

Semantic representation of Design for Manufacturing and Assembly offsite housing

Edlira Vakaj^{a,*}, Franco Cheung^a, Abdel-Rahman H. Tawil^a, Panagiotis Patlakas^a and Jianpeng Cao^b

^a Faculty of Computing, Engineering and Build Environment, Birmingham City University, United Kingdom

^b Institute of Construction and Infrastructure Management, ETH Zurich, Stefano-Franscini-Platz 5, Zurich 8093, Switzerland

E-mail: Edlira.Vakaj@bcu.ac.uk

E-mails: Franco.Cheung@bcu.ac.uk, Abdel-Rahman.Tawil@bcu.ac.uk, Panagiotis.Patlakas@bcu.ac.uk, cao@ibi.baug.ethz.ch

Abstract. Architecture, Engineering and Construction (AEC) is a fragmented industry dealing with heterogeneous data formats coming from different domains and, despite efforts such as Building Information Modelling (BIM), the gap in information exchange is often major. This challenge is particularly evident in the rapidly emerging field of Design for Manufacturing and Assembly (DfMA), which deviates from typical construction methodologies. Semantic web technologies are recognized for overcoming challenges of information exchange in isolated domains in many fields, via the publication of standardized linked data that are highly discoverable and machine processable. While ontologies have been proposed for manufacturing processes in general, this work is the first to apply semantic web technologies in the DfMA domain, supporting its integration to typical AEC workflows. A new domain ontology, Offsite Housing Ontology (OHO), is presented. OHO facilitates the semantic integration of offsite construction knowledge, enabling it to be used in DfMA practice. It semantically defines offsite construction domain terminology and relationships, describing a core vocabulary. This supports a unified model, required for efficient collaborative design management, while improving existing data flows. The efficiency and effectiveness of the OHO approach is demonstrated in a real-world DfMA scenario through the development of a Knowledge Based Engineering tool to automate cost estimation. As OHO is extensible, this approach can be adapted and extended to accommodate a very wide range of offsite housing, delivering important optimization and automation benefit from DfMA solutions.

Keywords: Offsite Construction, Ontology Engineering, Building Information Modeling, DfMA, Linked Building Data, Software Development

1. Introduction

Offsite construction aims to standardise and automate production processes in a factory environment through applying manufacturing design concepts [1]. To construct offsite effectively and efficiently, it requires products to be designed systematically through adopting Design for Manufacturing and Assembly (DfMA), a philosophy and design methodology [2]. DfMA evaluates and

improves the design by considering downstream processes of manufacturing and assembly in the early design stage [3]. DfMA is comprised of Design for Manufacturing (DfM) and Design for Assembly (DfA). DfM targets the selection of materials that minimise wastage, optimise processes and subprocesses, optimise parts and systems and fulfill tolerance requirements [4], whereas DfA focuses on minimising the number of modules for assembly and optimizing the assembly process [5]. In practice,

*Corresponding author. E-mail: Edlira.Vakaj@bcu.ac.uk.

DfMA contains a list of design guidelines [5–7]. An example of the guidelines can be avoiding undercuts or other features which require special operations and/or tools.

Although it is common practice to use Building Information Modelling (BIM) for design and production of information, DfMA has not been widely applied in building design due to the lack of information and structured knowledge [8, 9], big data sources [10], and documented sources of information on process modularisation [11, 12]. There are efforts to develop DfMA frameworks, e.g. the synthesis of construction-oriented DfMA guidelines [13], but there is not yet an approach to systematise DfMA knowledge that enables designers to evaluate modular design. One main obstacle to apply DfMA in designing modules is the lack of tools that evaluate modular design, typically requiring input from several professionals with expertise in off-site production, costing and scheduling. Current practice relies largely on either heuristics or broad estimates, without an analysis of the relations and interactions of processes and sub-processes to the activity level [14]. For DfMA to be applied widely, an appropriate data modelling approach and corresponding data models need to be developed.

