Periodica Polytechnica Architecture, 56(2–3), pp. 41–79, 2025

An Ontology-based Approach to Solving the Digital Representation Problem in Architectural Engineering

Dániel Bakonyi[1*], Gergely Dobszay[1]

[1] Department of Building Constructions, Faculty of Architecture, Budapest University of Technology and Economics, Műegyetem rkp. 3., H-1111 Budapest, Hungary
[*] Corresponding author, e-mail: dbakonyi@epsz.bme.hu

Received: 11 March 2025, Accepted: 30 July 2025, Published online: 01 September 2025

Abstract

This article addresses the need for computer-aided design systems in Architectural Engineering (AE) design for building construction, an essential aspect of every building's design that is currently underserved by existing software solutions. First, we analyse the various challenges facing the development of better tools and propose that a fundamental problem of representation lies behind most of them. Next, we suggest a desideratum for representing building constructions digitally in our tools, and we argue that a formal ontology is best suited for the task. Numerous systems have been proposed for developing ontologies, many of which rely on a domain-independent upper ontology. This common upper layer supports the creation of more specific ontologies in an interoperable and mutually supportive way. The BFO upper ontology and the corresponding methodology are briefly introduced, followed by a review of many existing ontologies relevant to AE. Lastly, we introduce a prototype proto-ontology for describing building constructions and their related phenomena: the building construction ontology. We present the most important terms and modules of BCON and then demonstrate its use through a small case study of a relatively simple construction detail. Finally, we highlight the many possible applications of such a representational system.

Keywords

Architectural Engineering, building construction, ontology, Building Information Modelling, linked data, Basic Formal Ontology, Semantic Web

1 Introduction

The adoption of digital modelling technologies in the Architecture, Engineering, and Construction (AEC) industry holds promise for numerous auxiliary benefits and applications. Although significant progress was made in Building Information Modelling (BIM), this was not the case across all AEC subdomains. Even for the subdomains with the most progress (architectural design and construction management), some of the most ambitious applications remain unrealised, as they turned out to be more complex than first anticipated (Sacks et al., 2020). Gholizadeh et al. (2018) found that only 7 of the 14 BIM functions they investigated were widely adopted. In the specific domain of the Architectural Engineering (AE) design of building constructions (henceforth AE) and other subdomains, there is a general lack of studies on digital technology adoption (Ferron and Turkan, 2019), and progress is slow.

AE engineers designing building constructions are responsible for detailing many aspects of buildings that are not directly contributed by other specialised fields while constantly liaising regarding the building's aesthetic, functional, physical, and material aspects. AE is not architectural design, structural engineering, building physics, Heating Ventilation and Air Conditioning (HVAC), or electrical design, but it is concerned with and related to all of these. Some tasks typically performed by AE engineers include waterproofing, thermal insulation, façade cladding design, and detailing, among others. The fact that such a circumspect definition is needed here illustrates that this field is less well-structured and recognised than others in the building industry, even though it is an inseparable part of architectural and civil engineering work. AE is often done by professionals with different backgrounds depending on the country and even the type of project: by architects with sufficient engineering background in some countries and in smaller projects, either as part of the general architectural design or as a separate contribution,

Cite this article as: Bakonyi, D., Dobszay, G. (2025) "An Ontology-based Approach to Solving the Digital Representation Problem in Architectural Engineering", Periodica Polytechnica Architecture, 56(2–3), pp. 41–79. https://doi.org/10.3311/PPar.40408

or by specialized engineers in other countries or bigger and more complicated projects. There are many specialised fields within AE, too, from waterproofing to façade design.

The design of building constructions has not always been a distinct engineering discipline; instead, it has been part of general design, based primarily on experience and intuition gained through trial and error. However, experimentation with constructional solutions happened over long timescales as the development of architectural ideas and constructional techniques was relatively slow. A distinct engineering discipline emerged when change started to occur more rapidly. Architectural trends began to yield more unique and complex designs as technical requirements increased at an ever-faster pace. This necessitated solutions with much higher element counts and specialised materials for different functions and complex interactions, while safety margins were always reduced to a necessary minimum for cost-effectiveness. All of this is facilitated by rapid advances in material sciences and construction techniques. The main characteristic of AE today is, therefore, change; in former times, the same basic design could be reused over and over. Nowadays, almost all buildings require at least some unique and innovative solutions.

AE design has two constant trains of thought: on the one hand, constructions must be designed to fulfil specific functions, while on the other, environmental and other impacts must not result in the failure of these constructions during their required lifespan. To design new types of constructions, we cannot rely solely on experience; we must understand why specific constructions work or fail, what causes their potential failure, and whether we can modify previous solutions to meet new and unique needs. Like all engineering designs, AE must model the relevant processes and interactions of the constructions and their environment. Possible solutions are evaluated, and their failure modes are analysed and assessed to reach decisions. Unlike some other fields, however, AE must cope with a seemingly unlimited number of interacting effects and processes, typically studied by many different fields (mechanics, building physics, acoustics, and building chemistry). Many of these are very difficult to model quantitatively, even independently, and their interactive nature makes it challenging to isolate any part of the whole building for modelling in the first place. Most modelling, therefore, is conceptual and qualitative, taking place in the designer's head, where they must rely on their professional expertise to reduce the problem to its most relevant aspects and thus make it manageable. AE engineers must also concern themselves with all parts of the building (unlike, for example, a structural engineer for whom much of the building is "just" forces acting on the structural members), and AE design must coordinate with almost all other engineering disciplines, it is therefore by its very nature more "holistic".

In the digital tools used by AE professionals, the functionalities most sought after (in addition to the ones shared by all fields) therefore revolve around the representation of the domain-specific information, data gathering, model exchange and cooperation with the other designers, utility functions for creating the drawings and documentations specific to AE, performance analysis and decision support.

However, the most interesting application of digital technologies in design lies beyond questions of representation and simulation when we reach the realm of Artificial Intelligence (AI) and actual computer-aided design: decision support and expert systems, design checking and perhaps design generation. While research is ongoing, AI in the AEC industry has proven to be especially more difficult than first anticipated (Sacks et al., 2020). Symbolic AI systems suffered from the need for excessive manual expert input and the challenges of translating natural language knowledge into logical languages. At the same time, statistical AI research requires training data that is difficult to obtain when relevant information is presented in heterogeneous formats with ambiguous semantics (Pauwels et al., 2011).

AE is a complex but comparatively small subdomain of the AEC industry, which has not seen much dedicated research in any of these fields; consequently, it has few dedicated digital tools beyond the general Computer Aided Design (CAD) and BIM applications. As a result, AE design is still, for the most part, unaided by digital technologies beyond simple "dumb" manually created models or, most of the time, only 2D drawings, whether on actual paper or in a CAD / Computer Aided Architectural Design (CAAD) / BIM application. However, there is potential for significant development, making it worthwhile to investigate further.

2 Problem statement for computer-aided AE design

In Section 2, we will summarise the major challenges we must overcome in creating more AE-related design software. Many of these, of course, are not exclusive to AE.

2.1 Terminology

There is a general lack of a unified terminology. As stated at the beginning of this article, it is sometimes challenging to name the field unambiguously, much less the numerous terms used by its practitioners. Even within the same language zone, there can be differences. These can amount to more than just a translation issue, as different user groups can have significantly different conceptualisations for key terms. Take, for example, the notion of a "watertight concrete" structure. For different groups of users, based on the language region and/or applicable standards and regulations, this can mean either some material properties of the concrete or the classification of some end result by the amount of water that is present in the internal (protected) side of a structure (which will also depend on many other conditions like hydraulic pressure, internal volume temperature and ventilation, or structural detailing). Terminological inconsistency makes information sharing at scale and with automated tools very difficult as a human expert is required to map between the various overlapping conceptualisations.

2.2 Design data

For multiple reasons, there is a general lack of good and reliable design data. This is partially the result of the aforementioned lack of terminology. As pointed out in Wang et al. (2010), it is also a problem that information is published in very heterogeneous, hard-to-search formats and sometimes in unpredictable places. It can also be highly contextual what data will be needed by the designers in the first place. The authors emphasise the need for a representation of this context that extends beyond mere keywords.

Next, we describe some areas where the lack of data is apparent.

2.2.1 Material, building product and building system performance data

The problem here is many-faceted. Some data is guaranteed to be published as it is needed for calculations mandated by regulation (e.g., material thermal conductivity for calculating thermal transmittance). However, where there is no such mandated calculation, manufacturers have a general disincentive to provide information (or at least a lack of incentive). Most measurements are expensive and time-consuming (like the measurement of detailed hygrothermal material properties), and some values are treated as quasi trade secrets (e.g., the density of mineral wool thermal insulation is rarely published).

Most of the relevant performance metrics are also not constant but are functions of one or more other parameters (like the temperature and moisture dependence of thermal conductivity). While this is widely known in the research literature and even reflected in certain calculation standards, getting such complex material data outside of research papers or "one size fits all" correction functions in standards is very rare.

It is also the case that there are no generally accepted measurement methods for many of the relevant characteristics considered by AE designers when making decisions (e.g., the lateral water creep of specific waterproofing systems (Dobszay et al., 2019)).

Some regulations reflect this situation; for example, the European Construction Product Regulation (CPR) (Council of the European Union, 2011) requires the publication of at least one technical property for a given product. This is understandable from the point of view that it would probably be impossible to define the kinds of data required objectively. It places designers in a difficult position, as detailed product specifications are needed from them, but data poverty makes this nearly intractable. As a result, practice generally tends to specify materials by reference to specific products rather than their performance characteristics. This suits manufacturers very well if their product is used as a reference, as any substitution would have to prove equivalence, which can be difficult. Additionally, it benefits designers because they do not have to identify and quantify all relevant metrics in complex cases. It is a system based mainly on previous experience, which is very good for simple cases but gives little support when a new situation is encountered.

2.2.2 Regulatory requirements

Regulatory requirements are numerous and constantly changing. They typically have several interrelated levels (ranging from laws and regulations to standards and guidelines) and are found in various locations in diverse formats. They also suffer from the lack of a unified terminology in their formulations. As pointed out by Dimyadi et al. (2015), there have been successful attempts to create application-specific logic-based systems to aid in the regulatory compliance of buildings by hard-coding rules; however, these have, for the most part, proved inflexible and unfeasible to maintain in the long run. The authors proposed to use more general-purpose and open tools instead.

2.2.3 Design loads

The accounting system for internal and external environmental loads on building constructions is not nearly as well-formulated as the system for loads in structural design. While regulators and developers demand structural solutions that withstand the expected impacts, these are, if quantified at all, difficult to find and are expressed in heterogeneous formats, not in a single unified vocabulary.

2.2.4 Expert opinions

While contemporary AE must constantly create new clean-sheet designs based on first principles, the actual, real-life empirical evaluation of earlier designs, in the form of expert opinions and studies, especially regarding defects and defect causes, is an invaluable resource. In Lee et al. (2016), the authors note that the material published is often found in hard-to-access locations, expressed in heterogeneous formats, and not indexed or tagged using a single unified vocabulary.

2.3 Data schemas

BIM data schemas have undergone significant development, and the widespread adoption of open BIM schemas has brought additional benefits, including easier cooperation through specific software and engineering subdomain boundaries. The most notable of these is Industry Foundation Classes (IFC) (ISO, 2018), an open international standard and a vendor-neutral open BIM framework for the digital description of the AEC industry and its designs, created and maintained by the buildingSMART Organisation. Unfortunately, there is relatively little AE relevant information that is easily and directly represented in BIM software, as the development of both closed (proprietary) and open BIM software and schemas lay very little focus on AE design, and much AE domain-specific data is yet to find a proper place in them (Tchouanguem et al., 2021). As a result, much of the data is added as simple hand annotations of drawings and in separate text files. BIM data schemas have an understandably limited number of properties directly associated with certain element types (like walls, slabs, and windows). It is, of course, possible to represent much more, for example, with the help of custom properties and property sets; however, that still requires a widely known and used, well-defined, and unambiguous schema, independent of the model itself, to be truly useful. The Building Smart Data Dictionary system (Building Smart Alliance, online) for IFC is an example that enables the creation of extensions for element types, properties, and property sets and their publication online. While this can be very useful, questions remain: who and by what mechanism is to create and update such extensions, how is consensus achieved, how do such small schema extensions work together, how are designers to know what schema to use and when, and what to do with properties that still lack a proper standardised terminology in the first place? Most design models and drawings are simply hand annotated and must be supplemented with lengthy written text (technical specifications). Lots of time is spent creating these.

Similar limitations of IFC are documented in Cutarelli (2024) for modelling Historical Architecture. The author analyses various approaches chosen in real projects and demonstrates the limitations of the IFC schema in terms of providing an adequate semantic structure to handle the specialised information, uncertainty, and heterogeneity of structures encountered in historical buildings. These seem to fall outside the primary design goals of the IFC schema and the connected software development, requiring considerable manual effort to be fitted into an IFC file. This often negates most or all of the benefits that open BIM is proposed to offer.

To go into more detail, there are some general problems with most contemporary approaches to BIM modelling concerning AE:

- The representation is usually ambiguous: there are many ways to model the same thing, which offers flexibility, but as a result, the modelling quality is almost entirely up to the user (Kitamura, 2006).
- This makes it very difficult to develop algorithms to process the resulting data. Despite the advances in open BIM, the translations between different formats and between BIM formats and simulation software are lossy and ambiguous; therefore, the interoperability of various tools, although much improved, is still far from being fully resolved.
- Topological information is usually completely lacking. AE design is all about topology at a level of granularity (connectedness of individual elements down to single nuts and bolts, strips of waterproofing membrane, etc.) way beyond what is usually represented in BIM models (which are typically restricted to connections between buildings, floors, spaces and elements in them). Topological relations between all constructional components and boundaries are essential for the design. These are typically manually documented in layer orders and separate construction details, as BIM software does not provide adequate solutions. Modelling every component in a 3D model, though possible (usually with enormous effort), still does not result in explicit connectedness and queryable topological information.

  Jabi et al. (2018) point out that most CAD and BIM systems utilise manifold topology for various geometrical entities; however, in reality, the connectedness of these elements forms non-manifold topologies. These hold much more information and enable, or at least make, particular tasks much easier than a simple collection of individual manifold objects. Jabi et al. (2018) created an open-source non-manifold topology tool, Topologic, to perform such tasks (energy and structural analysis, pathfinding), especially in the early design stage.

- There is a general lack of standardised data schemas for a whole host of information needed to describe building constructions in the AE domain. Some types of information that must find a proper home in our modelling include:
  - Complex material properties.
  - Connection information: besides the aforementioned topological information, there is usually no good way to represent the type and properties of connections either. Connections convey structural/mechanical, hygrothermal, acoustic, and various other types of information, including performance requirements, metrics, and technological details.
  - Design intent and function: all AE design methods rely heavily on function, which must either be inferable or directly represented for any model to be useful for almost any type of automated processing.
  - Processes and functioning: modelling building constructions involves not only describing their static material makeup but also modelling their processes. While actual quantitative physical modelling is best done by other programs, BIM models should facilitate the unification (inputs and outputs) of all relevant information in one place. Also, some processes have not (yet) established computable physical models and still need to be documented. Many processes also have many participant objects, which are difficult to capture in contemporary BIM schemas.
  - Additional relationships between entities, besides parthood and topological connectedness. Constructional elements can participate in numerous relationships with one another and other entities, such as functions and processes, and various kinds of dependence.
  - The easy identification of different subsets of constructional components to assign information to them (e.g., all the other interconnected parts of the envelope responsible for a specific function).

When no other, more straightforward way is provided, data must be added by hand annotations and written in separate accompanying documents.

2.4 Geometric representation in 3D models vs. printed plans

Most architectural drawings are abstractions of real geometry. We can think of the difference between an actual horizontal section of a 3D building model and a floorplan, the latter of which has 2.5D elements representing objects and features both below and above the theoretical section plane. Also, a floor plan's section plane is usually not continuous, as it jumps up and down to intersect different features (like windows) at various heights. In general, most 2D technical drawings are not simple projections of 3D entities; instead, they employ various tools to make them intelligible on 2D paper. The standards of representation were developed long before the advent of CAD and are based not on true geometry but rather on readability for humans.

Abstraction is especially relevant for AE drawings. Construction details deal with many components at different scales and complexity, see Figs. 1 and 2. While it is certainly possible to represent all elements in a single 3D model, it is very time consuming to do so, therefore much of the AE relevant elements, and geometric detailing are either simplified or absent. But even if this were to change, generating usable 2D plots of the details from 3D models that the industry expects is far from trivial. As a result, most drawings that reach the construction site, especially in case of construction details, are 2D vectorgraphics drawn and annotated mostly by hand. It is an interesting question if such 2D drawings could ever be entirely substituted by digital devices and smart views of 3D models on the constructions site. Another unfortunate fact is that the 2D vector graphic drawings we create to add informative annotations to, with much effort, are then mostly unsuitable for automated processing, for example, by a simulation tool. This is due to the very abstractions we made to make them human-readable (reduced dimensionality, separation of lines, and distortion of dimensions to make small elements readable). A representation is needed that can serve both functions in a more automated way: computer and human readability.

2.5 Geometric errors

General-purpose BIM software usually lacks any mechanism to enforce or at least strongly encourage and aid

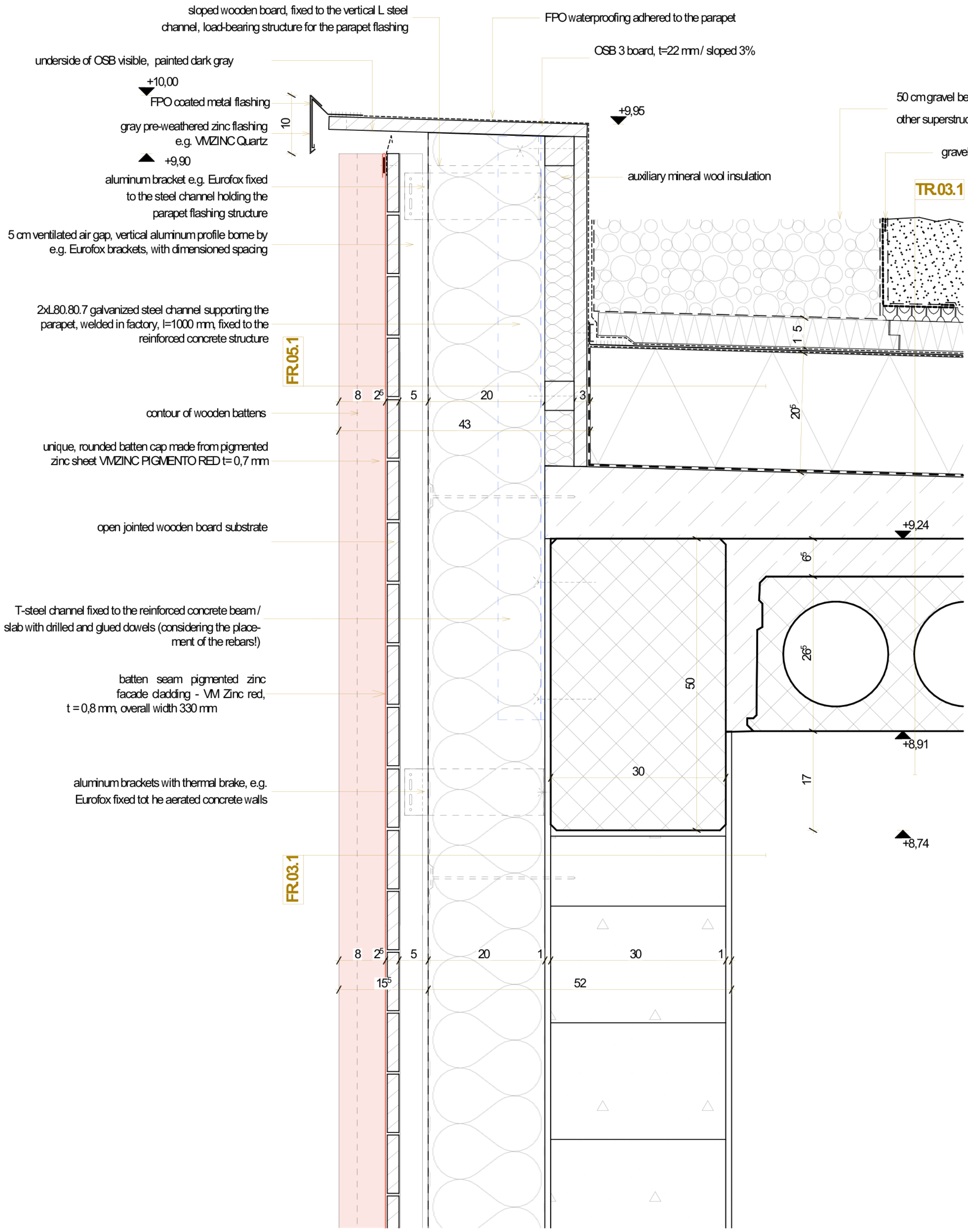

Fig. 1 Typical hand drawn and annotated construction detail drawing. It contains elements not found in the BIM model (dowels, brackets, screws, the exact size, and layout of brick layers in the masonry), and it uses its own symbology (dash lines representing hidden edges and screws, for example) and distortions for better readability

error-free geometry creation (to avoid geometries that cannot exist in real constructions) directly in the modelling software (besides collision detection tools). The resulting models are often "good enough" for many architectural purposes but not so for analysis. The kind of BIM models AE engineers receive from architects usually contain too many geometric errors (voids, unintended overlaps, orphaned elements hanging in the void, for example) to perform engineering analysis without a manual rebuild, see Fig. 3. It seems unfeasible to rely simply on user input

