W. (2015). Aboutness: Towards Foundations for the Information Artifact Ontology. *Sixth International Conference on Biomedical Ontology (ICBO). 1515*, pp. 1–5. CEUR.

Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., & ... & Rosse, C. (2005). Relations in biomedical ontologies. *Genome biology, 6(5)*, 1-15.

Sojić, A., & Terkaj, W. (2015, Sep). Ontology-based representation of IFC EXPRESS rules: An enhancement of the ifcOWL ontology. *Autom. Constr., 57*, 188–201.

Staab, S., & R. Studer, E. (2009). *Handbook on Ontologies.* Berlin, Germany: Heidelberg: Springer Berlin Heidelberg.

Tchouanguem Djuedja, J., Abanda, F., Kamsu-Foguem, B., Pauwels, P., Magniont, C., & Karray, M. (2021, February). An integrated Linked Building Data system: AEC industry case. *Advances in Engineering Software*, *152*, 102930.

Tchouanguem, D. J., Karray, M. H., Foguem, B. K., Magniont, C., & Abanda, F. H. (2019). Interoperability challenges in building information modelling (BIM). *Enterprise interoperability VIII* (pp. 275-282.). Berlin: Springer.

Terkaj, W., & Pauwels, P. (2016). EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology. *Automation in Construction*, 100-133.

TERKAJ, W., SCHNEIDER, G. F., & PAUWELS, P. (2017). Reusing Domain Ontologies in Linked Building Data: the Case of Building Automation and Control. *8th International Workshop on Formal Ontologies Meet Industry*.

Torma, S., & Zheng, Y. (2021, April 20). *Digital Construction Ontologies (DICO)*. Retrieved from digitalconstruction.github.io: https://digitalconstruction.github.io/index.html

W3C. (2018). *Linked Building Data Community Group*. (W. c. groups, Producer)

Wallace, E., Kiritsis, D., Smith, B., & Will, C. (2018). The Industrial Ontologies Foundry Proof-of-Concept Project. *IFIP International Conference on Advances in Production Management Systems*, *536*, pp. 402-409. New York.

Wix, J. I. (2015). What is IFC? *Proceedings of IFC/IFD/IDM Workshop.*