In this regard, this study first reviewed the current use of BIM and identified that there is a lack of manufacturing information in BIM models. To integrate manufacturing information into BIM, a data exchange is required. Semantic web technologies and Linked data has proved to facilitate data exchange from heterogeneous data sources in the construction industry [15]. However, through a literature review on current ontologies, we found that no ready-to-use and similar solutions can be adopted to integrate building design, product and production knowledge. Therefore, a new Web Ontology Language (OWL) domain ontology (i.e. Offsite Housing Ontology (OHO)) is proposed here to represent the domain knowledge of DfMA. The ontology developed was used to build an online knowledge-based engineering tool for estimating cost for modular house construction to demonstrate the use of OHO. In the end, the OHO was discussed to accommodate a wide range of offsite products and support other functions, such as the optimization of modular production.

2. Limitations and potentials of data models for DfMA

The introduction of BIM provides an opportunity to structurally embed data in relation to processes and other attributes within a three-dimensional (3D) building model [16, 17]. For instance, it is common to have process-related data, typically used for the scheduling (i.e. 4D BIM), and cost data, typically for estimating of building costs (i.e. 5D BIM), integrated to the BIM model. While the data for scheduling are process-related, they refer mainly to activities onsite. Cost estimating during design development, on the other hand, does not use process-related data at all [18].

Although there are tools attempting to integrate the process and cost data to enable simultaneous 4D and 5D modelling, their application in practice is limited due to data and disciplinary silos [19]. The use of DfMA offers an opportunity to break the silos as the manufacturing and assembly processes are defined in the design development stage. Unlike how geometric information is kept and represented in BIM models, there is a lack of manufacturing concepts defined in the models, e.g. manufacturing processes in relation to particular modular design, which increases dramatically the data exchange requirements.

Although Industry Foundation Classes (IFC) [20], a standard platform-neutral schema, is widely used in practise to support representations in semantic web formats (e.g., OWL), its data structure does not represent production processes enough [21]. For instance, the product classification for DfMA design is based on the product sub-assemblies, which are generally under-represented in the IFC schema. Also, the schema does not represent the processes of manufacturing production. Thus, the use of the IFC schema alone will miss the chance to implement DfMA that requires knowledge representation of the production and assembly attributes as well as cost [2]. For instance, setting a competitive price is found as one main challenge for prefabricated house builders [22].

3. Related works: Ontologies and Data Models

Semantic web technologies allow the development of a formal representation of information, irrespective of the adopted tool. They have been used in the AEC domain for heterogeneous data formats from different sources and domains [23], supporting flexible

1 data exchange and distributed data management, and
2 providing a basis for logical inference using rules
3 and ontologies [24]. **DfMA in construction has been**
4 **interpreted from three perspectives: (1) a holistic**
5 **product design strategy; (2) an evaluation system for**
6 **the efficiency of manufacturing and assembly process;**
7 **and (3) prefabrication and modular construction**
8 **technologies [25]. To cover the concepts of those three**
9 **perspectives, as well as the cost analysis, four different**
10 **domains of ontologies have been reviewed, including**
11 **building design and construction ontologies, product**
12 **ontologies, production ontologies and cost-estimation**
13 **ontologies. The limitations of these ontologies for**
14 **the purpose of DfMA implementation have also been**
15 **identified.**

16 3.1. Building Design and Construction Ontologies

17
18
19 The conceptual schema for IFC is defined using
20 EXPRESS, a data specification language [26]. Using
21 IFC, data from a BIM model can be exchanged
22 between heterogeneous software applications.
23 However, IFC itself is not a web-compliant standard.
24 The use of semantic web technologies [27], including
25 OWL and the Resource Description Framework (RDF)
26 resolves this limitation and can make IFC data widely
27 available and accessible over the web. The IFC schema
28 has been converted into an ontology in the OWL
29 (i.e. ifcOWL) to make it usable with semantic web
30 technologies. This makes it possible, for instance, to
31 make inferences using Description Logic (DL) rules
32 [28].