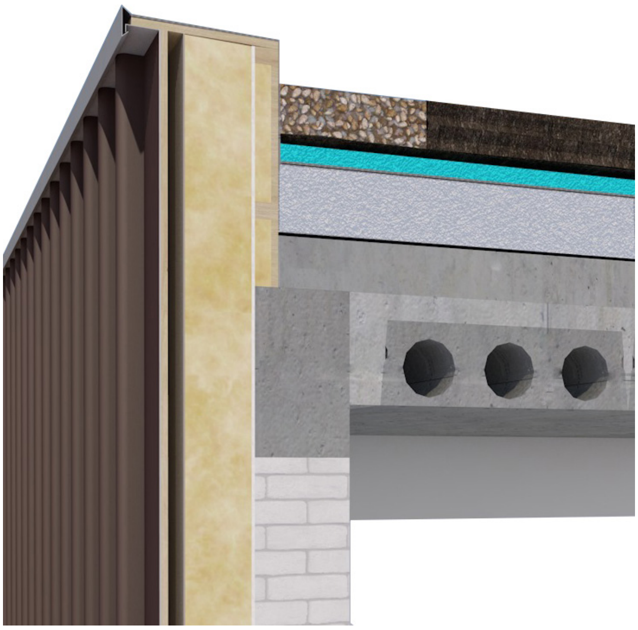

Fig. 2 The same detail as in Fig. 1 in the 3D BIM model. Many elements are missing (dowels, brackets, screws). Some geometry has been Periodica Polytechnica Architecture simplified for modelling convenience (the top of the wall is flat instead of sloped, the slope of the waterproofing is neglected, and a simple texture represents the masonry), and the hand-annotated information is missing

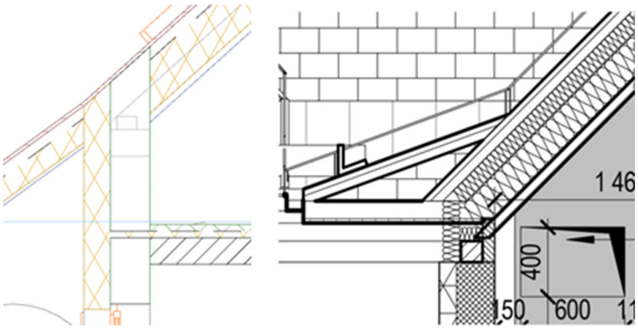

Fig. 3 Some typical examples from "bad" BIM models AE engineers often receive even in the relatively late stages of a project. While many errors can be eliminated with higher fidelity modelling and BIM quality control tools, there is an inherent tension between the different usages. Such details are perfectly usable for creating high-quality renders, and many of the drawings architects produce; however, they are entirely unsuitable for engineering analysis

in an entirely free geometry creation environment to result in error-free models (especially in situations where most BIM modelling and the analysis is done by different professionals often working for different firms).

Specialised BIM quality checking solutions exist, mostly in separate software such as the Solibri Model checker (Solibri Inc., computer program) and the Tekla Model Checker suite (Trimble Inc., computer program), which can handle more complex and parametrised geometric checks. However, model checking is still separate from modelling. Modelling standards and prescriptions for how to model specific constructions can also help, but they still open the door to human error.

Some of the errors can be traced back to the fact that most BIM software only deals with manifold topological entities, or rather a collection of such entities without explicit connections (see Section 2.3). When this connectedness information is needed for specific applications, it must be inferred and reintroduced into the model. We can find many works dealing with this issue, such as the papers of Lilis et al. (2015) and Giannakis et al. (2019) on the automated creation of building energy models from BIM models. Much of the effort is focused on recreating connections and detecting and correcting topological errors that would not necessarily be present if BIM modelling were not so far removed from non-manifold topology and topological connections in the first place.

A more ideal approach would perhaps be a CAD system with explicit connectedness information, as well as immediate feedback about common types of geometric errors, and even specialised geometry creation functions that at least restrict the types of possible errors.

Another source of geometric error is the construction process itself: the as-built construction will inevitably differ from the plans. As pointed out in Talebi et al. (2019), most BIM software does not handle building tolerances in a detailed way in their geometric representation, unlike specialised software in the manufacturing industry. This is unfortunate, as tolerance errors are amongst the most common building mistakes and can seriously impact the performance of many construction systems. Possible performance decreases due to tolerance issues could be investigated with physics-based modelling, but if this data is not represented in the 3D models, this would require the manual rebuild or "correction" of the idealised geometry. For example, an uneven wall surface can cause the individual boards of the thermal insulation layer built on top of it to separate, forming gaps that can lead to air circulation between or around the boards, drastically reducing their thermal performance.

2.6 Parametric design

Design iteration and parametric design are complex problems for most fields in the AEC industry. Much manual work is needed to update models after design changes, especially in later stages. This disincentivises optimisation in AE, especially as much of the work is already spent reacting to design changes made by other designers (architects, structural engineers). There is a pressing need for a more automated and parameterised model creation workflow to address this problem (Stals et al., 2021). As pointed out in Gürsel Dino (2012), parametric design requires the designer to model not only the end result but also the conceptual framework that leads to it, forcing the designer to clarify design intention early on. For AE, this could be easier than for architects, as AE engineers must always keep the building work order of their designs in mind. It is also challenging for the BIM modelling schema to provide the associated expressive semantics.

2.7 Query

The flip side of storing all the relevant information (geometry, topology, and all other kinds of data) is the ability to easily retrieve said information, preferably with some declarative query language (a language to express what we want to access while leaving the implementation details of the actual operation to a suitable program). As pointed out by Zhang et al. (2018), the most widely used open BIM schema, IFC, was designed for the documentation and sharing of design data, rather than for its retrieval *via* query mechanisms. It is also challenging to integrate data from various sources. There can, of course, be separate island solutions for different fields and databases, but the interesting inference tasks require a unified picture of all the data about the design in a single environment.

2.8 Interoperability issues

Cooperation with other disciplines can consume a significant portion of an AE engineer's time. The widespread adoption of open BIM standards, most notably IFC, has significantly reduced the issue of incompatible data silos and software, but problems persist. Open standards, such as IFC, are typically used as a data exchange format between proprietary models; however, the translation process is often lossy (Paskaleva et al., 2021; Pauwels et al., 2011), partly because IFC is redundant and ambiguous (Venugopal et al., 2015). As pointed out by Steel et al. (2012), this is at least partly because IFC aims to cover a very large domain and different contexts, and must, therefore, allow for different modelling styles that are difficult to translate automatically.

2.9 Lack of dedicated tools in general

The question of performance analysis of building constructions *via* simulation tools is very much related to the difficulties of data exchange. The interdisciplinary nature of AE design requires many tools (from acoustics to mechanical design and building energy and fire safety) with their own modelling styles, entities and data requirements. Without easy ways to translate between such models and the main working BIM model, each analysis requires considerable manual effort and additional systems to track the various simulation-related entities not included in the main model (Fernald et al., 2018; Habibi, 2017).

There are specialised simulation software for thermal bridges, hygrothermal, building energy, acoustic, or structural analysis, but these are only some subtasks of AE design. There is no software that provides a specialized utility for larger portions of the AE design work. Contrast this with, for example, structural engineering, where much of the calculation and documentation (at least for common structures) can be done in a much more unified environment (see Autodesk Revit for Structural Engineering (Autodesk, computer program), AXISVM (Software Development Company, computer program), Consteel (ConSteel Solutions Ltd., computer program), FEM-Design (StruSoft, computer program), etc.). AE designers use much of the same general-purpose modelling software as most architects (CAD, CAAD, and BIM). BIM software can be useful in glueing separate analysis models together and with the documentation. Nevertheless, much of the work is still manual. This is because the geometric problems already mentioned in Section 2.5 are still not completely solved. In general, it is fair to say that the dominant use of BIM software for AE is for documentation (sometimes of plans created at least partially by other means) rather than design itself.

2.10 Implicit need for background knowledge

Understanding any engineering model relies on considerable background knowledge of the type of modelling being done. As highlighted in Kitamura (2006), the modelling type depends on many factors like the terminology used, the abstraction and granularity level, the underlying domain and its domain theory (physical theory, if any) and even the kind of engineering task at hand. Human readers of plans and models are taught this and many other domain-specific pieces of information, which they use to make sense of construction drawings. However, most of the modelling in AE is conceptual in nature and takes place in the designer's head, based on extensive knowledge and experience that is hard-gained and very difficult to transfer effectively to even colleagues, much less into computer systems. Computer algorithms cannot rely on information they do not have direct access to or cannot derive on their own.

2.11 Automated model checking

It would be desirable to have tools that can analyse our designs and detect at least some potential errors. There is great unrealised potential even in simple geometric error and rule-checking BIM quality control software like Solibri or Tekla (Solibri Inc., computer program; Trimble Inc., computer program). For AE engineers, a large part of their job is to find errors and contradictions in plans as the choice between different design alternatives often comes down to a handful of local problems that preclude one approach but are easily solved by another. Finding these situations is tedious and not trivial even for seasoned practitioners, and it relies mainly on sheer experience. In large and complicated designs, it is all too easy to miss relevant situations, especially since the entire building's 3D model and the AE design is often created by separate individuals or even companies. An in-depth analysis of the business case for automated rule-checking systems by the buildingSMART Organisation (El-Diraby, 2019) also highlights the potential benefits and challenges in view of the numerous technical challenges.

Automated model checking is an inherently complicated problem to solve. There is the problem of implicit knowledge (see Section 2.9). Also, many rules that human designers use with relative ease are very hard to formulate in ways that lend themselves to algorithmic processing (Holnapy and Rédey, 1993). In Solihin et al. (2020), the authors note that, in addition to representing the rules in a computable manner, extracting the necessary information from the BIM models themselves is also a considerable challenge. Only a portion of the relevant information is represented directly, and much of it must be inferred by the system, which is not a strong suite of BIM schemas, such as IFC, as discussed in Section 2.7. The authors of Dimyadi et al. (2015) also point out the need for engineering analysis solutions in rule-checking systems, as many of the requirements are performance requirements that must be calculated first.

2.12 Basis for AI research and development

Model checking leads us to the broader question of AI. The drive to develop AI-enabled tools to aid designers is prevalent in all fields of the AEC industry. The general lack of suitable standard terminology and data models that can hold relevant information is a significant problem for anyone trying to apply symbolic approaches, and the connected lack of sufficient training sets for statistical AI approaches (Sacks et al., 2020).

3 The representation problem and the need for an AE ontology

As we saw in Section 2, the problem is many-faceted, but it has a fundamental core: how do we express, structure, store, gather, maintain, and query all the necessary information? It is fundamental in the sense that solving the other problems will rely heavily on our answer to them (for example, how we can build statistical AI depends significantly on the type of data we have). Much of the research and development effort into BIM, in general, focuses on developing standardised data schemas. However, we can state the problem in an altogether domain-independent way and realise that other fields have similar issues. In this way, it becomes clearer that we can take advantage of previous research and existing technologies.

The generalised problem is to create a kind of representation of a complex domain, which:

- is compatible with other overlapping and neighbouring domains of interest. Data for AE designers must be compatible with the data of structural and acoustic designers, for example, as well as with data from more distant fields, such as materials science or meteorology. Data interoperability is necessary for information sharing, collaboration with other designers and software interoperability. The lack of interoperability creates data silos that require considerable manual effort to combine or transfer any of their content.
- is unambiguous. Ambiguous representations are particularly problematic for resolving interoperability issues and algorithmic processing of designs.
- has sufficient expressive power.
- is built on a clear, transparent framework that is easily (enough) extendable and maintainable. Every field is constantly changing and progressing. A good representation system must keep up without constantly increasing complexity and without requiring the rewriting of existing codebases, thus endangering future progress.
- can enable the efficient and easy querying of the data with declarative query languages.
- can preferably enable automated inference, not necessarily to directly predict processes in the ways of physics-based simulations, but to infer implicit information from what is directly represented or find contradictions in the data or the representation system itself during development.

The part of the problem that is specific to AE is:

- the domain entities to be represented. The specific nuances of AE design refer to the types of entities, including their qualities and processes, that we want to describe. This does not necessarily dictate the data structure directly.
- as a part of this effort, there must be a way to create a standardised general terminology for AE.
- ultimately, we must find a type of representation that provides a solid foundation for developing domain-specific algorithms, AI research, and other applications.

Can BIM schemas, in their current or expected near-future forms, or with possible extensions, serve as a basis for this? Our thesis is that they are not. BIM standards are data schemas which, though very useful for certain tasks, lack certain features, such as any inbuilt mechanism to guarantee or even encourage unambiguous representation, declarative query mechanisms (at least in their original form), inference rules, easy extensibility, and modularity, but especially interoperability with domains not directly covered in them. Instead, we propose that the problem is best solved with the help of formal ontology.

Formal languages have clear, explicit syntax and semantics. The syntax or grammar is the set of rules specifying the well-formed sentences in the language, which are not enumerable but infinite. Nevertheless, syntax must be unambiguous. Semantics is the set of rules that unambiguously define the meaning of sentences in computable languages, allowing for the necessary computability and inference. Formal languages are the opposite of natural languages, which are much more complicated, much less well-defined, ambiguous, and very context-dependent, not just in their grammar but also in their meaning. Data schemas, like BIM schemas, are too defined and constrained by formal languages, like the EXPRESS (ISO, 2004) language for IFC. As a result, BIM schemas have clear syntax, and clearly, it is quite possible to create software that can read and write these schemas without error. However, their semantics regarding the actual world of buildings and constructions is much more unclear. Open BIM data schemas like IFC have definitions and descriptions that aim to connect their types, classes, and relations in their referents, but these are sometimes quite confusing (see Venugopal et al. (2015)), and they lack a basis in formal logic. The representation is ambiguous and has a very complex structure. The extension of the schemas is very laborious, and therefore, they lack expressivity. Formal ontology is a better system for the task.

The word ontology originates from philosophy, a branch of metaphysics concerned with being or existence and how entities can be grouped into basic categories. A very early example is the categories of Aristotle (see Fig. 4). However, the term ontology came much later, as an attempt

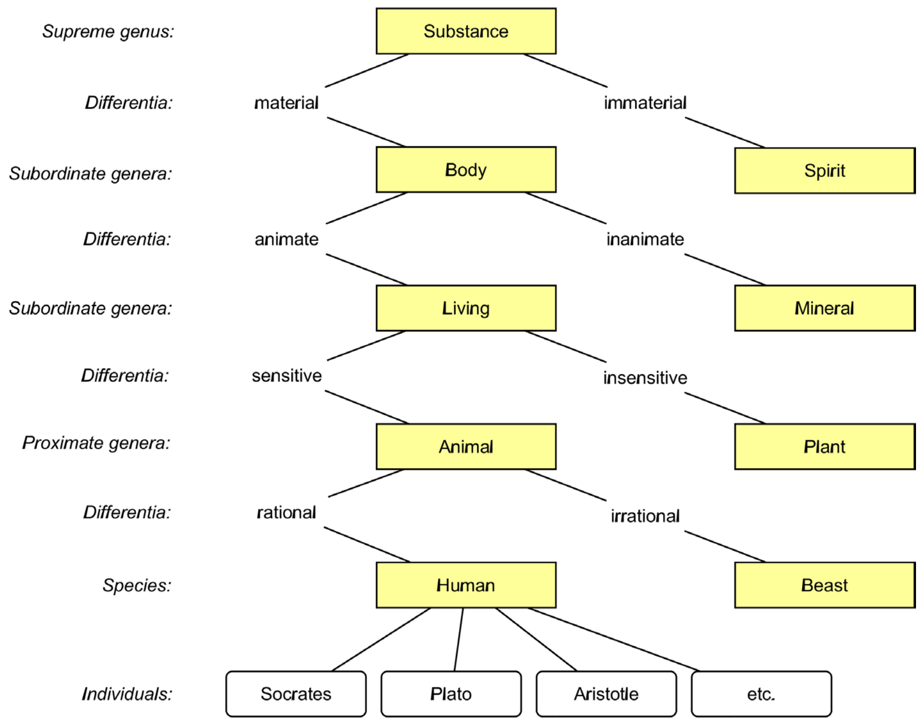

Fig. 4 The Porphyrian Tree: a visual representation of Aristotle's categories of Substance in a tree structure by the later philosopher Porphyry of Tyre

to categorise every entity we might encounter, at least at an elementary level (e.g., substances, accident qualities, processes, material, or immaterial things). The subdivisions are based on Aristotelian definitions, which consist of a genus and differentia. Ontology enables knowledge sharing by providing a common and definitive classification of entities.

Ontology as a method is experiencing a renaissance right now, as it has gained the interest of non-philosophers in categorising and describing entities within specific domains of interest. This stems from two major sources: the data-driven nature of much modern science, which requires a unified way to represent findings (Schulze-Kremer and Smith, 2005), and the enormous wealth of information in separate and incompatible databases and schemas that are very difficult to connect. A significant example of the latter is the internet itself, which also needs a structure for its wealth of knowledge to be more easily exploited. A problem that was the impetus for creating the Semantic Web (Berners-Lee et al., 2001), which aimed to extend previous web technologies with new standardised information technologies to unify data across the boundaries of separate organisations and data schemas.

Formal Ontologies clearly distinguish between universals and particulars, dealing directly only with the former. When speaking of universals (e.g., cars), we are speaking of what is true for any particular that instantiates that universal (e.g., this particular car here). This is what makes the ontology formal (as opposed to studying particulars) (Munn and Smith, 2008). Universal classes are organised in a hierarchical structure, ranging from the most general to the most specific, with natural language definitions and formal logical axioms. The universals documented in an ontology describe the types of entities that exist, specific entities in reality are instances of these types, analogous to the distinction between a data schema versus a particular database (content) built with that data schema.

Besides universal classes, ontologies also contain annotations for better human understanding, relationships that maintain between entities such as two universals (e.g., the universal car is a subclass of the universal vehicle) or a universal and its instances (e.g., a particular car is a instance of the universal car) or certain instance of specific classes (e.g., a particular car is owned by a particular individual), typed data properties that individuals can have (e.g., the license number of a particular car), certain features of these relationships (e.g., the inverse of the is owned by object property is has owner) and use them to express necessary and or sufficient conditions with their help for membership in universal classes (e.g., a certain kind of property could be defined as inhering in a certain kind of material entity).

Ontologies nowadays are created with the help of ontology languages grounded in formal logic, mostly in a machine (computer) readable way. Many such languages enable some form of automatic reasoning, with the additional benefit of inferring logical consequences. Many reasoning tasks revolve around the development and maintenance of the ontology itself, such as satisfiability (determining whether the ontology is completely broken or not), concept satisfiability (whether a certain class can have instances without contradictions), or concept subsumption checking (whether a class is indeed a subclass of another class). However, there is also reasoning with data, or, in other words, with the assertions made about individual instances using the classes and relations in the ontology. There are reasoning tasks, such as instance checking (determining whether a certain individual is an instance of a specific class according to the ontology) and query answering (answering queries while also considering implicit information, including implicit relationships and class memberships in the ontology).

The most widely used ontology language with well-defined reasoning capabilities is OWL2 (see in: "OWL 2 Web Ontology Language Document Overview (Second Edition)" (World Wide Web Consortium, online)), which is a W3C (World Wide Web Consortium, an international collaborative organisation to develop and maintain web standards) standard and part of the Semantic Web stack, see Fig. 5. The Semantic Web stack utilizes a whole host

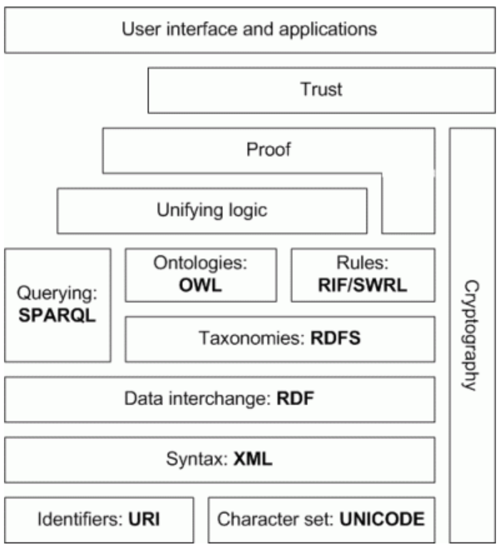

Fig. 5 Illustration of the Semantic Web Stack, as a collection of standards and technologies building on each other

of technologies building on each other to achieve it's goals. One basic building block is the notion of Universal Resource Identifiers (URIs), which are used to identify particular logical or physical resources using web technologies (like a URL identifies a webpage). The Resource Description Framework (RDF) is a way to represent and exchange information with the help of directed graphs using so called triples (see in: "RDF 1.1 Primer" (World Wide Web Consortium, online)), consisting of a subject, a connecting predicate, and an object represented with URIs. Other technologies are built on these like SPARQL to provide query functionality *via* graph pattern matching; other tools, like OWL, create ways to create taxonomies and ontologies, as RDF alone has a very limited expressivity. The Semantic Web standards have also given rise to a great many tools (database engines, query implementations, editors and logical reasoners) for developers and users alike. All are quite independent of the specific domain. The OWL2 standard for creating ontologies is part of this stack and is also based on Description Logics, a computable fragment of First-Order Logic (FOL), and has a large ecosystem of tools on its own.