33 The latest version of IFC scheme, namely IFC4
34 consists of 1293 classes and 1572 object properties.
35 The complexity and the counter-intuitiveness with
36 Semantic Web Principles [29] of this ontology
37 makes reasoning and management very hard and
38 inefficient, and inevitably, increases the need to
39 develop separate modules based on the existing
40 core IFC modules. An implementation of a modular
41 IFC ontology was proposed [30] and has initiated
42 several research initiatives, such as the W3C Linked
43 Building Data (LBD) Community Group, that focus
44 on how to modularize the IFC ontology to improve its
45 extensibility.

46 BIM adoption has been an industrial interest
47 and various countries and communities have their
48 trajectories to map BIM's development (e.g. Digital
49 Built Britain in the UK [31]). **To support the**
50 **communication inter-disciplinarily, the domain**
51 **ontologies for BIM are supposed to be extendable**

1 **according to the concept of Linked Building Data**
2 **(LBD) [32]. In this regard, the Building Topology**
3 **Ontology (BOT) forms a small part of a broader**
4 **concept of LBD, in which additional domain**
5 **ontologies can be proposed to further extend the core**
6 **of BOT, e.g. with building element classifications and**
7 **properties. The current version of BOT (i.e. version**
8 **0.3.1) consists of 7 classes, 14 object properties, and**
9 **one datatype property. The ontology is adopted to**
10 **answer the competency questions (CQs) relating to the**
11 **concepts of Zone, Site, Building, Storey, Space and**
12 **Elements [33]. BOT has seven main classes; they are**
13 **bot:Zone, bot:Element, bot:Interface,**
14 **bot:Building, bot:Site, bot:Space and**
15 **bot:Story. BOT is aligned with other ontologies,**
16 **such as BRICK (A uniform metadata schema for**
17 **buildings [34]) and SAREF (Smart Applications**
18 **REference) [35]. It can easily be combined with**
19 **ontologies that describe products, IoT devices or**
20 **sensor observations.**

21 A construction knowledge management ontology,
22 Domain Ontology for Construction Knowledge was
23 introduced in 2013, DOCK 1.0 [36]. DOCK 1.0 is a
24 traditional construction management specific ontology
25 that provides defined key concepts in construction
26 (such as actor, resource, product and state). Similarly,
27 the BIM4EE project has very recently proposed
28 digital construction ontologies, DiCon, as shared
29 representations of construction domain knowledge
30 that describes construction activities [37]. The aim
31 of DiCon is to integrate heterogeneous construction
32 workflow information to support site management
33 tasks such as assessing site environmental conditions,
34 quality inspection and scheduling of works. Although
35 DiCon has the class of activity and attempts to
36 develop process-based knowledge, the knowledge
37 representation and relationships defined are for the
38 onsite context, which is not suitable for modelling the
39 knowledge for offsite activities.

40 3.2. Product Ontologies

41
42
43 Building domain specific ontologies have the
44 capacity to include more building related information.
45 For instance, the SolConPro ontology describes multi-
46 functional building products with high level of details.
47 **However, due to a lack of modularity, the reuse of the a**
48 **part of the ontology might need to import non-related**
49 **functionalities, thus cause inefficiency to link to other**
50 **ontologies. [38].**
51

There were initiatives by the W3C LBD group to translate IFC to product related machine-readable ontologies in RDF format, PRODUCT and PROPS. The PRODUCT ontology consisted of the IFC classes underneath the IfcElement node, while the PROPS ontology included all properties defined in the IFC property sets. **Both are simple ontologies and do not support complex building structure modeling, which contains assemblies.**

The Building Product Ontology (BPO) is a multi-layered product ontology, which was designed to overcome these shortcomings. It allows products to be defined according to the assembled components, entity interconnections, component and product properties. To facilitate modular construction, BPO needs to incorporate further ontologies in order to facilitate various types of evaluations, e.g. on cost and time.

3.3. Production Ontologies

The Manufacturing's Semantics Ontology (MASON) is a manufacturing upper ontology that establishes a common semantic vocabulary in the manufacturing domain [39]. It is organized in three main concepts: Entities, Operations and Resources. Entities give an abstract view of the product in terms of geometric features, raw material and cost entities. Operations describe all processes linked to manufacturing including machining operations, control or assembly and logistic operations, human operations and launching operations as well. Resources classify all resources linked to manufacturing such as geographic resources, machine tools, human resources etc. [40]. While the ontology is a standard for the manufacturing domain, its application to modular construction products is limited.

Extending from a MASON-based approach, An et al. [41] developed extended ontologies for wood frame assemblies. The authors identified the limitation of using the BIM model alone to explain how wood frame is assembled for production, i.e. it does not include information about how the product is fabricated. Three classes are created, they are product, operation and resource. The ontology developed helps to identify the required manufacturing operations for wood panel frame construction although the ontology is not detailed enough to perform cost and time estimations and to explain the sequence in which a product is assembled.

3.4. Cost Estimation Ontologies

BIM technologies are used in cost estimation making the cost estimation process more efficient, mainly because standard machine-readable formats can be used in combination with standard measurement methods to automate the construction cost estimation. BIM-based cost estimation tools, such as CostX and BIMMeasure, are designed to streamline the quantity measurement processes, with some degree of automation built-in in those tools. However, data models are yet to have process-related data incorporated and thus, those tools have not been developed to be able to populate or extract process related data, an aspect that is crucial for implementing DfMA, i.e. to facilitate simultaneous design evaluation.

Differing from traditional approach to estimate cost that demands individual professionals to either measure or extract quantities, link quantities with unit rates and present cost report, the use of ontology can automate the estimation processes. The use of semantic web technologies has been proposed for almost 20 years. However, the existing ontologies for cost estimation are either building component-based (e.g. elemental costing) [42–44] or resource-based [45, 46]. Either of them has little account for processes. For instance, Abanda et al. proposed a cost estimating ontology based on the New Rules of Measurement 1 (NRM 1) [44]. Its data structure breaks down a building into elements, sub-elements, group elements and components and has NRM 1-based rules of reasoning, to enhance reasoning. Liu et al. [45] adopted a semantic approach using Autodesk Quantity Take-Off (QTO) and the Revit data model for the preparation of a light-frame building estimate. A construction-oriented product ontology was defined and used to provide an ontology-augmented BIM model in RDF. Neither of the ontologies proposed is suitable for implementing DfMA. Other researchers have looked at more specific problems, such as automating the costing of tiling works [46], labour costs [45] and labour content [47]. These studies are small in scope, and do not allow for generalisation, especially for modular production concepts in DfMA.

3.5. Research Gap

DfMA is aimed to capture the constraints in the manufacturing process in the design process. An ontology for DfMA needs to contain common

1 building vocabularies, offsite activities, manufacturing
2 processes and concepts, and cost aspects. By
3 comparing the existing ontologies (in Table 1), It
4 shows that while existing ontologies capture product
5 concepts, none of them can be applied directly without
6 further ontology development, particularly for most in
7 the aspects of manufacturing processes and concepts.
8 For example, BOT and BPO only capture the building
9 element classifications and properties, but do not
10 include semantics for offsite manufacturing. Also,
11 they are mainly developed to support a product-based
12 or component-based design approach, typically not
13 supporting the consideration of the resources, activities
14 and processes in production directly. DOCK 1.0 and
15 DiCon develop semantics for processes of construction
16 works but are largely based on a site management
17 context.

18 Manufacturing-oriented methodologies, such as
19 MASON, fail to capture the richness and complexity
20 of buildings. Finally, existing cost estimation
21 ontologies adopted for producing estimates are
22 building component-based or resource-based rather
23 than process-based. The latter is needed to assess the
24 cost implications for offsite systems, as individual
25 systems have corresponding production methods,
26 which will incur cost differently [48, 49].

27 A DfMA-specific ontology (i.e. OHO) was
28 developed in this study to develop the semantics that
29 represent manufacturing processes and concepts as
30 well as the associated cost. Key terms and relationships
31 in relation to production and assembly, are explicitly
32 defined. OHO has concepts that are matched to those
33 in BOT and BPO, which can be connected to both
34 ontologies through general associations rather than
35 strict formal connections. For instance, a `oho:House`
36 `is-a bot:Building` or when the concepts are used
37 together, they can be linked or distinguished using
38 computer-readable relationships, such as `sameAs` and
39 `differentFrom`. A more detailed description of
40 each of these classes is given in Section 5. OHO
41 can complement BOT and BPO by adding new
42 offsite construction knowledge (terminology and
43 relationships), defining specific core vocabulary that
44 can be used in DfMA practice.