However, as pointed out by Rudolph and Schneider (2011), reasoning capability, a central question for ontology development in the Semantic Web, comes at the price of expressivity. The full FOL is very expressive but semi-decidable, meaning there is a procedure that will always terminate if the formulas are valid (we can always find a proof if one exists), but if the formulas are not valid, the algorithm may never terminate (we may never find a refutation). OWL2 has several variants or profiles with varying levels of expressivity. OWL2 full is undecidable, but the OWL DL variants are decidable but with a limited expressivity. For example, OWL limits the expressivity of relationships (compared to full FOL) between entities: a relationship can only be a binary relation (e.g., this windshield is part of this car). Expressing more complicated connections that require more parts is not directly possible (like this windshield part of this car at time (*t*)). Also, except for Superclass and Subclass relations, OWL relations are defined only between individuals.

Undecidability can also occur when combining ontologies with other tools, such as rule languages. Nevertheless, as pointed out by Rudolph and Schneider (2011), there can be applications where a logical system is needed only for development, while automatic reasoning with the result is not necessarily required. In such a case, some or all computability can be sacrificed for increased expressivity. Furthermore, even semi-decidable reasoning can have utility in some instances. When all domain knowledge cannot be known anyhow, the inference that something is a logical consequence of the knowledge base is much more useful, than the opposite, which the automatic reasoner can have trouble with. There is also the possibility of creating ontologies with a language that enables efficient reasoning and includes more expressive logical axioms to augment expressivity, at least for human users.

There are other systems for writing ontologies, such as the Open Biological and Biomedical Ontology (OBO) format (Open Biological and Biomedical Ontologies Foundry, online), or one can even write ontologies directly in FOL or Common Logic (CL) directly. The OBO format originated from ontological efforts in the life sciences and is more human-readable than most syntactic variants of OWL but was less well-defined semantically (Golbreich et al., 2007). OBO ontologies can now be translated into OWL (and back) for reasoning support (Tirmizi et al., 2011). On the other hand, CL is a system that sacrifices decidability, even beyond the only semi-decidable FOL, in exchange for even more expressivity (ISO/IEC, 2018).

Building on a formal ontology, we can create formal representations with query and inference support, as well as checks of internal logical consistency *via* inference tools. To provide the rest of what we need, an actual useful representation of our domain that could solve the terminology issue, has good maintainability and extensibility, sufficient expressive power, and the ability to handle overlapping domains of interest, we need to follow a proper methodology when engineering this ontology.

4 The importance of methodology in ontology engineering

Ontology, in its non-philosophical use for solving knowledge representation problems, also known as applied ontology, is employed by many groups, and the number of publicly available ontologies has increased significantly in recent years (McCrae et al., online). This is not necessarily a good thing, as an ontology is more than just its digital form; for it to be useful, it must be widely recognized and used. Having many competing and mutually incompatible ontologies tends to defeat the purpose of having them in the first place. There is no universally accepted definition of ontology. In this article, we utilise the definition, methodology, and principles outlined in Arp et al. (2015). According to the authors, an ontology is a representational artefact that captures what is general in reality by representing universals, defined classes and the relations between them by combining definitions, axioms, rules and constraints. This definition is very different from the most popular competing definition: a formal, explicit specification of a shared conceptualisation by Gruber (online). The difference is what is called ontological realism: an ontology's entities should always refer to universals in reality, as opposed to specific conceptualisations (Smith and Ceusters, 2010), thereby reducing the number of ontologies while maximising their connectedness, which is only possible if people accept certain principles for building ontologies. "The realist methodology is based on the idea that the most effective way to ensure mutual consistency of ontologies over time and to ensure that ontologies are maintained in such a way as to keep pace with advances in empirical research is to view ontologies as representations of the reality that is described by science. This is the fundamental principle of ontological realism." This is an important distinction, as we are discussing real-world constructions. To further elucidate this point in terms of AE, we can think of the difference between building an ontology on first principles and the actual fundamental nature of constructions versus trying to represent our existing heterogenous conceptualisations (e.g., laws, standards and guidelines, not to mention BIM models) that may or may not have a tenuous contact with actual constructions, practices, and processes, not to mention to each other's terms.

Another important methodological issue in building ontologies is the interoperability of ontologies of neighbouring and partially overlapping domains. This is an issue even when adopting a realist approach if these are developed independently. To address this, a hierarchical structure of ontologies is necessary, just as there is a hierarchical structure of terms within them. This hierarchical structure must go from the most general towards the most specific domains, at each level reusing and, if need be, further subdividing (specifying) the previous (upper) levels' terms and making connections to neighbouring ontologies where possible, ensuring interoperability within the broadest possible scope and thus avoiding competing systems and providing economies of reuse.

A branching hierarchical structure raises the need for an uppermost level ontology, known simply as an upper ontology, to contain the most general and completely domain-independent terms. Using such a starting point enables the interoperability of ontologies built on it, at least to the most basic level. One level below the upper ontology should be works that are still very general, often referred to as domain-independent reference ontologies. These should encompass terms used across various domains, such as notions of time or units of measure, to name just two. From this point, specific fields should continue to use a hub-and-spokes approach, clustered around specific domains of interest, building domain-dependent reference ontologies, domain ontologies, and finally, application ontologies for the most specific subjects. No domain ontology should redefine terms widely used by other domains, as this will result in interoperability issues and confusion. Instead, the ideal is to extend (further subdivide and specify) the relevant upper-level terms.

The methodology in (Arp et al., 2015) also encourages open sourcing of ontologies, the establishment of organisational structures for the evaluation, curation, maintenance and updating of ontologies, single inheritance for reference ontologies (a class should be a direct subclass of only one parent class), good naming conventions for the different entities and the use of proper natural language definitions based on the Aristotelian method of genus and differentia to help the human users of the work.

The requirements for an upper ontology (ISO/IEC, 2021a) and the upper-level Basic Formal Ontology (BFO) (ISO/IEC, 2021b) that meet these criteria have recently been formalised as ISO standards. The detailed methodology for building ontologies with BFO is laid out in the book of the same title (Arp et al., 2015) and several underlying articles (Schulze-Kremer and Smith, 2005; Seppälä et al., 2017; Smith and Ceusters, 2010; Smith et al., 2005; Smith et al., 2006).

BFO itself, as an upper ontology, has only a handful of classes and object properties. A basic overview is presented in Fig. 6. BFO's first significant distinction is between occurrents (processes and other entities that unfold over time) and continuants (entities that do not have a temporal part but instead continue to exist and maintain their identity throughout time). Continuants are further subdivided into three major categories: Independent Continuants that exist on their own (like a car), Specifically Dependent Continuants that exist only depending on one or more specific Independent entities (like the colour of a car), and Generally Dependent Continuants that are also dependent on other entities for their existence, but are not tied to specific ones and can migrate between them (like a piece of information represented in a book, but not tied to a single specific instance of a book). A proper, in-depth introduction exceeds the scope of this article; the reader is referred instead to ISO/IEC (2021b) and Arp et al. (2015).

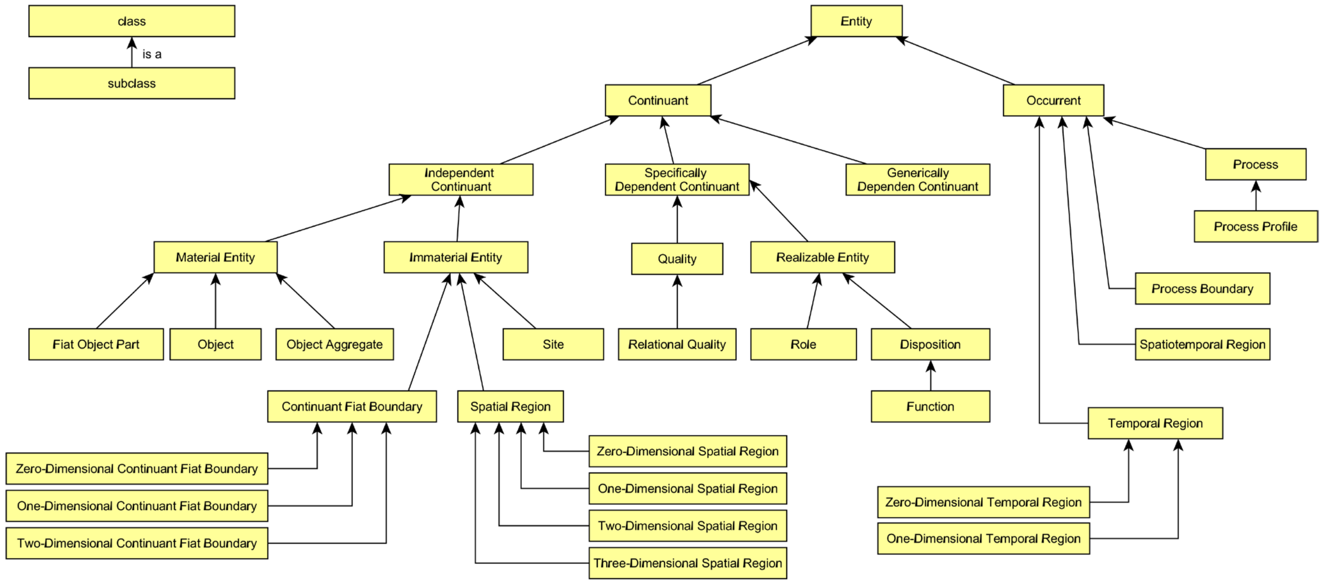

Fig. 6 Overview of the BFO ontology classes

This approach has proven to be very successful in the biological sciences, starting with the Gene Ontology initiative (The Gene Ontology Consortium, 2019) or GO in short, which was established in 1998 to create a unified system for referring to gene functions in the data-driven field of genetics research. The problem is that while certain types of data, like gene sequences, are straightforward to represent in a format that allows easy integration with other sources and analysis, other types of data, like clinical data related to these genes, are not. Annotating data gathered by many different groups, otherwise reported in many ways, such as languages, with the same GO identifiers makes it much easier to handle. The success of GO led to the establishment of the OBO Foundry and a suite of interoperable biomedical ontologies that extended the efforts of GO to other fields of biological research (the entities being studied) (Smith et al., 2007).

5 Hypothesis

We can now formulate our hypothesis in full:

The fundamental problem for computer-aided AE design is to solve the representation problem. This is best done not by extending current data schemas but by basing the representation on a formal ontology built with a suitable upper ontology BFO and using its corresponding methodology.

The importance of the representation problem is well demonstrated in Section 2. Once again, the representation problem is fundamental, not exhaustive, in that the solution to the other problems must rely on an adequate representation. Section 6 present an investigation of the second part of the hypothesis.

6 Survey of existing works

One important principle of the BFO methodology is the reuse, extension, and integration of new work with existing efforts to avoid duplication and achieve a broader user base. The ideal situation would be if there were an established architectural construction or at least an engineering suite of ontologies to which an AE ontology effort could connect. We surveyed existing relevant resources and evaluated them in terms of their potential applicability. This review is not exhaustive and focuses on some of the most directly related works (to AE and building constructions). Other ontologies that are relevant but more general (such as material data) and ontologies of disciplines that neighbour AE engineering will not be reviewed here in detail.

6.1 Classification systems

The AEC industry and regulators have long ago realised the need for classifying entities in the lifecycle of buildings in a unified fashion independent of the different tools (like open or closed BIM schemas) different groups may be using and have created many such classification systems like FreeClass (Inndata Datentechnik GmbH, online), MasterFormat (see in: "Masterformat" (CSI, online)), Uniformat (Conspectus Inc., online), Uniclass (Hubexo North UK Ltd., online) and OmniClass (see in: "About Omniclass" (CSI, online)).

Classification systems are simpler than ontologies and lack some of their most useful features, such as a theoretical background in formal logic and formal ontology, as well as rich relationships and axiomatizations. As pointed out by Afsari and Eastman (2016), different classification systems have distinct purposes, scopes, and entities and properties, as well as different basic structures (classification frameworks), grouping principles (hierarchical or faceted), and very different taxonomies of entities. As a result, it is challenging to create translations or mappings between them, which is another example of the general data silo problem. Classification systems are often classified based on their grouping hierarchies as either hierarchical or faceted. Hierarchical systems group all individuals into a single distinct category, while faceted systems are more flexible and have many dimensions (facets) that can be used independently to classify the same individual. A system built on a formal ontology has significantly more expressive power than even a faceted classification system, as it can construct knowledge graphs with multiple classes of entities in specific relations to each other, even when describing data about a single individual instance. Some classification systems, like the early version of FreeClass, blur the line and have versions in ontology formats. However, having something like an OWL (ontology) version does not necessarily change the nature of the system and make it an actual ontology. Nevertheless, established classification systems can potentially aid ontology creation by identifying many of the types of things that have to be incorporated.

6.2 Domain independent reference ontologies

Although primarily focused on interoperable ontologies for the biological sciences, the OBO Foundry has produced resources with much wider applicability. The Relations Ontology (RO) (see "OBO Metadata Ontology" by Smith et al. (online)) extends the relationships (object properties in OWL parlance) found in the BFO that can be used to form connections between ontological terms. A subset of these, ROCore, is independent of the biological domain. Other ontological resources in the OBO Foundry that have possible uses in other domains are, for example, the Environment Ontology (ENVO) (Buttigieg et al., 2013), the Information Artifact Ontology (Zheng, 2020), the Ontology for Biomedical Investigations (OBI) (which though biology focused has general relevance to scientific study in other domains too) (Bandrowski et al., 2016), and the Geographical Entity Ontology (Hogan et al., online), or the OBO Metadata Ontology (Smith et al., online).

The Common Core Ontologies were created by US government-sponsored projects that use ontologies. It is a collection of mid-level ontologies based on BFO, which are relevant across multiple domains, including Extended Relation Ontology, Time Ontology, Geospatial Ontology, Information Entity Ontology, Artefact Ontology, and Event Ontology (Rudnicki, 2019). It serves many of the same purposes as the more general usage ontologies in the Open Foundry Ontology (OFO) foundry.

6.3 Engineering ontologies in general

Much of the ontological work related to engineering focuses on systems engineering rather than the specific issues of individual engineering fields. In systems engineering ontologies, the main issues are the system description and the function and requirements assigned to the various parts. Mizoguchi and Kitamura (2009) point out that most system engineering ontologies, in terms of describing the actual system, are either device (component and subsystem) or process (phenomena occurring in each part to obtain the desired output) based. Device ontologies are more popular as they are easier to create and to form a clear hierarchy of terms. They also enable the attribution of functions to devices (agents), which is challenging in a process-centric view. The limitation of a device-centric view is that it treats devices as essentially black boxes with specific inputs, outputs, and connections. This hides a lot of the complexity of the actual processes, making them easier to reason with in simple cases. However, they provide very little support when new and innovative solutions are needed, particularly for modified or entirely new types of components and processes.

In Borst et al. (1995), the authors present an ontology of physical systems. Components serve as the fundamental building blocks of systems and are described in terms of mereology (part of relations), topology (the connectedness of the components) and system theory. The topological relations are entities themselves because they need to express that there can be multiple such relations, which can be of a particular kind (which would not be practical with object properties only). However, they also propose a process ontology and a mathematical description of these processes in terms of a mathematical ontology. Another article (Lin et al., 1996) investigates requirement management. Systems are decomposed into multiple subsystems, which are composed of various subsystems or components. Customer requirements must also be decomposed into requirements for the various sub-systems and components.

Nevertheless, requirements are usually defined very informally, which raises many issues in their management.

The authors propose an ontological system comprising a Device Ontology to describe the components and their features and a Requirement Ontology for managing and decomposing requirements. Requirements are described in terms of their decomposition, source (external or internal), and type, as well as physical, structural, performance, functional, and cost. In another article, Darlington and Culley (2008) also propose an Engineering Design Requirement Ontology (amongst two other ontologies). The authors also point out that: "the labels adopted within industry and design research to signify 'design needs' at various stages in the evolution of the design requirement and the design process have not been standardised, and they are used indiscriminately with varying levels of imprecision" (Darlington and Culley, 2008:p.119). We can see that the lack of terminological unity is by no means limited to AE design.

Kitamura (2006) present an ontology of functional design knowledge as an extension to previous systems that were predominantly component or process-centric. They give a taxonomy of functions (functional concept ontology) and separate them from the "way of function achievement". This functional ontology is described in detail in Kitamura et al. (2002). The authors point out that a function-based view identifies parts of a system differently than a component-based view.

The Function-Behavior-Structure (FBS) ontology (Gero, 1990; Gero and Kannengiesser, 2014) relies on design research and gives three non-overlapping ontological categories to describe design objects: function (F) behaviour (B) and structure (S), with connections only between F and B, and B and S, but not between F and S. The FBS framework uses the ontology to represent the design process with certain transformation between the three ontological dimensions. This was then expanded upon as the situated FBS framework, where situatedness means that designing occurs in three interacting 'worlds': the real external world, the interpreted world in the designer's head, and an expected world, which is akin to predictions about future goal-driven outcomes, also in the designer's head. Entities of all three ontological dimensions can have instances in all three worlds. This framework, along with many other systems engineering ontologies, attempts to encompass a wide range of entities, but these can have quite distinct meanings. Take, for example, the various ontological views on functions investigated in a special issue of the journal Applied Ontology. In one of the articles, Spear et al. (2016) analyse and compare basic ontological approaches to functions. The etiological view treats the history of selection (either natural through evolution or artificial design) as a critical part of a function; i.e., a function is a function because its bearer came to be in order to perform a specific task, thereby fulfilling that function. The systemic view treats the contribution the functioning makes to some larger process or system as the key item. The third, or "life chance", approach extends the systemic view and is inspired by evolution, treating only the fact that something contributes something that is also life-enhancing to the larger system or context as the key to being a function. The view of BFO follows the etiological view: "A function is a disposition that exists in virtue of the bearer's physical make-up, and this physical make-up is something the bearer possesses because it came into being, either through evolution (in the case of natural biological entities) or through intentional design (in the case of artefacts), in order to realize processes of a certain sort." (see in: "BFO-2020" by Smith et al. (online)). In the FBS ontology, on the contrary, functions are defined as serving a particular goal, but they are only connected to behaviour and not the structure of artefacts.

Additionally, FBS is not a realist ontology, as it encompasses functions of material things and functions as various mental entities. While a complete description of a design process requires a way to deal with the mental entities and processes of designers, they are very different from the bonafide functions of material entities. This illustrates the need for an upper ontology.

Szőts and Simonyi (2009) investigated the possibility of creating ontology-based software to aid in mechatronic design by semi-automatically generating project-specific ontologies from a general domain ontology. Their main finding was that any meaningful user interaction with an ontological system requires a well-structured and modular ontology, which they described in terms similar to the principles of ontology engineering, as outlined by the authors of BFO/OBO. They also noted the need for an upper ontology and general engineering reference ontologies. They analysed the problem that in engineering design, an ontology must state things about real entities to be manufactured and design entities "encoding" their properties. They found that the resulting axiomatisation (for the connections between the design and physical entities and their encodings) could not be expressed in description logics (or even full first-order predicate logic) in which the semantics of ontologies is laid down (it required second-order predicates and quantification over classes). They described several attempts to solve this but did not offer a definite recommendation. They also distinguished four levels of representation:

1. enumerations of components with assigned properties,
2. topological connection of components without geometry (like circuit diagrams),
3. detailed geometric information,
4. description of functioning, possibly in a verifiable way.

The difficulties of representing planned entities in simulations with realist ontologies are also analysed by Cheong and Butscher (2019). Engineers want to treat possible future entities as real when modelling to fully utilise the ontology's capabilities to represent real-world qualities and processes. As they point out, one saving grace of treating such entities in engineering models as real is that they still must conform to reality to some extent for the models to be useful in the first place.

Despite extensive work in the engineering and manufacturing domains, far too numerous to enumerate here, the success of ontologies in the biomedical sciences has yet to be duplicated. Borgo and Lesmo (2008) observed that the number of ontologies was increasing, even in industrial and engineering domains. However, most of them are what they named "shallow" ontologies (ones without rich axiomatisation, definitions and documentation) with few additional benefits compared to simple taxonomies. They investigated issues such as ontology interoperability and reuse, which, as they state, necessitate a foundational ontology (upper ontology). Tchouanguem et al. (2021) attribute the relative lack of success of ontologies in industrial applications to the ad-hoc nature of ontology development in engineering, namely the lack of a proper methodology. To improve on this, more projects have started to adopt BFO as an upper ontology and its principles of ontology development in recent years. A comprehensive collection of publications on engineering ontologies using BFO can be found at the National Center for Ontological Research (online).