46 4. Method

47 4.1. Ontology engineering method

48 The methodology used to design OHO follows
49 best practices for defining such domain ontologies,
50
51

1 including the use of competency questions, re-using
2 existing ontological resources (e.g. process maps for
3 production), creating links to other ontologies (e.g.
4 ifcOWL), and using competency questions to assess
5 the ontology. The development of OHO followed
6 the NeOn methodology framework, which is a
7 scenario-based methodology aiming to speed up the
8 development of ontologies through reusing existing
9 ontologies, non-ontological resources and ontology
10 design patterns [44]. There are various models
11 comprising various phases and scenarios to implement
12 NeOn. The Six-Phase Waterfall Ontology Network
13 Life Cycle Model was chosen as it incorporates 2
14 phases, i.e. reuse and re-engineering phases that are
15 very relevant to the context of OHO, in addition to
16 the base 4-phase model comprising initiation, design,
17 implementation and maintenance phases. For each
18 phase, NeOn has a number of scenarios (9 in total)
19 for users to define the tasks for individual phases.
20 Figure 1 shows the phases and the corresponding
21 scenarios chosen according to NeOn framework for
22 OHO development.

23 A summary of the Ontology Requirements
24 Specification Document (ORSD) was produced
25 (Table 2) in the *Initiation* phase. Then, an analysis
26 of non-ontological resources such as BIM model,
27 manufacturing process reports etc. was done in the
28 *Reuse* phase. In the *Re-engineering* phase, a literature
29 review was conducted on: i) existing ontologies based
30 on IFC; ii) existing ontologies for building design,
31 construction, and related domains; iii) concepts and
32 relations in addition to what was identified in i) and
33 ii). The review identified a lot of shortcomings for
34 DfMA implementation and concluded that the existing
35 ontologies do not represent offsite construction
36 sufficiently.

37 Then, a set of competency questions were drafted
38 with reference to the guide for developing an ontology
39 from Stanford University [50] in the *Design* phase.
40 The competency questions guided the discussion with
41 the stakeholders and experts of the offsite house
42 production case involved, who are the architect, the
43 production engineer, site engineer, steel manufacturer,
44 client and cost consultant (Hereafter, the term 'group
45 of experts' is used to refer to this set of respondents).
46 There were 4 rounds of group discussions and a
47 number of one-to-one interviews. The competence
48 questions were identified and agreed with the group
49 of experts in the first meeting. Then, a meeting was
50 carried out to agree on the data structure for the
51 various OHO modules. One to one interviews were

Table 1
DfMA requirement and native support in existing ontologies.

Existing Ontologies	DfMA requirements				
	Building design and construction	Product structure	Manufacturing and assembly processes	Cost estimation	Complexity
ifcOWL	Yes	Yes	No	No	High
BOT	Yes	No	No	No	Low
BPO	Yes	Yes	No	No	Low
MASON	No	No	Yes	Yes	Medium
PRODUCT / PROPS	Partially	No	No	No	Low
Liu et al [47]	Yes	No	No	Yes	Very Low
DiCon	Partially	No	No	No	Medium
DOCK 1.0	Partially	No	No	No	-