Perhaps the largest-scale engineering ontology initiative is the Industrial Ontologies Foundry (IOF), with the direct goal of replicating the success of the OBO Foundry for the manufacturing industry. The IOF aims to create a whole suite of reference ontologies for the manufacturing and engineering domain. The focus on manufacturing differs significantly from ours, but it may be extended to the building industry in the future. The project has published relatively little of its results to date, including a proof-of-concept ontology (Smith et al., 2019), a proto-product life cycle ontology (Otte et al., 2019), and a proto-production planning Reference Ontology (Šormaz et al., 2020).

The IOF core ontologies will certainly contain many relevant terms for AE. The product lifecycle in IOF is analogous to the building life cycle, while the production planning process also has much in common with the building planning process. The principle of ontology reuse would strongly suggest that any possible future AEC ontology effort be aligned with the IOF.

The Physics Based Simulation Ontology (Cheong and Butscher, 2019), which is unfortunately only described in a single article without an actual formalised, published ontology, is another BFO-based work aimed at describing how physics-based phenomena are represented in simulations. It has two parts: Physics based Simulation Ontology (PSO)-Physics, describing real-world physical entities, and PSO-Sim, which describes entities in simulations that represent real-world entities in PSO-Physics.

Hagedorn et al. (2019) have investigated engineering ontologies in general, with a specific focus on applications in additive manufacturing for medical purposes. They also state that the lack of success of previous ontologies for engineering is due mainly to their inability to work together if the development lacks a suitable methodology to promote interoperability, modularity, and reuse. Like the IOF, they then utilise BFO to construct an ontology framework for their proposed design method in additive manufacturing. They partially reused existing ontologies and taxonomies by aligning them with the BFO and the Conceptual Cognitive Ontology (CCO) while also creating new, clean-sheet ontology modules for areas where no suitable earlier work could be found. They demonstrate the value of aligning different ontology modules under a system like BFO, using a case study of a design exercise. In their conclusions, they highlight four key challenges to the use of ontologies in engineering: the lack of acceptance due to previous failed ontology attempts, the difficulty of using Semantic Web tools for non-experts, the time-consuming nature of developing engineering ontologies and interoperability issue between ontologies and existing design software. They claim that all four points can be answered with the correct methodology and tools. Regarding the time-consuming nature of ontology development, they point out that the work becomes much easier once one has some previously properly constructed engineering ontology resources that only need to be further developed and specified for the specific engineering domain.

6.4 AEC industry: BIM to Semantic Web

There have been many attempts to bridge the gap between BIM data schemas, most notably the IFC schema, and the world of the Semantic Web by creating automatic translations between the BIM data schema and a corresponding ontological system and, as a result, between BIM models and the Semantic Web's preferred data format RDF. The goal is to utilize the tools developed for the Semantic Web (e.g., query languages like SPARQL and description-logics-based reasoning) (Boje et al., 2020) to try to solve BIM interoperability issues and to enable the linking of entities in BIM models to anything in the Semantic Web itself (Pauwels and Terkaj, 2016), thereby going around the expressivity limitations of the specific BIM schema. In Dinis et al. (2022), the authors review articles on the semantic enrichment of BIM data. The works they analysed focused on adding semantic information to existing schemas, most notably IFC, to address their limited expressivity and lack of automated inference methods. They found many approaches using various AI techniques, but most reviewed articles utilised Semantic Web technologies as a primary tool. The additional general trend they found was a move toward open-source tools and cloud-based applications. Regarding ontology development, some authors highlight its difficulty. This is undoubtedly true, but the central question is: which other methods show promise in creating semantic systems that are potentially interoperable across multiple domains and subdomains?

The difficulty of such a translation for IFC lies in its foundation on STEP (Pratt, 2001) and its reliance on the EXPRESS modelling language (ISO, 2004), an industrial product data exchange format. IFC was neither developed as an ontology nor used in the principles outlined earlier and in the BFO-related literature. The terms in STEP and IFC are too broad and will result in clashes with other fields. While it is possible to translate at least most parts of the IFC specification into the OWL syntax, this does not automatically guarantee all the benefits of good ontologies. The main positive aspect of STEP, on the other hand, is its geometrical representational capabilities, which, for the most part, are not the primary focus of ontology development (Smith, 2017). Such translations include ifcOWL (Beetz et al., 2009; Pauwels and Terkaj, 2016), ifcWoD (de Farias et al., 2015), which does not include geometric data, SimpleBIM (Pauwels and Roxin, 2016), which is also a simplification with no geometric and representation data, that collapses IFC property classes to direct data properties of OWL classes and EifcOWL (Tchouanguem et al., 2021).

Some initiatives, such as the Building Element Ontology (Pauwels, online), utilise only part of the IFC schema, specifically the IfcBuildingElement subtree, to just reference physical elements. The Building Product Ontology (BPO) (Wagner et al., 2022) aims to provide a semantic description of building products, their components, and their connectedness, as well as the assignment of properties to different data types. No geometry or even material composition is included. It is not based directly on the IFC but on the buildingSMART Data Dictionary, which creates extensions to the IFC schema.

Similar projects exist for other open BIM data schemas, such as COBie and COBieOWL (Farias et al., 2015).

ifcOwl is an official project of the buildingSMART alliance, the organisation responsible for IFC, and utilises a completely automated translation. As pointed out by Venugopal et al. (2015), IFC has inherent drawbacks, such as ambiguity due to its modelling freedom, which, on the other hand, can make even seemingly straightforward tasks, like calculating the volume of a construction, not straightforward.

As pointed out by Building Topology Ontology (BOT) (Rasmussen et al., 2020), IFC also has constructs (e.g., ordered lists, objectified relations) that are necessary to maintain a two-way translation but deviate from best practices for ontologies, making any ifcOWL very inefficient to reason with. The resulting ontology is very large, just as challenging to understand for human users as IFC, and does not address the modularity and extensibility issues of the base schema itself. Its use is for translation to link data to BIM models based on other ontologies.

SimpleBIM and ifcWoD try to mitigate some of these issues by dropping the geometric representation of the IFC schema altogether and replacing objectified relationships with object properties (the direct way of representing relationships between classes and individuals and between individuals in OWL, without intermediate steps), and in the case of SimpleBIM the objectified IFC property classes to data properties. This makes querying the resulting models much easier, as many intermediate steps are eliminated. However, these efforts aim to improve query performance for a subset of the data already represented in IFC, rather than creating an ontological system with better expressivity. Another similar approach, though not built as an ontology, is BIMRLSS (Solihin et al., 2020).

The authors of EifcOWL analysed the other IFC to OWL translations in terms of their adherence to some of the ontology engineering principles laid out by the authors of BFO and GO. They point out that IFC does not have a well-developed single-inheritance hierarchical structure for its classes, as most inherit directly from IfcRoot, while the generated ontologies often lack useful definitions. The authors attempted to remedy this by providing proper natural language definitions, placing their IFC classes under the corresponding BFO classes, and pruning the inheritance tree. Their analysis lacks an evaluation of IFC in terms of other principles, such as ontological realism. IFC is full of definitions, such as "IfcProperty is an abstract generalisation for all types of properties that can be associated with IFC object through the property set." An abstract generalisation of a property is not the same as a property. It is also challenging to put IFC classes under BFO classes, like EifcOWL:IfcProperty under BFO:Quality, as it is not clear that the things one can put in an IFC property set can be qualities in the BFO sense.

Another issue with translating IFC models into RDF graphs *via*, for example, ifcOWL for linking and querying is that many of the queries we would like to make are inherently geometrical in nature. While the raw geometrical information can also be translated, a geometric kernel is still required to process it, as graph query languages typically search for patterns within the graph only. BimSPARQL seeks to add such functionality to the Semantic Web's SPQARQL query language (Zhang et al., 2018). The publication also lists other similar works. These works are important steps toward a unified BIM query and rule-checking system that requires geometric information.

An alternative approach utilises the IFC schema's ability to store arbitrary metadata, enabling the direct embedding of RDF data within the BIM model itself. The additional problem with this approach is that the resulting system remains completely file-based, so linking to other resources is one-directional only (Beetz et al., 2014).

6.5 AEC industry: original ontologies

Due to the widespread application of existing open BIM standards, it is understandable that many ontology initiatives attempt to adhere to them. However, there are inherent contradictions between data schemas, such as IFC/STEP, and best practices for ontology development. Clean sheet ontologies can avoid these contradictions at the risk of developing additional separate systems with the same interoperability issues that open BIM intended to solve in the first place.

A good collection of ontologies in the AEC domain is found in (Abanda et al., 2013). The authors note that there has been a general trend to create increasingly more comprehensive 'heavyweight' (richly axiomatized, annotated and documented) ontologies, as opposed to the earlier dominance of 'lightweight', vocabulary-like works. Construction works dominated earlier works, but efforts focused on BIM and sustainability have recently become more numerous. They also note that most ontologies failed to become widely used, and there are still very few actual services that utilize them for regular users. From the table of ontologies they created, it is immediately apparent that engineering design and actual construction objects (as opposed to construction works and management entities) are poorly represented. Of the few engineering ontologies, they list only one that directly deals with structural engineering, which is no longer available (Zhang et al., 2010). AE is entirely missing, apart from theoretical works that lack original ontological content or exhibit only slight partial overlaps with fields such as building energy and sustainability.

The DOGONT (Bonino and Corno, 2008) ontology is aimed at home automation and indoor IoT networks.

The WE3 Linked Building Data Community Group developed the BTO, or Building Topology Ontology (Rasmussen et al., 2020). Their goal was to facilitate the sharing of BIM data in the cloud (linked building data) using existing Semantic Web technologies. While, as they write, this is mainly achieved through some IFC-to-OWL translation (such as ifcOWL), the non-IFC-based approaches also require a structure to align with each other for interdisciplinary communication. To this end, they created a small ontology of building topological entities (building, spaces, floors, zones) and their relationships, representing only the topological structure without specific geometry. Any specific geometry or specific BIM model entity can, if required, be linked to these topological entities.

The Building Ontology (Chávez-Feria and Villalón, online) is an extension to BTO by the BIMERR Ontology Network. Its focus is quite narrow: energy retrofit projects. Unfortunately, it does not appear to follow a formal methodology for ontology building, and it defines terms that are used in many other, much broader domains, such as "Boiler", "Fan", and "Outlet". The purpose of the work appears to be to support the design tools developed by the project itself rather than creating an ontological system of broader utility.

The BIM shared ontology (BIMSO) and the BIM design ontology (BIMDO) (Niknam and Karshenas, 2017) are another clean-sheet design, as the authors deemed it unfeasible to extend the IFC schema to include all types of entities in the AEC domain. It is built using the NeOn methodology to create a shared ontology for the whole AEC domain. BIMSO utilises building element classification schemes, the QUDT ontology, and FreeClass OWL, and is built on a building element-based foundation. It handles element identity, size, material properties, and some relations, such as 'intersection' or 'hosts'. The BIMDO design ontology contains terms about the project properties. Unfortunately, the ontologies themselves have not been made public.

The E-Cognos ontology (Lima et al., 2005) focuses on construction works. It utilises the BSI (2008) and UniClass classification systems and mirrors some of the structures of IFC connections, although not in all cases. The basic ontological model is "a group of Actors uses a set of Resources to produce a set of Products following certain Processes within a Work Environment (Related Domains) and according to certain Conditions (Technical Topics)".

The article of Lee et al. (2016) is perhaps the closest in its subject matter to AE practice. They present a framework for gathering and querying building construction defect information aided by a small ontology (Lee et al., 2016). Unfortunately, their ontology is very small and narrow in its focus, with only six major classes: Work_result, Space, Element, Material, Defect_cause and Defect_type, the first four of which are intended to be instantiated with data extracted from BIM models (of the faulty construction). It uses the Omniclass classification scheme to subclass the major classes. It does not follow any upper-ontology or an ontological realist framework.

Another tangentially relevant project is Simultan (Bednar et al., 2020), a program to integrate various simulation tools for modelling buildings or urban contexts. It aims to solve the interoperability problem and data redundancy when performing such calculations with different specialised software. The theoretical background is described in Paskaleva et al. (2021). The authors analyse the techniques usually proposed to solve data exchange problems between simulation tools, which they categorise into syntax only, common semantics, copy of reality, common knowledge, and common data. The ontological approach categorises this as common semantics and critiques it by stating that an expressive enough representation becomes unmanageable. They put IFC in the same category and give a detailed explanation of why it is not suitable for the task either. Their proposed solution to the problem is based on Model Based Engineering, but it only applies to software with an exposed public API of a particular kind. They also propose a small and flexible ontology to describe models, inputs, outputs, and functionalities of different models, although it is not an ontology in the sense used in this article. It has only 19 classes and seven enumerations in total, which the software uses to build project-specific "ontologies". The project aims to develop a web service for building these simulation systems, including real-time collaboration among different project participants and version tracking of all data.

The Digital Construction Ontologies (DICO), previously known as DiCtion, is a shared ontology suite for digital construction workflows (Zheng et al., 2021) specifically designed to aid in planning renovation works for occupied buildings. Such projects involve complex scheduling tasks, constraints, and differing views of the project among multiple participants. It is difficult to maintain situational awareness, a term borrowed from military jargon meaning "A single identical display of relevant information shared by more than one command." (Chairman of the Joint Chiefs of Staff, 2011:pp.46), in such a situation. The project is working on software solutions to share the status of activities and documents in a unified system. They also utilise Lean construction research (Koskela, 1999) and Activity flow modelling (Garcia-Lopez, 2017) from construction planning research to track activities in relation to the material entities involved (the construction itself, materials, tools, people, etc.). Their research also includes the evaluation of other ontologies in the construction industry with the general critique that they mostly lack terms for information and information systems and cannot represent multiple contexts of the same project. As a result, they developed their new clean sheet ontology, which is aligned with the BFO and many W3C ontologies, and has a modular structure. The main building block is the description of construction flows, which involves keeping track of the corresponding material entities. Of the numerous modules of DICO, the most relevant to us are the Entities and Materials modules (Valluru et al., 2020). The entity module is quite limited in terms of expressivity due to the construction management focus of the whole project. The main class is a very generic BuildingObject that can be anything. A subclass of BuildingObject in the Materials module is MaterialObjectStructure, which allows for the assignment of materials through its subclasses: Layers and LayerSets, Constituents and ConstituentSets, and Profiles and ProfileSets. This mimics the corresponding schema in IFC. Topological representation can be achieved by using BOT and geometry with ifcOWL, where the construction objects are mapped to the DICO Building Object class.

While DICO is aligned with BFO, certain aspects that mimic IFC may be questionable from an ontological realism standpoint. For example, a MaterialObjectStructure is defined as a "concept to represent material-related information that has material-related properties". A wall of a building is not a concept to represent material-related information. DICO also distinguishes an abstract material class from a class whose members are portions of some materials. This also violates the principle of ontological realism, as there can be no instance of an abstract material that has no actual material part. In DICO, this is probably done to conform to IFC. Many material properties are treated as simple data properties of the material classes, which does not allow for the expression of complex material functions or physical state and measurement method-dependent properties.

The Digital Construction Information Ontology (Törmä and Zheng, online), which is part of DICO, is an ontology module for information entities in the AEC domain. Instances of information entities are not real-world things but rather pieces of information that may refer to real-world things. A BIM model is an information entity, as opposed to a real building the model represents. This important distinction is overlooked in BIM schemas like IFC that freely mix the two in its definitions (IfcMaterial is a homogeneous or inhomogeneous substance that can be used to form elements -physical products or their components, while IfcMaterialProperties is defined as something that assigns a set of material properties to associated material definitions).

6.6 Summary of the literature review

The literature is vast, and there is much more than can be summarized in a single article. There is also a cautionary component to this: many of the ontological works published do not live beyond the initial research phase, and there is still little actual use of ontologies in the AEC sector. As stated earlier, much of this is a consequence of a lack of proper methodology and the fact that ontology is simply hard, while the momentum of existing schemas is enormous.

In terms of engineering in general, most of the work focused on system engineering theory, with limited information available about specific engineering domains. Systems engineering ontologies often focus on different aspects of the process (such as components, processes, functions, and requirements), but none are dispensable.

Ontological research in the AEC domains is heavily focused on BIM, with most efforts aimed at solving the BIM interoperability problem or bridging BIM with the world of linked open data. The presence of IFC is understandably dominant here, as it serves as the logical meeting point for many technologies. Even ontological projects not directly based on IFC often adopt some of the IFC structure, along with classification schemes. This inherits many of the problematic features of IFC, which is not very friendly for AE. Besides BIM, the construction management and sustainability fields stand out in the recent literature, both of which are only tangentially related to AE. In terms of AE, in particular, we can find relatively little beyond what is typically represented in BIM models. This review did not include some adjoining domains not exclusive to the AEC industry (such as materials science, products, and physics-based simulation).

The two most important connection points identified are the DICO and IOF. DICO is more closely related, but its scope, at least thus far, is somewhat limited, as it is a product of a specific application-focused project. IOF, on the other hand, intends to encompass many of the adjoining fields, but it has published very little to date, and the AEC industry is not among its core interests. Both are BFO-based, but there have been no publications about their mutual interoperability.

7 BCON: a new ontology suite for the AE of building construction

7.1 Domain statement

We propose creating a new system of ontologies called Building Construction Ontologies (BCON), which is intended to cover the domain of AE in building construction. The immediate goal is to focus primarily on the actual constructions as they are and function, rather than the results and objects of the design, and not on the design process itself. The treatment of domain-specific aspects of the AE design process could be a later extension. The capture of building works themselves must form part of this ontology, as much as it is directly relevant to the structures (as opposed to other questions, such as construction management). The description of building construction must at least encompass their materials and other qualities and dispositions, structure and connectedness, the building operations that create them, their functions and functioning, the processes that affect these, as well as the measurement and assessments of the functions and processes.

BCON does not intend to be a BIM model or create its own geometric representation; however, for many of its possible applications, it will need to be coupled with some CAD-like geometric representational capabilities. It must also be capable of linking to operate with existing BIM schemas, such as IFC/IfcOWL, but not necessarily *via* an ontological mapping of entities.

By its very nature, any AE ontology will have to deal with many adjoining fields, such as architectural design, construction techniques and management, material science and manufacture, various engineering fields like structural engineering, design principles in general, physical phenomena from mechanics to building physics and fire safety, other sciences like building chemistry and biology, climatology and meteorology, which are huge and complex fields in their own right. BCON cannot intend to cover these in detail according to the principles of ontology reuse and modularity (and general common sense) but will, in due time, form an opinion on what other ontologies its users should rely on in these fields.

7.2 Principles

We intend BCON to be an open-source suite of modular ontologies, eventually evolving into a small collection of tools under a Creative Commons BY 4.0 License (Creative Commons, online). The proto-ontology introduced in this article is found at (Bakonyi and Dobszay, online).

BCON will utilise BFO (ISO/IEC, 2021b) as the upper ontology and the ontology development principles outlined in Arp et al. (2015). The most important one is ontological realism: our goal is to describe constructions as they are without relying on any design process, existing product, standards, guidelines, or BIM models.

In terms of ontology reuse, alignment with other ontologies in the AEC domain and the adjoining domains listed earlier is key. However, we have not yet finalised our exact strategy. As stated earlier, the Digital Construction Ontology (DICON) and IOF ontologies are the best candidates so far. In this paper, we present a small and more or less clean-sheet proto-ontology as a proof of concept without frontloading the effort with all the problems resulting from the relative underdevelopment of ontologies in the AEC domain. We will borrow terms from other ontologies where appropriate, but we do not intend this to be a final product. The proto-ontology presented here will also borrow terms from the Common Core Ontologies and the Physics Based Simulation Ontology.

7.3 Main goals

The first goal is to initiate the development of a universal terminological system for the AE domain. Such an agreement is needed not only in modelling but also for general knowledge and data sharing in both the academic and practical sectors.

The second goal is to achieve an unambiguous and meaningful representation in models. We aim to highlight the significant aspects of building constructions and utilise them for more direct representation rather than relying on legacy systems. This representation should strive to achieve all the desiderate elements laid out in Section 3 (adequate expressivity, topological information, extensibility, maintainability, and so on). In doing so, we aim to create a more transparent system for users, one that is also better equipped to support querying, domain-specific algorithm development, and AI research.

This research is an application of formal ontology; the goal is to create a framework rather than include all possible types of entities or formalise all AE/building construction knowledge. In Section 7.4, we propose a "framework for this framework": not all the necessary terms, classes, and relations, but a first proposal for the high-level structure. We provide a high-level overview of the main parts of the ontology (with other scans to follow later), as well as our preliminary findings about them. More study is needed for each of these groups.

7.4 The structure of the core ontology

In this article, we introduce the core classes, along with more specific subclasses that may eventually belong to specialised modules (such as waterproofing or thermal insulation), added primarily for illustration purposes. As stated earlier, no selection was made for the domain-independent and domain-dependent reference ontologies to use. Many classes presented will ideally be imported from such ontologies to allow ontology interoperability. The main classes defined by BCON are not exhaustive, either, and many are only described in the linked proto-ontology. A high-level overview of the ontology is presented in Fig. 7.

7.4.1 Material entities

The basic building blocks to describe the makeup of the whole 'Building' (viewed as the material entity comprising all of the constructional parts) are '**Building Construction Object**s', **'Building Construction Systems'** and the materials and products they are constructed from. In BFO, Material entities are subclasses of Independent Continuants that have some portion of matter as part and are extended in three dimensions. Construction Objects and Systems have to be described in terms of mereology (part-of relations) and topology (connections) *via* '**Building Construction Connection**s'. Building Construction Objects and their connections are created or modified *via* some **'Act of Building'** (a process, see later) in relation to some preexisting objects. In BFO, Objects can have other objects as parts, which gives flexibility for describing constructions at different levels of granularity. **'Batches of material**s and **'product**s form the material inputs to the Acts of Building that create the Construction Objects.

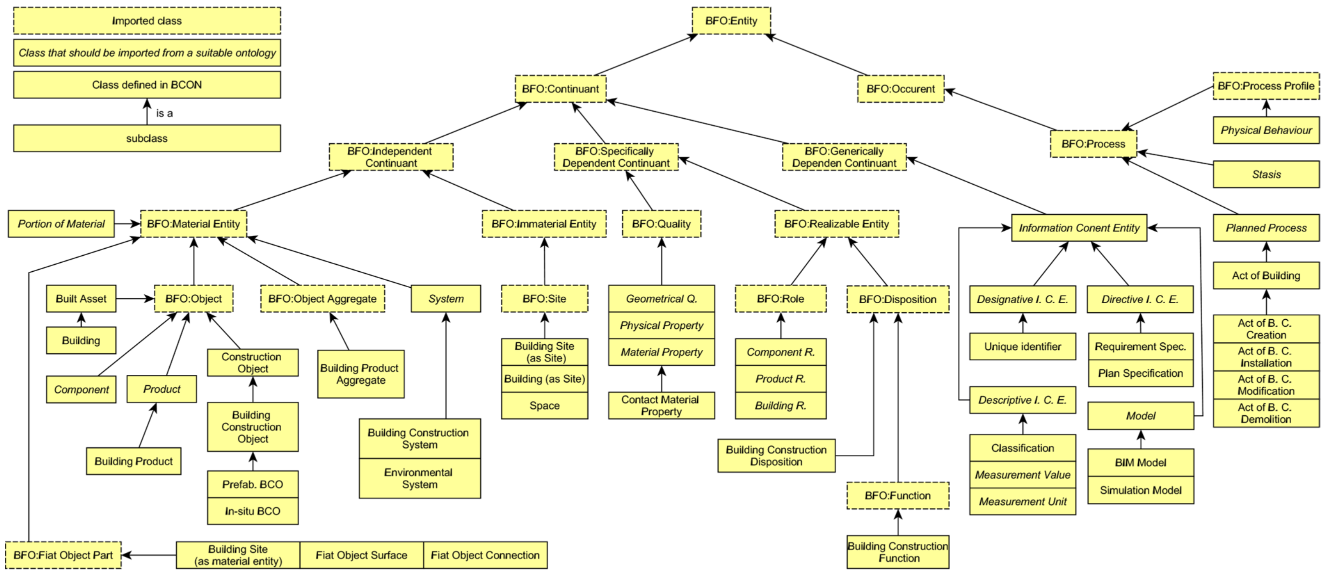

Fig. 7 Overview of the major classes in BCON, under the appropriate BFO classes

Any particular entity can belong to (instantiate) multiple classes throughout its life. As an example, a single window can be both a Product and a Building Construction Object once it is installed. Like in the IOF ontologies, the same piece of matter can "become" an instance of a different universal class through some process. Another example is a masonry block that becomes part of a structural member through a process of bricklaying. Still, BCON can maintain a distinction between more or less raw input materials or building blocks and the actual construction objects that form parts of the building after a process of construction. If needed, we can also distinguish a third life stage of a building's material entities: the rubble that a construction becomes after a process of demolition (or collapse).

'Building Construction Connections' between different Building Construction Objects are of a specific type and bear qualities such as physical and even material properties. The latter is the case because some material and physical properties must be directly associated with them, and where different materials meet, the material structure is often altered (for example, consider construction joints in concrete structures or contact corrosion problems). Due to these factors, it is not enough to represent connections of objects simply as relations between individuals (object properties in OWL), but a Connection must be a separate class (a reification of the connection), which is a material entity itself (so it can bear physical and material properties). The PSO ontology faced a similar problem when assigning material properties to surfaces (like the longwave infrared emissivity, which is a property of the surface, not the bulk material) needed for properly defining boundary conditions. PSO introduced the class of **Fiat Object Surface** as an extension/subclass of the BFO **Fiat Object Part**: a material entity that is a Fiat part of an object (demarcated not by actual physical boundaries but by fiat) that is minimal in one spatial dimension. BCON expands on this notion and introduces the class of **Fiat Connection Surface**, which is formed between two objects. We can use Fiat Connection Surfaces to describe all that we need about the connections of Building Construction Objects, see Fig. 8. The Fiat Connection Surface can have a subclass taxonomy to represent different types of connections (such as the connection between masonry and plaster, welding, or simple mechanical contact.). These reified connections can bear qualities and dispositions that further describe their nature, while other connections can either be inferred (as in the example of determining which connecting element came first in the example) or be stated through other relationships (e.g., whether one element bears the weight of another).

Regarding '**Building Construction Objects**', we have said very little thus far. Most BIM and classification systems have large taxonomies of such entities. But what should be the basis for the subclass hierarchy of construction objects in BCON? Due to modern architectural and construction trends, it makes little sense, in our opinion, to create a hierarchy of building products or construction objects based on function, which is usually a core part of classification systems. Nowadays, a product or constructions traditionally classified in some way is used for multiple functions (like vapour-tight membranes that also act as airtight barriers, temporary waterproofing, separation layers, possibly Ultra Violet (UV) light

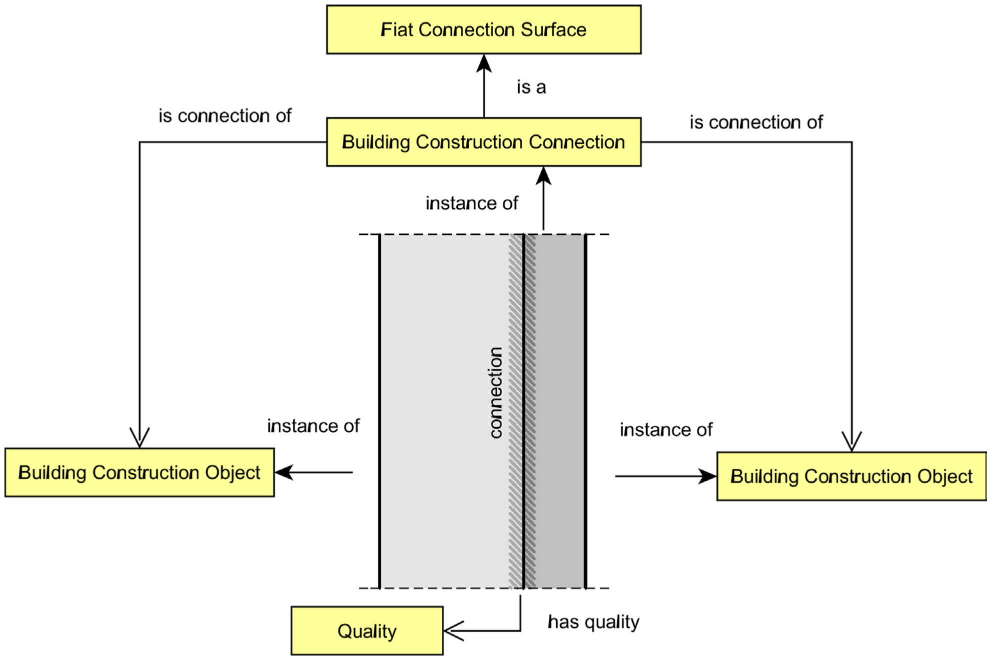

Fig. 8 Illustration of Building Construction Connections in BCON: connection between two touching objects, like a masonry and a plaster layer

protection for other layers, or an installation layer which is NOT thermal insulation even if it is made from thermal insulation etc.) or even in a completely different ways to what was the original intention (think of the architectural trend of putting all kinds of claddings on roofs conventionally unsuitable for the role (Dobszay, 2011)). Even a wall, which is usually found in every BIM system and at a cursory glance may appear to be a simple case, can become complicated, see Fig. 9. The problem is the same when trying to use material and/or product as the basis: the same materials can have a hugely different function depending on the situation; therefore materials and products are best left as another dimension of construction objects and treated separately. Existing classification systems of materials and products can also be linked, but as they really are: pieces of information relating physical objects to a system of conceptualisation, not as actual, real-world categories of construction.

Fig. 9 A few examples of what architects can call a wall: (a) a simple case in classical architecture; (b) a perforated wall; (c) a curved and inwardly sloped wall in a Zaha Hadid building; (d) a wall that seamlessly becomes a roof or a slab

Instead of a functional taxonomy of Building Construction Objects, BCON proposes, for the time being at least, only two major subtypes: *in situ* and Prefabricated BCOs. A Prefabricated BCO (see Fig. 10) is one that is identical (or at least a significant part of it) to a particular Product, which served as input to the act of building (regardless of what that product was like). The word "prefabricated" is used differently than it is traditionally in the construction industry; we refer to all construction objects as prefabricated that are created by installing products that were essentially formed to their final material consistency and shape before they arrive at the site where the construction is being built. This includes prefabricated structural members, as well as items such as windows, Doors, waterproofing membranes, and even thermal insulation boards.

In situ construction objects (see Fig. 11), on the other hand, are created from material parts in a way that their form and final material structure are new, and they are not identical to any of the inputs of the process. This usually involves some process of phase change (like concrete or plaster setting). Examples include plaster or mortar layers, *in situ* cast concrete or liquid-applied films.

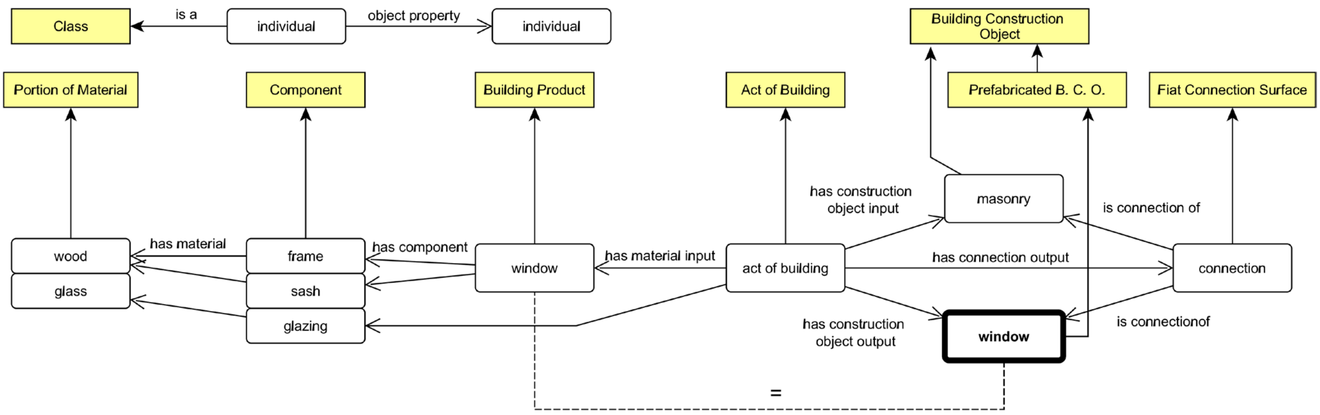

Fig. 10 An example of a Prefabricated Building Construction Object instance and the most important connected classes and instances

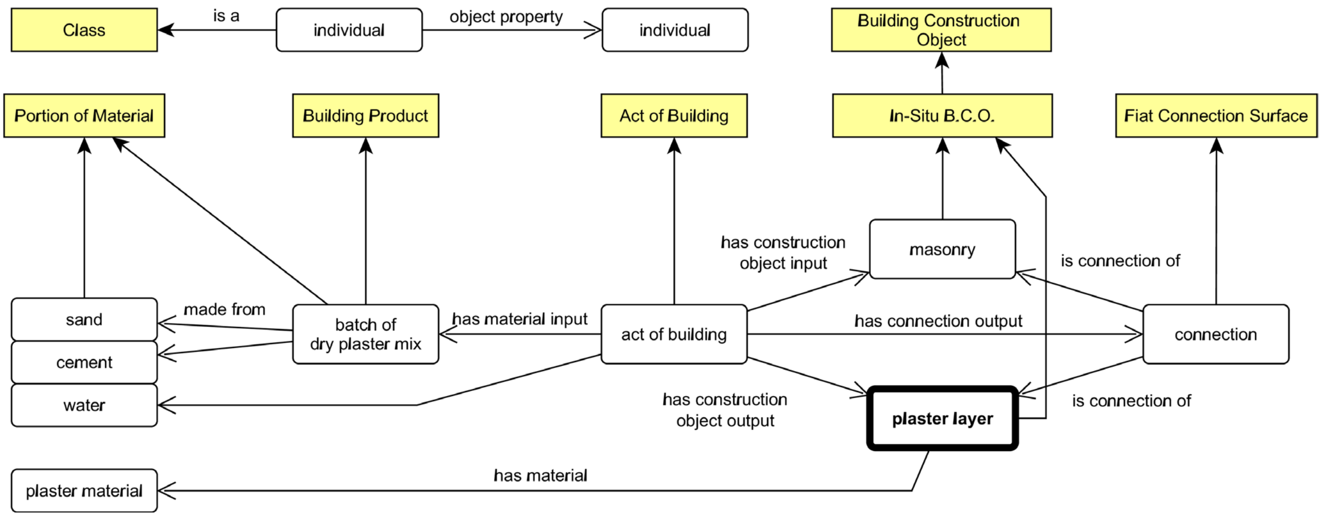

Fig. 11 An example of an *in situ* Building Construction Object instance and the most important connected classes and instances

We believe that this prefabricated vs. *in situ* distinction captures something fundamental and important about Building Construction Objects, just as the material and Product versus Building Construction Object distinction does. Further subclasses of Prefabricated and *in situ* Building Components (BCOs) will be added later. Readers may still object: how can BCON do without the familiar construction types found in BIM schemas and classification systems? Let us look at an example: should we have a class of "Ring Beam" (a ring beam is a type of beam found at the top of walls around the perimeter of slabs that is mainly used to support horizontal loads, distribute vertical loads more evenly, and join together other structural members)? What things do people at times understand under the notion of a ring beam? A structural member, or multiple structural members, a structural member that, at certain places, can also serve as a lintel, a structural member that is often part of a larger structural element (the slab), a structural member with a typical cross-section and reinforcement, or a collection of functions stated earlier.

Additionally, a ring beam is typically made from reinforced concrete, although not always, and can exhibit various subtypes and properties. A functional hierarchy of such structural members would either lack expressivity or would have to result in a combinatorial explosion of subtypes (even for faceted classifications). A good ontology will have to disambiguate the different parts, qualities, dispositions, functions, and processes of ring beams (and any other elements), after which one can be properly and unambiguously described with all its ontological dimensions. Another way to look at it is that many terms used in AE design refer to very abstract concepts. These can encompass many things (such as the notion of a Ring Beam and all its constituents listed earlier) and are, therefore, very useful for human designers to reason with efficiently. However, they are not suitable for a realist ontology that deals only (at least for the time being) with actual physical constructions and aims for unambiguity. Another way to look at it is that many entities in traditional classification systems are more properly patterns of multiple entities and relations in a graph model created based on BCON.

'Building Construction Systems' are formed from many Construction Objects and their connections. In AE design, these systems constitute a minimum sufficient set of constructional elements that are built to perform a specific function. These systems are more than just the sum of their physical parts, as they require the proper interaction of their components through specific processes to produce the desired result. One example is a waterproofing system on a roof that serves to keep liquid water away from the interior and parts of the structure itself by creating an impenetrable barrier and providing a means of collecting and disposing of the water that falls on it. Buildings have many similar systems, and these systems can even overlap with each other when certain individual construction objects play multiple roles.

7.4.2 Functions

The constructions of a building as a whole serve certain well-defined functions (like the separation of various environments visually, thermally, and acoustically.) and the various surfaces/parts of the whole building are created to (ideally) perform these and derived functions in their own context (e.g., a vertical wall is in a very different context than a flat roof in terms of keeping rainwater out). Due to the multiplicity and demanding nature of these functions (like the very limited thermal transmittance allowed for contemporary envelope constructions), often dedicated (sub)systems are created to perform certain functions even within a single planar construction (separation of functions). An example would be a layered wall in a reinforced concrete framed building: the main loads are carried by the frame, sound insulation is mainly provided by a heavy masonry, airtightness by plaster layers on the masonry, thermal insulation by a thermal insulation layer at the outside of the masonry, protection against external weathering by a specialized reinforced plaster layer on the insulation. These functions are realised in corresponding processes, but this is not always the case as components and subsystems fail. Therefore, the functioning must be evaluated according to specific requirements, using specific measurements and classifications. BCON will need to have a functional ontology, a requirement ontology, and an ontology of related processes (including physical processes, measurement, and evaluation processes).

In BFO, functions are a subset of Dispositions (Spear et al., 2016), which in turn are so-called realisable entities. Realisable entities inhere in other entities but do not have to be realised all the time. They are realised in certain processes, as shown in Fig. 12. Indeed, some building construction functions are exhibited all the time, such as the function of a column to hold up a roof, and some are exhibited only occasionally, like the function of a roof to keep out rainwater. Some are not exhibited at all, like the fire protection function of particular constructional elements that never experience a fire. Dispositions are inherent in the physical makeup of their bearer, while functions are a subset of dispositions whose realisation is the goal of creating some entities in the first place.

7.4.3 Qualities and dispositions

Functions and processes are both intimately related to the corresponding qualities and dispositions of building constructions. In BFO, qualities are entities that inhere in and are therefore dependent on other entities, just as dispositions and functions do. Unlike dispositions and function, however, qualities are always fully realised and exhibited. At a minimum, BCON will have to cover types of qualities, including relational qualities, geometrical qualities, physical and material properties, and constructions. After Cheong and Butscher (2019), physical and material properties are distinguished as qualities that are characteristic of the physical state of a material entity (like temperature or pressure) or characteristic of the material makeup of the entities (like density or thermal conductivity). There can also be some generic qualities that do not fall into the previous categories, such as the cleanliness or dryness of a surface.

In BFO, qualities are distinct from their measurements or values. Values are derived *via* a process of measurement and are represented separately with information

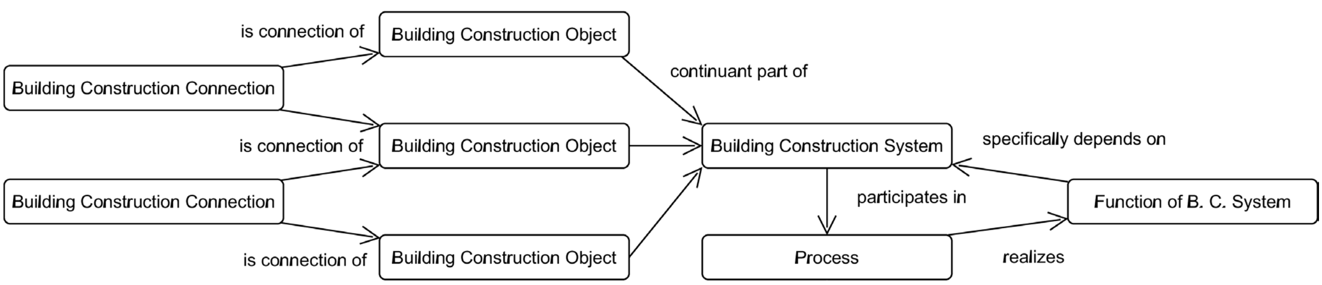

Fig. 12 Building Construction Objects, connections and systems, processes and functions

entities. Qualities are not normative either; i.e., they are neither good nor bad. For example, the thermal conductivity of a piece of thermal insulation is a quality that can be measured according to a specific methodology to obtain a value, which is then evaluated based on a metric to yield a normative result (i.e., is it low enough or not). This realistic approach to understanding is beneficial in avoiding certain kinds of confusion and limitations in existing models and schemas. To stick with the thermal conductivity example, it is often given with a particular numerical value. This is also the case in the DICO ontology (Törmä and Zheng, online), where material properties are simply object properties (values) of the material class or instance. However, that is often not sufficient. While the quality (in the sense of the material makeup) may remain unchanged, the value we derive (how the quality is expressed) depends on several factors. For example, the thermal conductivity of a thermal insulation material we get on the datasheet (a declaration value measured according to a certain standard and in a physical state set up by that standard) is often not the one we need for calculation (but a design value representative of the state the insulation is during actual "use"), see (ISO, 2007). The difficulty we must address is that many qualities are functions (in the mathematical, not ontological sense) of several physical properties and, therefore, change over time depending on the state of the material entity in question. Continuing the example, the thermal conductivity is often a function of temperature and moisture content. We have not seen any AE or engineering ontologies directly addressing this problem, though the more general problem is well known (Grewe et al., 2016).

The difficulty of the solution in an ontology stems from the limitations of the OWL ontology language and its underlying semantics, which are based on Description Logic. OWL cannot express relations that have more than two terms: we can say that 'Material property a has value *b*', but we cannot express 'Material property a has value *b* at time *t* at temperature *x* and relative humidity *y*'. BCON proposes to utilise the concept of Stasis, as defined in the CCO Event ontology. A Stasis is defined as a process in which a specific independent entity (e.g., a Building Construction Object) remains in an unchanging condition. We can define a Stasis of Material Property that occurs during a specific period. If the measurement is performed during the same period, it can be inferred that the value we get is characteristic of that particular state of the material, see Fig. 13. If the corresponding physical properties are also specified quantitatively, we could encode complex material functions this way in terms of discrete values, but this is quite verbose, and leaves open the question what the material property is like between these discrete states (stasis). Further work is needed to encode the (mathematical) material function more directly.

Dispositions in BFO-like functions are realisable entities: they do not have to be realised all the time, but there must be a causal connection. Dispositions can only vary in strength, ranging from weaker dispositions that are exhibited only some of the time when the corresponding situation exists to the strongest ones, which are realised every time. An example in the AE domain can range from the weak disposition of heat-treated glass plates to spontaneously "explode" due to certain material impurities to the strong disposition of soft PVC membranes to become brittle and break when in direct contact with Polystyrene (due to a chemical reaction).

7.4.4 Processes

The functions of building constructions and their evaluation depend on numerous processes, as building con-

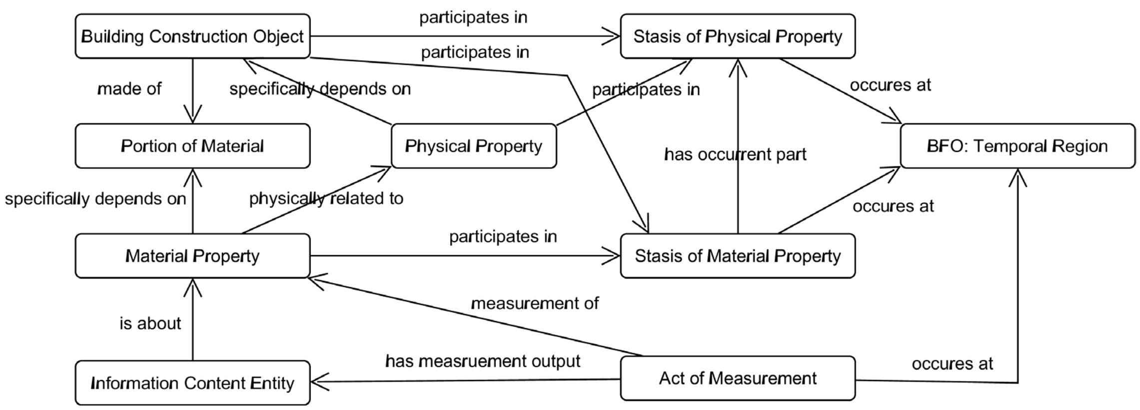

Fig. 13 Representation of a complex material property

structions are in constant dynamic interaction with one another and their environment. BCON will have to have a process ontology of at least the following types of processes: building processes (including maintenance processes and demolitions processes), processes related to the functioning and/or failure of the constructional (sub)-systems during their operational life and the measurement and evaluation processes to evaluate the functioning. While all processes are unique entities, they have many common characteristics. The BFO approach to treating such entities involves the use of Process Profiles, which serve as dimensions of comparison for processes. In the PSO ontology, Physical Behaviour is a subclass of Process Profile that demarcates a part of a process according to a physical law, such as thermal conduction in solids.

Many things that we usually treat as simple qualities or properties in models are, in fact, processes or are related to processes according to BFO. One such example is the thermal transmittance or U value of a wall. Thermal transmission is a physical process which only occurs in the presence of a temperature difference across a construction. The thermal transmittance, or U-value, is then a measurement for a specific type of this process: a stationary thermal transmission process (Fig. 14). It is not a quality in its own right, though it can be calculated from geometrical and material properties. Other similar process profiles include water penetration and water permeability, sound transmission, and sound insulation value, among others.

A further aspect of these processes that we did not elaborate on in this article is that whenever possible, a designer will want to use physical simulations to investigate them, so there needs to be a mapping between the reality-based classes and the classes needed for the simulation tools, see in Cheong and Butscher (2019).

7.4.5 Information entities

While our goal is a reality-oriented ontology, we also have information artefacts that do not directly correspond to real entities but are "about" some real entities, such as classification systems, component specifications (one of the main outputs of AE design), and model entities, like those in BIM models. Luckily, BFO has a way of dealing with these while clearly distinguishing them from other types of entities: generally dependent continuants. Good starting points in the form of domain-independent reference ontologies do exist, such as the CCO Information Entity Ontology and the OBO Information Artifact Ontology. The CCO has three main types of information entities: designative, prescriptive, and descriptive. BIM models and their entities are also information entities, but they often have elements of all three. Some examples of information entities are visualized in Fig. 15.

8 Case study: modelling a simple building construction detail

To demonstrate how modelling building constructions *via* BCON might look, let us consider a simple detail presented

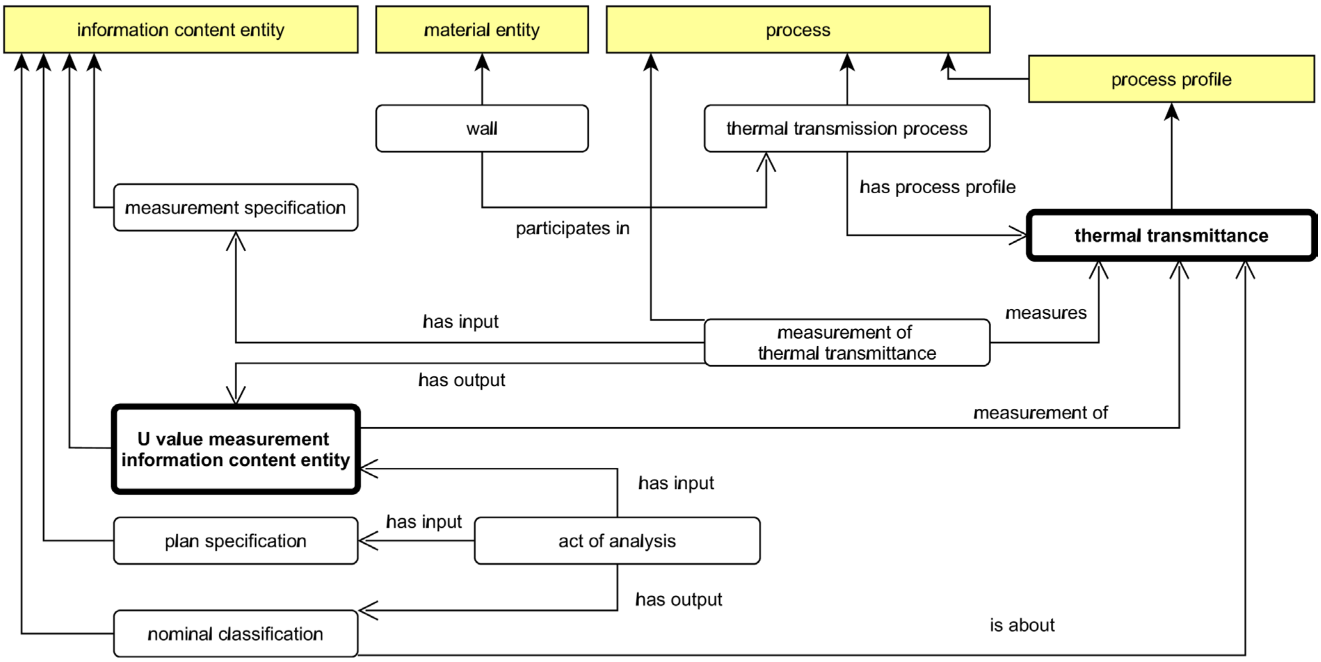

Fig. 14 The thermal transmittance and some of the associated classes and instances

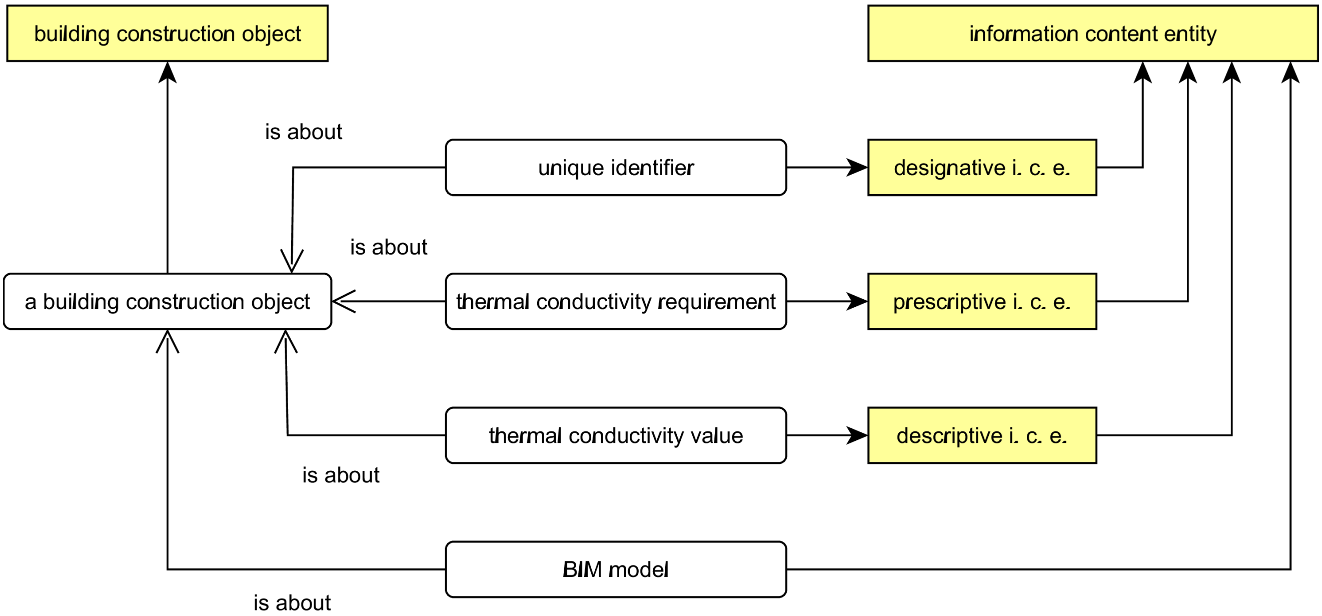

Fig. 15 Some different types of information content entities about a material object

in Fig. 16. It is an aerated clay masonry with an External Thermal Insulation Composite System (ETICS). It is a common type of external wall used in small to medium-sized residential or public buildings in Central Europe. The masonry is constructed from large aerated clay blocks featuring thin horizontal mortar layers and dry vertical tongue-and-groove joints, which reduce the amount of building work and enhance the thermal insulation of the finished masonry. As the masonry blocks are not solid (they have vertical cavities to improve thermal performance) and as they are placed next to each other without mortar, the masonry is in itself not airtight; an internal and external plaster layer is needed to make it so (the neglection of which is a common building mistake). The thermal insulation boards are glued in place with adhesive mortar and are additionally secured with low thermal conductivity dowels. The thermal insulation is finally covered with a special fibreglass mesh-reinforced composite plaster system.

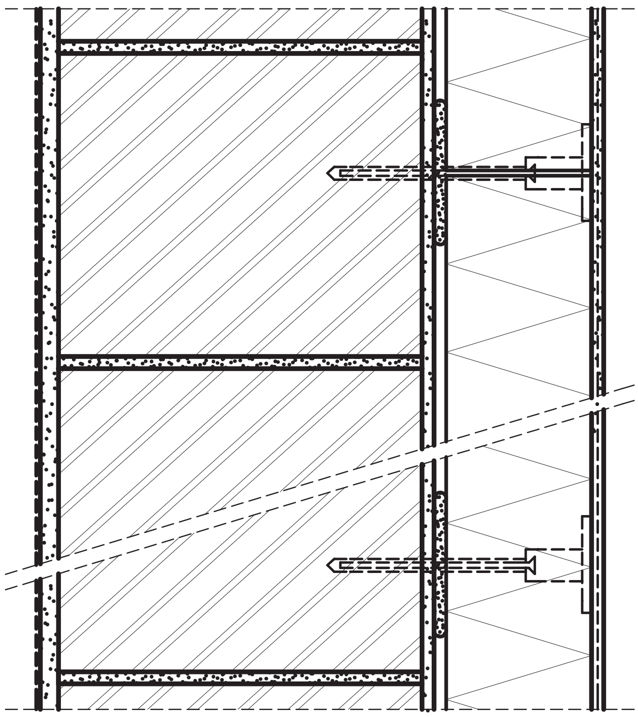

Fig. 16 Example detail: an aerated clay masonry wall with an external thermal insulation composite system

8.1 Modeling the detail with BCON

In a BIM application, such a wall is usually modelled as a layered construction or as a collection of separate objects without explicit connections. As such modelling is geometry-based, elements of very small size (like paint layers) and/or elements whose explicit representation would hugely increase modelling complexity and file size (like a large number of insulation dowels) are either ignored in the model and relegated to accompanying textual documentation or treated as properties on other elements ("has paint"). With BCON, it is possible to list all separate material entities and explicitly record their connectedness, thereby distinguishing between products and materials and construction objects. As our goal is only to demonstrate the modelling capabilities, we made some simplifications to the construction, limited ourselves to only some types of entities, and listed only a subset of each type. Some of the resulting entities are listed in Tables 1 and 2, along with some of their relationships in Table 3.

This kind of representation is quite verbose and entirely intractable by hand, but the same is true of BIM schemas

Table 1 The major material entities

BCON class	Instances
Product	Aerated clay masonry block
	Thermal Insulation board
	Thermal insulation dowel
(Product) component	Thermal insulation plaster fibreglass mesh
	Thermal Insulation dowel plastic shell
	Thermal Insulation dowel metal screw
Portion of material	Portion of masonry dry-mix mortar
	Portion of masonry mortar (as built)
	Portion of interior dry-mix plaster
	Portion of interior plaster (as built)
	Portion of exterior dry-mix plaster
	Portion of exterior plaster (as built)
	Portion of interior wall paint
	Portion of interior wall paint (as built)
	Portion of thermal Insulation adhesive dry-mix mortar
	Portion of thermal insulation adhesive mortar (as built)
	Portion of thermal insulation base plaster dry-mix mortar
	Portion of thermal insulation finishing plaster dry-mix mortar
	Thermal insulation composite plaster (as built)
Prefabricated Building Construction Object	Masonry
	Thermal insulation layer
	Thermal insulation dowel
In situ Building Construction Object	Internal paint layer
	Internal plaster layer
	Horizontal mortar joint
	External airtight and levelling base plaster layer
	Thermal insulation adhesive blob
	Thermal insulation fibreglass reinforced composite plaster layer
Object	Airgap between plaster and thermal insulation (air as fluid)
Fiat Object Surface	Internal surface (of internal plaster layer)
	External surface (of external plaster layer)
Fiat Connection Surface	Paint-to-plaster connection
	Plaster-to-masonry connection
	Masonry block to masonry mortar connection
	Masonry block to masonry block connection
	Adhesive mortar to plaster connection
	Adhesive mortar to thermal insulation connection
	Thermal insulation to thermal insulation connection
	Dowel to thermal insulation connection
	…
Building Construction System	Load bearing system
	External envelope system
	Thermal insulation system
	Sound insulation system
	Airtightness system
	…
Environmental system	Internal environment system
	External environment system

Table 2 The major processes

BCON class	Instances
Act of Building Construction (A.o.B.C.)	A.o.B.C. that created the masonry A.o.B.C. that created the internal plaster layer …
Physical Behaviour	Thermal transmittance process in wall Moisture transport process in wall …

Table 3 The major relations (object properties) between material entities

Instance1	BCON relation	Instance2
Masonry	Made of	Masonry block
Masonry mortar layer	Made of	Portion of masonry mortar (as built)
Interior plaster layer	Made of	Portion of interior plaster (as built)
Exterior plaster	Made of	Portion of exterior plaster (as built)
	…	
Interior plaster layer	Made from	Portion of interior plaster dry mix
Interior plaster layer	Made from	Portion of water
	…	
A.o.B.C. that created the internal plaster layer	Has material input	Portion of interior plaster dry mix
A.o.B.C. that created the internal plaster layer	Has construction object input	Masonry layer
A.o.B.C. that created the internal plaster layer	Has construction object output	Internal plaster layer
A.o.B.C. that created the internal plaster layer	Has connection output	Internal plaster-to-masonry connection
A.o.B.C. that created the internal plaster layer	Occupies temporal region	Some temporal region
	…	
Masonry mortar layer	Continuant part of	Masonry
Fibreglass mesh	Continuant part of	Thermal insulation fibreglass reinforced composite plaster layer
Masonry	Continuant part of	Loadbearing system
Masonry	Continuant part of	External envelope system
	…	
Paint to plaster connection	Connects from	Internal plaster layer
Paint to plaster connection	Connects to	Internal paint layer
	…	
Internal environment system	Is environment of	Internal surface (of internal plaster layer)
	…	

with less information. Dedicated computer programs can be developed, and some information can even be deduced from partial inputs. The main point is that we can build models rich in AE design-relevant information. With a suitable computer tool, most entities would not have to be entered manually.

8.2 Modeling background information with BCON

Thus far, we have only stated the material composition and connectedness of the detail, as well as some limited information about its construction. However, there is additional knowledge related to building constructions in general and about the type of construction in our example more specifically. Some knowledge can be encoded directly into the ontology *via* class hierarchies, relationships, and axiomatisation. For example, we could define the class of aerated clay masonry Building Construction Objects as not having any connection (without any intermediate layers) to the external environment (as this would quickly destroy such bricks). Modelling an exposed brick wall would clearly raise an error. However, we cannot necessarily encode directly some more complex information, such as "an aerated clay masonry block if it is used as part of the external envelope or with sound insulation requirement must be provided with some layer that ensures its airtightness." This is a limitation of the expressivity of the formal logical system behind the OWL ontology language we are using.

Unfortunately, the design of building constructions is relying heavily on statements of such or higher complexity. Today such statements are found in many different places: laws, standards, guidelines, manuals, textbooks, and personal experience; written (or spoken) in many different natural human languages using incompatible terminologies. Even if we had a comprehensive database of such statement keyword searches would often not be enough due to the aforementioned lack of a unified terminology and the extreme context dependence of the information. This makes it very difficult to filter and deliver the most relevant knowledge to the designer. Fortunately, while an ontology cannot contain all relevant knowledge directly in its structure, it can be a basis for building specialised tools that can. This is the principal benefit of ontologies in any scientific field, which is best highlighted by their use in the field of biomedical research. When an ontological system provides a unified and unambiguous terminology, the terms and connections can be used to annotate data, as well as articles and other natural language sources. The resulting linked database of information is much more suitable for widescale knowledge sharing and queries.

To demonstrate this capability, let us look at background knowledge highly relevant to our example detail:

External thermal insulation composite systems (ETICS) require a certain level of surface evenness on the planar constructions they are built upon. Aerated clay masonry walls often exhibit surface irregularities if not constructed properly. This irregularity must then be evened out by the external plaster layer, which is added to them (under the ETICS). The necessary thickness of the external plaster layer is, therefore, dependent on the surface evenness of the masonry.

While the BCON proto-ontology does not yet have enough expressivity to describe all of this, and the specific rules and regulations where this is laid out in detail are not important right now, we can encode a part of the information in a graph: Fig. 17. There is already quite a lot of information even in this small fragment, though it is formulated in terms of instances. In OWL, object properties (relationships) only apply between instances of classes. Classes can be described in terms of their relationships (class restrictions), which can be one way to encode universal "rules" about the various constructions. In other ontology standards, such as OBO, we can also define relationships between classes to achieve a similar effect.

9 Some possible applications of realist ontological models of building constructions

The principles for building a realist ontology dictate that the structure of the ontology should not be based on any

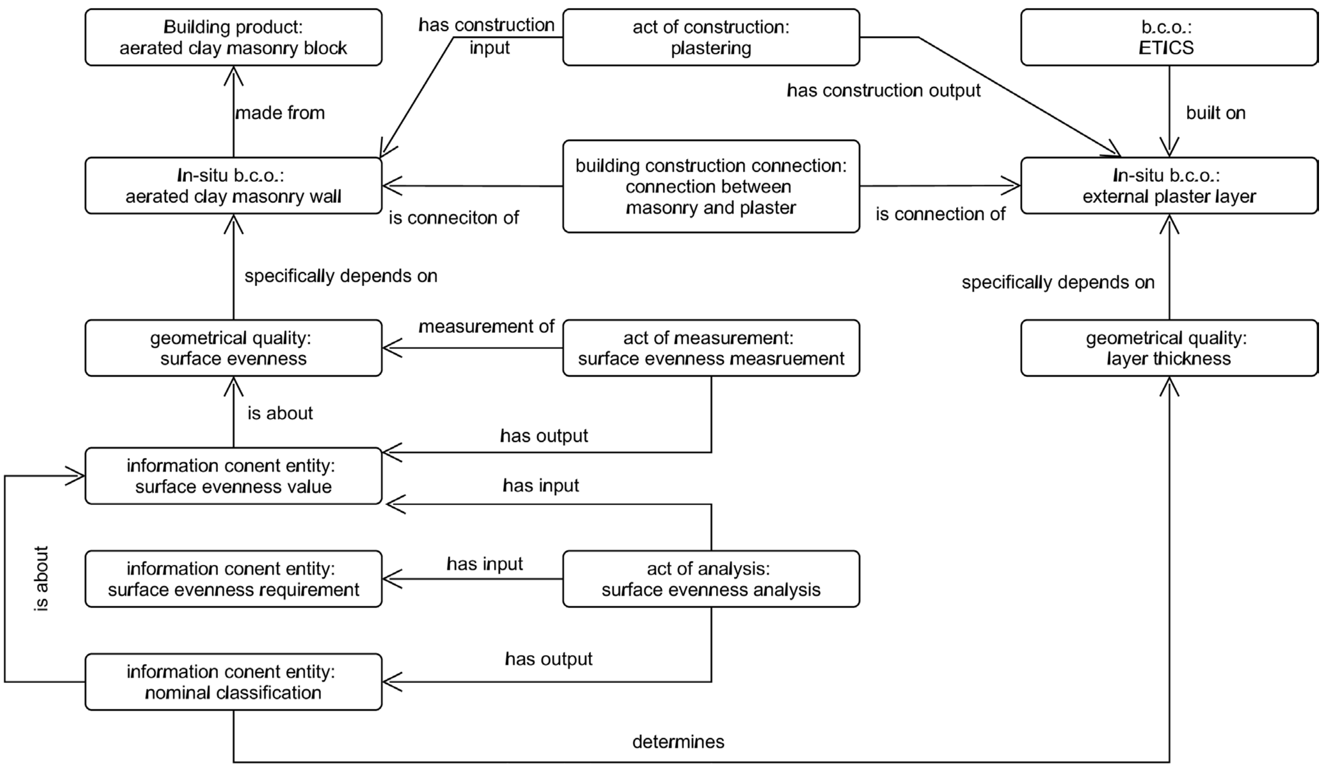

Fig. 17 A simple graph encoding relevant background knowledge about building construction objects

specific application with its possibly idiosyncratic conceptualisations. Nevertheless, the end goal is to create applications. After presenting the small case study in Section 8.1, we can now better elucidate at least some of the possible applications to aid the various tasks around the AE design of building constructions, that could greatly benefit from a realist ontology of building constructions.

9.1 Modeling

9.1.1 Increased expressivity

Even with the small proto-ontology presented here, we can express important information about building construction details that would be very hard in many BIM schemas, such as:

- the precise connection between batches of material used for construction, construction objects and the materials of the construction objects;
- the connectedness of Building Construction Objects, the type of these connections and their properties;
- the prerequisites (in terms of existing construction objects) of specific acts of building;
- the functions of various components and systems;
- etc.

9.1.2 Flexible custom schemas

While the core ontologies must be carefully curated by a well-defined, responsible organisation, given the modular ontology structure, users can also create their own purpose-built extensions as a particular project or company ontology. This can extend far beyond mere custom property sets in terms of expressivity, and the resulting model would still be understandable to others, at least with the information present in the base ontology system. The extension ontology could also be easily distributed together with the actual model.

9.1.3 Interoperability in general

By separating a construction object from its qualities and dispositions (already defined in other interoperable ontologies), we are in a much better position to create representations of cross-domain applicability, as opposed to having very specific data models with complex and domain-specific types and numerous data fields.

9.1.4 Automated inference

An ontology-supported model enables us to infer knowledge that is not explicitly stated (*via* the axiomatisation of the ontology, most of which is not presented in this article). For our small case study in Section 8.1, we could infer, for example, that:

- the batches of material/building product used to create *in situ* Building Construction Objects, like the plaster layers, are not identical to these;
- the products used to create prefabricated Building Construction Objects are identical to part of those objects (like the bricks inside the masonry);
- the connectedness of Building Construction Objects, beyond the direct connections stated explicitly;
- the approximate way how specific Building Construction Objects support each other (the general direction of load transfer);
- the building order of Building Construction Objects;
- etc.

9.2 Cooperation

The way the model is built using Semantic Web Technologies makes it very easy to share data *via* the internet and link to other pieces of information on the web. Not all relevant information about the model needs to reside in a single silo (e.g., BIM file), and therefore not all participants need to edit the same file to add information to the same entity.

9.3 Engineering analysis

9.3.1 Reality oriented representation

Basing the representation on a reality-oriented upper ontology brings significant benefits, with the philosophically grounded and well-tested ability to distinguish between real-world entities and information entities. Engineering analysis often relies on simulation tools with their virtual entities (such as meshes and boundary conditions) that stand in an "is about" relationship to the actual physical entities. This explicit relationship enables us to describe the mapping between physical entities and the inputs and outputs of simulation software, as demonstrated in Cheong and Butscher (2019).

9.3.2 Interoperability of simulation software

Such an approach can also serve to connect different simulation software with each other *via* ontology, as also demonstrated by Cheong and Butscher (2019).

9.4 Model checking

9.4.1 Querying

The proper structure of the model also allows for the retrieval of much relevant information. In the case study presented in Section 8.1, we could run queries such:

- to find whether there are discontinuities in the thermal insulation layer (if any);

- to find out whether there are any unplastered aerated clay masonry surfaces;
- to find out whether two objects of specific material composition are in contact somewhere (e.g., to check for problems of contact corrosion).

9.4.2 Rules

Based on the structure of the ontology-based representation, we can also create logical rules that cannot be directly included in the ontology itself. These rules could be used to perform design checking on the model.

9.5 Knowledge sharing

9.5.1 Ontological annotations

As briefly introduced and demonstrated in Section 8.1, databases and even natural language sources could be annotated based on the ontology. Searching for research findings, raw design data, product documentation, and reports about building faults in linked databases built this way would be much more efficient than current approaches (Costa and Lima, 2014; Hill et al., 2008).

9.5.2 Model sharing

Even in the world of contemporary BIM models, we can find a wealth of downloadable content to use in our models and save work. However, in terms of AE design, these are very static, and finding appropriate objects is no easy task. Collections of ontology-based models and parts, whether within a company or through public services, would contain much more relevant information, allowing for the automated retrieval and analysis of existing solutions based solely on the description of a problem.

9.5.3 Computer-aided AE design

Ultimately, the most important application would be a union of all previous points: an integrated system of computer-aided AE design. Solutions for the different design tasks should benefit from each other and the same representation system without the need for manual or ambiguous and lossy transformations.

10 Conclusions and future work

We proposed that solving the knowledge representation problem is fundamental in creating better digital design tools for the AE design of building constructions. There is a need for a formal language with a set of features that we believe is best suited for a formal ontology. After reviewing existing works, we did not find any ontological works that focus specifically on the task or could be easily extended with a specialised module, although several are related and are prime candidates for later alignment. Instead, we proposed developing a new system of ontologies for AE design, called BCON, based on the BFO upper ontology and its accompanying methodology and ontological realism. BCON aims to establish a new framework for describing and analysing building constructions, a system that is interoperable with, yet free from, some of the legacy problems of existing BIM technologies.

The research and ontology development are in their early phases, but as we move forward, we intend to establish an organisation for their development, evaluation, curation, and maintenance. More specialised expert knowledge is required from both academia and the AEC industry for the project to succeed. A prototype core proto-ontology is presented in this article. Currently, it is a clean-sheet design, and it only contains the most upper-level terms, which need to be further extended and modularised. By its very nature, AE design deals with many adjoining engineering and scientific disciplines that are both absolutely required for AE design but fall outside the scope of what BCON should ideally cover directly if ontologies are to work together without constantly redefining everything. Further study is needed to determine which related ontologies BCON is to be aligned with in this regard. The DICO and the IOF are two obvious candidates. While BCON aims to cover the key terms of building constructions, there is a need for more specialised application ontologies in the various subfields of AE design, such as waterproofing, thermal insulation, and acoustic insulation.

In addition to theoretical work, the ontology must be evaluated through practical use; therefore, we intend to create demo applications and recruit test users from the relevant profession to provide feedback. We also believe that the long-term adoption of ontologies is determined by the applications that use them, so the theoretical work and application development must proceed hand in hand. Ultimately, our vision with BCON is to create a framework for modelling building constructions and knowledge sharing that is not separate from the actual design thinking of practising engineers, but can be an integral part of it.

Acknowledgement

We thank Dr Balázs Nagy (Budapest University of Technology and Economics) and Géza Kapovits (Budapest University of Technology and Economics) for their contributions to the workshops in the early stages of the research, and Balázs Fürtön (Budapest University of Technology and Economics) for his comments on the manuscript.

References

Abanda, F. H., Tah, J. H. M., Keivani, R. (2013) "Trends in built environment semantic Web applications: Where are we today?", Expert Systems with Applications, 40(14), pp. 5563–5577. https://doi.org/10.1016/j.eswa.2013.04.027

Afsari, K., Eastman, C. M. (2016) "A Comparison of Construction Classification Systems Used for Classifying Building Product Models", presented at 52nd ASC Annual International Conference Proceedings, Provo, UT, USA, Apr. 13–16. https://doi.org/10.13140/RG.2.2.20388.27529

Arp, R., Smith, B., D. Spear, A. D. (2015) "Building Ontologies with Basic Formal Ontology", The MIT Press. ISBN 9780262527811

Autodesk "Revit for Structural Engineering, (2026)", [computer program] Available at: https://www.autodesk.com/products/revit/structural [Accessed: 25 July 2025]

Bakonyi, D., Dobszay, B. "BCON-Proto, (4.0)", [computer program] Available at: https://github.com/dbakonyi/BCON-Proto [Accessed: 26 July 2025]

Bandrowski, A., Brinkman, R., Brochhausen, M., Brush, M. H., Bug, B., Chibucos, M. C., Clancy, K., …, Zheng, J. (2016) "The Ontology for Biomedical Investigations", PLoS ONE, 11(4), e0154556. https://doi.org/10.1371/journal.pone.0154556

Bednar, T., Bothe, D., Forster, J., Fritz, S., Gladt, M., Handler, C., Haufe, N., …, Ziegler, M. (2020) "SIMULTAN - Simultane Planungsumgebung für Gebäudecluster in resilienten, ressourcen- und höchst energieeffizienten Stadtteilen" (Simultaneous planning environment for building clusters in resilient, resource-efficient, and highly energy-efficient urban areas), Bundesministerium für Verkehr, Innovation und Technologie, Vienna, Austria, Rep. 3. [online] Available at: http://hdl.handle.net/20.500.12708/39593 [Accessed: 13 March 2025] (in German)

Beetz, J., Coebergh van den Braak, W., Botter, R., Zlatanova, S., de Laat, R. (2014) "Interoperable data models for infrastructural artefacts – a novel IFC extension method using RDF vocabularies exemplified with quay wall structures for harbors", In: Proceedings of the 10th European Conference on Product and Process Modelling (ECPPM 2014), Vienna, Austria, pp 135–136. ISBN 978-1-138-02710-7 [online] Available at: https://resolver.tudelft.nl/uuid:bb9a7dff-52c7-4aaf-a6b8-898432270620 [Accessed: 11 March 2025]

Beetz, J., van Leeuwen, J., de Vries, B. (2009) "IfcOWL: A case of transforming EXPRESS schemas into ontologies", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 23(1), pp. 89–101. https://doi.org/10.1017/S0890060409000122

Berners-Lee, T., Hendler, J., Lassila, O. (2001) "The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities", Scientific American, 284(5), pp. 34–43.

Boje, C., Guerriero, A., Kubicki, S., Rezgui, Y. (2020) "Towards a semantic Construction Digital Twin: Directions for future research", Automation in Construction, 114, 103179. https://doi.org/10.1016/j.autcon.2020.103179

Bonino, D., Corno, F. (2008) "DogOnt - Ontology Modeling for Intelligent Domotic Environments", In: 7th International Semantic Web Conference (ISWC 2008), Karlsruhe, Germany, pp. 790–803. ISBN 978-3-540-88564-1 https://doi.org/10.1007/978-3-540-88564-1_51

Borgo, S., Lesmo, L. (2008) "The Attractiveness of Foundational Ontologies in Industry", In: Proceedings of the 3rd Formal Ontologies Meet Industry (FOMI 2008), Torino, Italy, pp. 1–9. ISBN 978-1-58603-871-7 https://doi.org/10.3233/978-1-58603-871-7-1

Borst, P., Akkermans, H., Pos, A., Top, J. (1995) "The PhysSys Ontology for Physical Systems", In: Ninth International Workshop on Qualitative Reasoning '95, Amsterdam, Netherlands, pp. 11–21. [online] Available at: https://www.researchgate.net/profile/Jan-Top/publication/40149124_The_PhysSys_Ontology_for_Physical_Systems/links/00b7d524c36e839402000000/The-PhysSys-Ontology-for-Physical-Systems.pdf [Accessed: 11 March 2025]

BSI (2008) "BS 6100-6:2008 Building and civil engineering. Vocabulary - Construction parts", British Standards Institution, London, UK.

Building Smart Alliance "buildingSMART Data Dictionary (bSDD)", [online] Available at: https://www.buildingsmart.org/users/services/buildingsmart-data-dictionary/ [Accessed: 13 March 2025]

Buttigieg, P. L., Morrison, N., Smith, B., Mungall, C. J., Lewis, S. E. (2013) "The environment ontology: contextualising biological and biomedical entities", Journal of Biomedical Semantics, 4(1), 43. https://doi.org/10.1186/2041-1480-4-43

Chairman of the Joint Chiefs of Staff (2011) "DOD Dictionary of Military and Associated Terms", [pdf] U.S. Army, The Pentagon Arlington County, VA, USA, Available at: https://www.tradoc.army.mil/wp-content/uploads/2020/10/AD1029823-DOD-Dictionary-of-Military-and-Associated-Terms-2017.pdf [Accessed: 26 January 2022]

Chávez-Feria, S., Villalón, M. P. "Building ontology", [online] Available at: https://bimerr.iot.linkeddata.es/def/building/ [Accessed: 11 March 2025]

Cheong, H., Butscher, A. (2019) "Physics-based simulation ontology: an ontology to support modelling and reuse of data for physics-based simulation", Journal of Engineering Design, 30(10–12), pp. 655–687. https://doi.org/10.1080/09544828.2019.1644301

Conspectus Inc. "Uniformat", [online] Available at: https://www.conspectusinc.com/uniformat [Accessed: 11 March 2025]

ConSteel Solutions Ltd. "Consteel, (18)", [computer program] Available at: https://consteelsoftware.com/products/consteel/ [Accessed: 25 July 2025]

Costa, R., Lima, C. (2014) "Knowledge representations with ontology support for collaborative engineering in architecture engineering and construction", Journal of Information Technology in Construction (ITcon), 19, pp. 434–461. [online] Available at: http://www.itcon.org/2014/26 [Accessed: 11 March 2025]

Council of the European Union (2011) "Regulation (EU) No. 305/2011 harmonised conditions for the marketing of construction products", Brussels, Belgium. [online] Available at: https://eur-lex.europa.eu/legal-content/EN/LSU/?uri=celex:32011R0305 [Accessed: 11 March 2025]

Creative Commons "Creative Commons BY 4.0 License", [online] Available at: https://creativecommons.org/licenses/by/4.0/ [Accessed: 11 March 2025]

CSI "About Omniclass", [online] Available at: https://www.csiresources.org/standards/omniclass/standards-omniclass-about [Accessed: 11 March 2025]

CSI "Masterformat", [online] Available at: https://www.csiresources.org/standards/masterformat [Accessed: 11 March 2025]

Cutarelli, S. (2024) "Historical Architecture and BIM Modelling: Between Representation of Reality and Conceptual Abstraction", Periodica Polytechnica Architecture, 55(2–3), pp. 109–120. https://doi.org/10.3311/PPar.37895

Darlington, M. J., Culley, S. J. (2008) "Investigating ontology development for engineering design support", Advanced Engineering Informatics, 22(1), pp. 112–134. https://doi.org/10.1016/j.aei.2007.04.001

de Farias, T. M., Roxin, A.-M., Nicolle, C. (2015) "IfcWoD, Semantically Adapting IFC Model Relations into OWL Properties", In: Proceedings of the 32nd CIB W78 Conference on Information Technology in Construction, Eindhoven, Netherlands, pp. 175–185. [online] Available at: https://www.researchgate.net/publication/280047339_IfcWoD_Semantically_Adapting_IFC_Model_Relations_into_OWL_Properties#fullTextFileContent [Accessed: 11 March 2025]

Dimyadi, J., Pauwels, P., Spearpoint, M., Clifton, C., Amor, R. (2015) "Querying a Regulatory Model for Compliant Building Design Audit", In: Proceedings of the 32nd CIB W78 Conference, Eindhoven, Netherlands, pp. 139–148. https://doi.org/10.13140/RG.2.1.4022.6003

Dinis, F. M., Poças Martins, J., Guimarães, A. S., Rangel, B. (2022) "BIM and Semantic Enrichment Methods and Applications: A Review of Recent Developments", Archives of Computational Methods in Engineering, 29(2), pp. 879–895. https://doi.org/10.1007/s11831-021-09595-6

Dobszay, G. (2011) "Building constructions of stone cladded roofs in contemporary architecture", Facta Universitatis - Series: Architecture and Civil Engineering, 9(1), pp. 35–56. https://doi.org/10.2298/FUACE1101035D

Dobszay, G., Nemes, R., Andriska, F., Heincz, D., Kovács, K., Reisch, R., Simon, T. K. (2019) "Performance of adhesive waterproofing as regards of lateral water filtration", Materials Structures Technology, 2(1), pp. 77–93. https://doi.org/10.31448/mstj.02.01.2019.77-93

El-Diraby, T. (2019) "Beyond e-permitting: Framing the Business Case for Automated Rule Checking in AEC in the Era of Big Data", buildingSMART International, Kings Langley, UK, Rep. TR 1012. [online] Available at: https://www.buildingsmart.org/wp-content/uploads/2020/06/buildingSMART-RR-TR1012-Framing-the-Business-Case-for-Automated-Rule-Checking-v1.1-Final-Dec-2019.pdf [Accessed: 11 March 2025]

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, Rhodes, Greece, pp. 361–377. ISBN 978-3-319-26148-5 https://doi.org/10.1007/978-3-319-26148-5_24

Fernald, H., Hong, S., O'Brien, W., Bucking, S. (2018) "BIM to BEM translation workflows and their challenges: a case study using a detailed BIM model", In: Proceedings of eSim 2018: the 10th Conference of IBPSA-Canada, Montreal, Canada, pp. 482–491. ISBN 978-2-921145-88-6 [online] Available at: https://publications.ibpsa.org/conference/paper/?id=esim2018_2-3-A-3 [Accessed: 11 March 2025]

Ferron, W., Turkan, Y. (2019) "Toward a Roadmap for BIM Adoption and Implementation by Small-Sized Construction Companies", In: Proceedings of the 35th CIB W78 2018 Conference: IT in Design, Construction, and Management, Chicago, IL, USA, pp. 873–879. ISBN 978-3-030-00220-6 https://doi.org/10.1007/978-3-030-00220-6_105

Garcia-Lopez, N. P. (2017) "An activity and flow-based construction model for managing on-site work", PhD Thesis, Stanford University. [online] Available at: https://purl.stanford.edu/jw488vx5801 [Accessed: 11 March 2025]

Gero, J. S. (1990) "Design Prototypes: A Knowledge Representation Schema for Design", AI Magazine, 11(4), pp. 26–36. https://doi.org/10.1609/aimag.v11i4.854

Gero, J. S., Kannengiesser, U. (2014) "The Function-Behaviour-Structure Ontology of Design", In: Chakrabarti, A., Blessing, L. T. M. (eds.) An Anthology of Theories and Models of Design: Philosophy, Approaches and Empirical Explorations, Springer, London, pp. 263–283. ISBN 978-1-4471-6338-1 https://doi.org/10.1007/978-1-4471-6338-1_13

Gholizadeh, P., Esmaeili, B., Goodrum, P. (2018) "Diffusion of Building Information Modeling Functions in the Construction Industry", Journal of Management in Engineering, 34(2), 04017060. https://doi.org/10.1061/(asce)me.1943-5479.0000589

Giannakis, G. I., Katsigarakis, K. I., Lilis, G. N., Rovas, D. V. (2019) "A Workflow for Automated Building Energy Performance Model Generation Using BIM Data", In: Proceedings of the 16th IBPSA Conference., Rome, Italy, pp. 167–174. ISBN 978-1-7750520-1-2 https://doi.org/10.26868/25222708.2019.210689

Golbreich, C., Horridge, M., Horrocks, I., Motik, B., Shearer, R. (2007) "OBO and OWL: Leveraging Semantic Web Technologies for the Life Sciences", In: 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference (ISWC 2007 + ASWC 2007), Busan, Korea, pp. 169–182. ISBN 978-3-540-76298-0 https://doi.org/10.1007/978-3-540-76298-0_13

Gruber, T. "What is an Ontology?", [online] Available at: http://www.ksl.stanford.edu/kst/what-is-an-ontology.html [Accessed: 11 March 2025]

Grewe, N., Hastings, J., Jansen, L., Ruttenberg, A., Mungall, C., Schulz, S. (2016) "Expressing time-dependent relations through temporal qualifications", [pdf] Studylib. Available at: https://studylib.net/doc/6849748/1-basic-formal-ontology-bfo [Accessed: 11 March 2025]

Gürsel Dino, İ. (2012) "Creative Design Exploration by Parametric Generative Systems in Architecture", METU Journal of the Faculty of Architecture, 29(1), pp. 207–224. https://doi.org/10.4305/METU.JFA.2012.1.12

Habibi, S. (2017) "The promise of BIM for improving building performance", Energy and Buildings, 153, pp. 525–548. https://doi.org/10.1016/J.ENBUILD.2017.08.009

Hagedorn, T. J., Smith, B., Krishnamurty, S., Grosse, I. (2019) "Interoperability of disparate engineering domain ontologies using basic formal ontology", Journal of Engineering Design, 30(10–12), pp. 625–654. https://doi.org/10.1080/09544828.2019.1630805

Hill, D. P., Smith, B., McAndrews-Hill, M. S., Blake, J. A. (2008) "Gene Ontology annotations: what they mean and where they come from", BMC Bioinformatics, 9(5), S2. https://doi.org/10.1186/1471-2105-9-S5-S2

Hogan, W. R., Hicks, A., Donovan, K. M., Brochhausen, M. "Geographical Entity Ontology", [online] Available at: https://github.com/mcwdsi/geographical-entity-ontology [Accessed: 11 March 2025]

Holnapy, D., Rédey, G. (1993) "Experiences in Expert System Development", Periodica Polytechnica Civil Engineering, 37(1), pp. 13–30. [online] Available at: https://pp.bme.hu/ci/article/view/3808 [Accessed: 11 March 2025]

Hubexo North UK Ltd. "uniclass – Unified Construction Classification", [online] Available at: https://www.thenbs.com/our-tools/uniclass [Accessed: 11 March 2025]

Inndata Datentechnik GmbH "freeClass – die freie Klassifikationsstruktur" (freeClass - the free classification system), [online] Available at: https://www.freeclass.eu/ [Accessed: 11 March 2025] (in German)

ISO (2004) "ISO 10303-11:2004 Industrial automation systems and integration — Product data representation and exchange: Part 11: Description methods: The EXPRESS language reference manual", International Organization for Standardization, Geneva, Switzerland.

ISO (2007) "ISO 10456:2007 Building materials and products — Hygrothermal properties — Tabulated design values and procedures for determining declared and design thermal values", International Organization for Standardization, Geneva, Switzerland.

ISO (2018) "ISO 16739-1:2018 Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries: Part 1: Data schema", International Organization for Standardization, Geneva, Switzerland.

ISO/IEC (2018) "ISO/IEC 24707:2018 Information technology — Common Logic (CL) — A framework for a family of logic-based languages", International Organization for Standardization / International Electrotechnical Commission, Geneva, Switzerland.

ISO/IEC (2021a) "ISO/IEC 21838-1:2021 Information technology — Top-level ontologies (TLO): Part 1: Requirements", International Organization for Standardization / International Electrotechnical Commission, Geneva, Switzerland.

ISO/IEC (2021b) "ISO/IEC 21838-2:2021 Information technology — Top-level ontologies (TLO): Part 2: Basic Formal Ontology (BFO)", International Organization for Standardization / International Electrotechnical Commission, Geneva, Switzerland.

Jabi, W., Aish, R., Lannon, S., Chatzivasileiadi, A., Wardhana, N. M. (2018) "Topologic - A toolkit for spatial and topological modelling", In: Proceedings of the 36th Education and Research in Computer Aided Architectural Design in Europe Conference (eCAADe 2018), Lodz, Poland, pp. 449–458. ISBN 978-94-91207-16-7 https://doi.org/10.52842/conf.ecaade.2018.2.449

Kitamura, Y. (2006) "Roles of Ontologies of Engineering Artifacts for Design Knowledge Modeling", In: Proceedings of the 5th International Seminar and Workshop Engineering Design in Integrated Product Development (EDIProD 2006), Gronów, Poland, pp. 59–69. ISBN 83-7481-045-9 [online] Available at: file:///C:/Users/user/Downloads/DS37_59-69_Kitamura.pdf [Accessed: 11 March 2025]

Kitamura, Y., Sano, T., Namba, K., Mizoguchi, R. (2002) "A functional concept ontology and its application to automatic identification of functional structures", Advanced Engineering Informatics, 16(2), pp. 145–163. https://doi.org/10.1016/S1474-0346(02)00006-X

Koskela, L. J. (1999) "Management of Production in Construction: A Theoretical View", In: 7th Annual Conference of the International Group for Lean Construction, Berkeley, CA, USA, pp. 241–252. ISBN 1929465017 [online] Available at: https://www.researchgate.net/publication/2323248_Management_Of_Production_In_Construction_A_Theoretical_View [Accessed: 11 March 2025]

Lee, D.-Y., Chi, H., Wang, J., Wang, X., Park, C.-S. (2016) "A linked data system framework for sharing construction defect information using ontologies and BIM environments", Automation in Construction, 68, pp. 102–113. https://doi.org/10.1016/j.autcon.2016.05.003

Lilis, G. N., Giannakis, G. I., Rovas, D. V. (2015) "Detection and Semi-automatic Correction of Geometric Inaccuracies in IFC Files", In: Proceedings of BS2015: 14th Conference of International Building Performance Simulation Association, Hyderabad, India, pp. 2182–2189. ISBN 978-93-5230-118-8 [online] Available at: https://publications.ibpsa.org/proceedings/bs/2015/papers/bs2015_2581.pdf [Accessed: 11 March 2025]

Lin, J., Fox, M. S., Bilgic, T. (1996) "A Requirement Ontology for Engineering Design", Concurrent Engineering, 4(3), pp. 279–291. https://doi.org/10.1177/1063293X9600400307

Lima, C., El-Diraby, T., Stephens, J. (2005) "Ontology-based optimisation of knowledge management in e-construction", Journal of Information Technology in Construction, 10, pp. 305–327. [online] Available at: http://www.itcon.org/2005/21 [Accessed: 11 March 2025]

McCrae, J. P., Abele, A., Buitelaar, P., Cyganiak, R., Jentzsch, A., Andryushechkin, V., Debattista, J., Nasir, J. "The Linked Open Data Cloud", [online] Available at: https://lod-cloud.net/ [Accessed: 11 March 2025]

Mizoguchi, R., Kitamura, Y. (2009) "A Functional Ontology Of Artifacts", The Monist, 92(3), pp. 387–402. https://doi.org/10.5840/monist200992322

Munn, K., Smith, B. (eds.) (2008) "Applied Ontology: An Introduction", Ontos Verlag. ISBN 978-3-938793-98-5 https://doi.org/10.1515/9783110324860

National Center for Ontological Research "BFO-Based Engineering Ontologies", [online] Available at: https://ncorwiki.buffalo.edu/index.php/BFO-Based_Engineering_Ontologies [Accessed: 11 March 2025]

Niknam, M., Karshenas, S. (2017) "A shared ontology approach to semantic representation of BIM data", Automation in Construction, 80, pp. 22–36. https://doi.org/10.1016/J.AUTCON.2017.03.013

Open Biological and Biomedical Ontologies Foundry "The OBO Flat File Format Guide, version 1.4", [online] Available at: https://owlcollab.github.io/oboformat/doc/GO.format.obo-1_4.html [Accessed: 11 March 2025]

Otte, J. N., Kiritsi, D., Ali, M. M., Yang, R., Zhang, B., Rudnicki, R., Rai, R., Smith, B. (2019) "An ontological approach to representing the product life cycle", Applied Ontology, 14(2), pp. 179–197. https://doi.org/10.3233/AO-190210

Paskaleva, G., Mazak-Huemer, A., Wimmer, M., Bednar, T. (2021) "Leveraging integration facades for model-based tool interoperability", Automation in Construction, 128, 103689. https://doi.org/10.1016/j.autcon.2021.103689

Pauwels, P. "Building Element Ontology, (v0.1.0)", [computer program] Available at: https://pi.pauwel.be/voc/buildingelement/index-en.html [Accessed: 11 March 2025]

Pauwels, P., Roxin, A. (2016) "SimpleBIM: from full ifcOWL graphs to simplified building graphs", In: 11th European Conference on Product and Process Modelling (ECPPM 2016), Limassol, Cyprus, pp. 11–18. ISBN 978-1-138-03280-4 [online] Available at: http://hdl.handle.net/1854/LU-8041826 [Accessed: 11 March 2025]

Pauwels, P., Terkaj, W. (2016) "EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology", Automation in Construction, 63, pp. 100–133. https://doi.org/10.1016/j.autcon.2015.12.003

Pauwels, P., Van Deursen, D., Verstraeten, R., De Roo, J., De Meyer, R., Van de Walle, R., Van Campenhout, J. (2011) "A semantic rule checking environment for building performance checking", Automation in Construction, 20(5), pp. 506–518. https://doi.org/10.1016/j.autcon.2010.11.017

Pratt, M. J. (2001) "Introduction to ISO 10303—the STEP Standard for Product Data Exchange", Journal of Computing and Information Science in Engineering, 1(1), pp. 102–103. https://doi.org/10.1115/1.1354995

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: – Interoperability, Usability, Applicability, 12(1), pp. 143–161. https://doi.org/10.3233/SW-200385

Rudnicki, R. (2019) "An Overview of the Common Core Ontologies", [pdf] CUBRC, Inc., Buffalo, NY, USA. [online] Available at: https://www.nist.gov/system/files/documents/2021/10/14/nist-ai-rfi-cubrc_inc_004.pdf [Accessed 26 January 2022]

Rudolph, S., Schneider, M. (2011) "Utility and Feasibility of Reasoning beyond Decidability in Semantic Technologies", In: 19th International Conference on Conceptual Structures (ICCS 2011), Derby, UK, pp. 19–34. ISBN 978-3-642-22688-5 https://doi.org/10.1007/978-3-642-22688-5_2

Sacks, R., Girolami, M., Brilakis, I. (2020) "Building Information Modelling, Artificial Intelligence and Construction Tech", Developments in the Built Environment, 4, 100011. https://doi.org/10.1016/j.dibe.2020.100011

Schulze-Kremer, S., Smith, B. (2005) "Ontologies for the life sciences", In: Jorde, L. B., Little, P. F. R., Dunn, M. J., Subramaniam, S. (eds.) Encyclopedia of Genetics, Genomics, Proteomics and Bioinformatics, John Wiley & Sons. ISBN 9780470849743

Seppälä, S., Ruttenberg, A., Smith, B. (2017) "Guidelines for writing definitions in ontologies", Ciencia da Informacao, 46(1), pp. 73–88. [online] Available at: https://philpapers.org/archive/SEPGFW.pdf [Accessed: 11 March 2025]

Smith, B. (2017) "Overview of BFO for the Industrial Ontologies Foundry", Available at: https://www.youtube.com/watch?v=glmdyYbEjy-c&list=PLyngZgI13WTiUmcchT_Zf2peiEZwx72G_&index=5 [Accessed: 11 March 2025]

Smith, B., Goldstein, A., Ruttenberg, A., Goldfain, A., Peters, B., Torniai, C., Mungal, C., …, He, Y. O. "OBO Metadata Ontology", [online] Available at: https://obofoundry.org/ontology/omo.html [Accessed: 11 March 2025]

Smith, B., Ruttenberg, A., Goldfain, A., Duncan, B., Peters, B., Mungall, C., Osumi-Sutherland, D., …, He, Y. O. "BasicFO-2020", [online] Available at: https://github.com/bfo-ontology/BFO-2020 [Accessed: 11 March 2025]

Smith, B., Ameri, F., Cheong, H., Kiritsis, D., Sormaz, D., Will, C., Otte, J. N. (2019) "A First-Order Logic Formalization of the Industrial Ontologies Foundry Signature Using Basic Formal Ontology", presented at 2019 Joint Ontology Workshop (JOWO 2019), Graz, Austria, Sept. 23–25. [online] Available at: https://ceur-ws.org/Vol-2518/paper-FOMI6.pdf [Accessed: 11 March 2025]

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., …, Lewis, S. (2007) "The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration", Nature Biotechnology, 25(11), pp. 1251–1255. https://doi.org/10.1038/nbt1346

Smith, B., Ceusters, W. (2010) "Ontological realism: A methodology for coordinated evolution of scientific ontologies", Applied Ontology, 5(3–4), pp. 139–188. https://doi.org/10.3233/AO-2010-0079

Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., Mungall, C., Neuhaus, F., Rector, A. L., Rosse, C. (2005) "Relations in biomedical ontologies", Genome Biology, 6(5), R46. https://doi.org/10.1186/gb-2005-6-5-r46

Smith, B., Kusnierczyk, W., Schober, D., Ceusters, W. (2006) "Towards a Reference Terminology for Ontology Research and Development in the Biomedical Domain", In: KR-MED 2006 "Biomedical Ontology in Action", Baltimore, MD, USA, pp. 57–65. [online] Available at: https://www.researchgate.net/publication/221426466_Towards_a_Reference_Terminology_for_Ontology_Research_and_Development_in_the_Biomedical_Domain [11 March 2025]

Software Development Company "AXISVM, (X8)", [computer program] Available at: https://axisvm.eu/axisvm-eng/ [Accessed: 25 July 2025]

Solibri Inc. "Solibri Model Checker, (V9.9.5)", [computer program] Available at: https://www.solibri.com/ [Accessed: 11 March 2025]

Solihin, W., Dimyadi, J., Lee, Y.-C., Eastman, C., Amor, R. (2020) "Simplified schema queries for supporting BIM-based rule-checking applications", Automation in Construction, 117, 103248. https://doi.org/10.1016/j.autcon.2020.103248

Spear, A. D., Ceusters, W., Smith, B. (2016) "Functions in Basic Formal Ontology", Applied Ontology, 11(2), pp. 103–128. https://doi.org/10.3233/AO-160164

Stals, A., Jancart, S., Elsen, C. (2021) "Parametric modeling tools in small architectural offices: Towards an adapted design process model", Design Studies, 72, 100978. https://doi.org/10.1016/j.destud.2020.100978

Steel, J., Drogemuller, R., Toth, B. (2012) "Model interoperability in building information modelling", Software & Systems Modeling, 11(1), pp. 99–109.
https://doi.org/10.1007/s10270-010-0178-4

StruSoft "FEM-Design, (24.00.003)", [computer program] Available at: https://strusoft.com/software/3d-structural-analysis-software-fem-design/ [Accessed: 25 July 2025]

Šormaz, D., Sarkar, A., Terkaj, W. (2020) "Progress on IOF's Process and Production Planning Reference Ontology", prestented at Proceedings of the Workshops of I-ESA 2020, Tarbes, France, Nov. 17–19. [online] Available at: https://ceur-ws.org/Vol-2900/WS5Paper3.pdf [Accessed: 11 March 2025]

Szőts, M., Simonyi, A. (2009) "A jól szerkesztett mérnöki ontológiákról" (On well structured engineering ontologies), In: Magyar Számítógépes Nyelvészeti Konferencia (6.), Szeged, Hungary, pp. 162–171. ISBN 978-963-482-982-9 (in Hungarian) [online] Available at: https://acta.bibl.u-szeged.hu/58706/1/msznykonf_006_162-171.pdf [Accessed: 11 March 2025] (in Hungarian)

Talebi, S., Tzortzopoulos, P., Koskela, L., Poshdar, M., Tommelein, I. D., Tezel, A., Antunes, R. (2019) "A Vision for the Future of the Computer-Aided Tolerance Management in Construction Based on the Lessons Learned from Manufacturing", In: Proceedings of the CIB World Building Conference 2019: Constructing Smart Cities. International Council for Research and Innovation in Building and Construction, Hong Kong, China, pp. 926–935. ISBN 978-962-367-821-6 [online] Available at: https://www.researchgate.net/publication/338359405_A_Vision_for_the_Future_of_the_Computer-Aided_Tolerance_Management_in_Construction_Based_on_the_Lessons_Learned_from_Manufacturing [Accessed: 11 March 2025]

Tchouanguem, J. F., Karray, M. H., Foguem, B. K., Magniont, C., Abanda, F. H., Smith, B. (2021) "BFO-based ontology enhancement to promote interoperability in BIM", Applied Ontology, 16(4), pp. 453–479.
https://doi.org/10.3233/AO-210254

The Gene Ontology Consortium (2019) "The Gene Ontology Resource: 20 years and still GOing strong", Nucleic Acids Research, 47(D1), pp. D330–D338.
https://doi.org/10.1093/nar/gky1055

Tirmizi, S. H., Aitken, S., Moreira, D. A., Mungall, C., Sequeda, J., Shah, N. H., Miranker, D. P. (2011) "Mapping between the OBO and OWL ontology languages", Journal of Biomedical Semantics, 2(1), S3.
https://doi.org/10.1186/2041-1480-2-S1-S3

Törmä, S., Zheng, Y. "Digital Construction Information", [online] Available at: https://digitalconstruction.github.io/Information/v/0.5/index-en.html [Accessed: 11 March 2025]

Trimble Inc. "Tekla Model checker suite, (24.1.1.0)", [computer program] Available at: https://support.tekla.com/help/tekla-structures/not-version-specific/model_checker_suite [Accessed: 11 March 2025]

Valluru, P., Karlapudi, J., Menzel, K., Mätäsniemi, T., Shemeikka, J. (2020) "A Semantic Data Model to Represent Building Material Data in AEC Collaborative Workflows", In: 21st IFIP WG 5.5 Working Conference on Virtual Enterprises (PRO-VE 2020), Valencia, Spain, pp. 133–142. ISBN 978-3-030-62412-5
https://doi.org/10.1007/978-3-030-62412-5_11

Venugopal, M., Eastman, C. M., Teizer, J. (2015) "An ontology-based analysis of the industry foundation class schema for building information model exchanges", Advanced Engineering Informatics, 29(4), pp. 940–957.
https://doi.org/10.1016/j.aei.2015.09.006

Wagner, A., Sprenger, W., Maurer, C., Kuhn, T. E., Rüppel, U. (2022) "Building product ontology: Core ontology for Linked Building Product Data", Automation in Construction, 133, 103927.
https://doi.org/10.1016/j.autcon.2021.103927

Wang, H.-H., Boukamp, F., Elghamrawy, T. (2010) "Ontology-Based Approach to Context Representation and Reasoning for Managing Context-Sensitive Construction Information", Journal of Computing in Civil Engineering, 25(5), pp. 331–346.
https://doi.org/10.1061/(ASCE)CP.1943-5487.0000094

World Wide Web Consortium "OWL 2 Web Ontology Language Document Overview (Second Edition)", [online] Available at: https://www.w3.org/TR/owl2-overview/ [Accessed: 11 March 2025]

World Wide Web Consortium "RDF 1.1 Primer", [online] Available at: https://www.w3.org/TR/rdf11-primer/ [Accessed: 11 March 2025]

Zhang, C., Beetz, J., de Vries, B. (2018) "BimSPARQL: Domain-specific functional SPARQL extensions for querying RDF building data", Semantic Web: – Interoperability, Usability, Applicability, 9(6), pp. 829–855.
https://doi.org/10.3233/SW-180297

Zhang, X., Di, R., Liang, Y. (2010) "Ontology Based Knowledge Modeling for Structural Engineering Experiment Information Management", In: 2010 Ninth International Conference on Grid and Cloud Computing, Nanjing, China, pp. 40–45. ISBN 978-1-4244-9334-0
https://doi.org/10.1109/GCC.2010.21

Zheng, J. (2020) "Information Artifact Ontology", [online] Available at: https://obofoundry.org/ontology/iao.html [Accessed: 04 February 2022]

Zheng, Y., Törmä, S., Seppänen, O. (2021) "A shared ontology suite for digital construction workflow", Automation in Construction, 132, 103930.
https://doi.org/10.1016/j.autcon.2021.103930