Six-Phase Waterfall Ontology Network Lifecycle

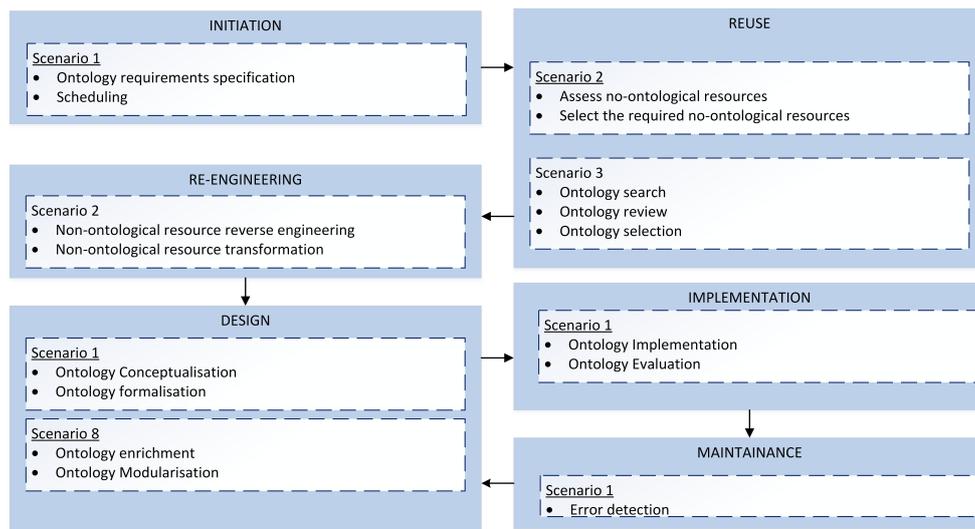


Fig. 1. Phases for development of OHO ontology

then conducted to clarify and confirm the concepts and relationships defined in the ontology individually with the relevant experts that hold the expertise in which OHO attempts to represent. The OHO ontology were then reviewed and refined in the last two group meetings. Finally, the OHO ontology was formalized in a machine understandable format using the OWL language in the *Implementation* phase.

5. OHO Overview

OHO is a DfMA domain ontology, which complements existing ontologies. It defines a model of categories within the offsite construction manufacturing Universe of Discourse (UoD), plus sufficient knowledge about those categories to allow process reasoning and automatic cost calculation. OHO is intended as a Common Reference (CORE) model for offsite construction. The proposed ontological model is language independent, using the broader term ‘terminology’ for a semantic model

Table 2
Ontology Requirement Specification Document

Scope:

The ontology focuses on the products and components of a modular house designed using DfMA (hereafter named as DfMA house), the process of manufacturing and assembling products, the resources consumed, and their cost and carbon emissions quantitative impact.

Intended Users:

- Modular Manufacturers
- Architects
- Engineers
- Specifiers
- Quantity Surveyors
- Domain Experts

Intended Use:

- Use Case 1: Cost estimation (demonstrated in this paper)
- Use Case 2: Life cycle carbon emissions (future work)
- Use Case 3: Semantic digital twin of DfMA (future work)
- Use Case 4: Time monitoring using sensor information (future work)
- Use Case 5: Production waste classification (future work)

Ontology Requirements:

Group of Competency Questions defined in the *OHO Competency Questions* section.

linked to the offsite construction manufacturing domain.

At the core of the OHO ontology, a limited number of very high level concepts is needed, such as OHO:Production. This high level schema is a prerequisite for good integration with other data models. This principle is similar to the principle behind BOT as a central core ontology [51]. This OHO core is illustrated in Figure 2 and it responds to the following requirements:

1. Fits closely with building standards especially in applications for design and manufacturing assembly or in the retrieval and classification of OHO concepts.
2. Sufficiently general to be used in different applications for decision support and interoperability.
3. Formally defined in OWL Description Logic (DL)
4. Acts as a general-purpose modelling language for offsite manufacturing.

5. Supports OWL-DL reasoners to allow core OHO concepts to be combined to create new descriptions of classes and instances constructed according to constraints implemented within the ontology.
6. Supports intuitive and practical collaboration between different groups, being easily understood and application independent.

Furthermore, the OHO core has two domains of specialisation, which are presented here as two modules that extend the core. An ontology module is a reusable component of a larger or more complex ontology [52], which is self-contained but has connections and associations with other ontology modules, and can be viewed as an extension of the original ontology [53]. OHO-Pro represents the production module (Production Section), allows the definition of production line data, and imports the OHO core. OHO-Cost, the costing module (OHO Costing Module Section), facilitates the cost estimation of a DfMA house. The OHO-Core, OHO-Pro and OHO-Cost are linked via explicitly defined

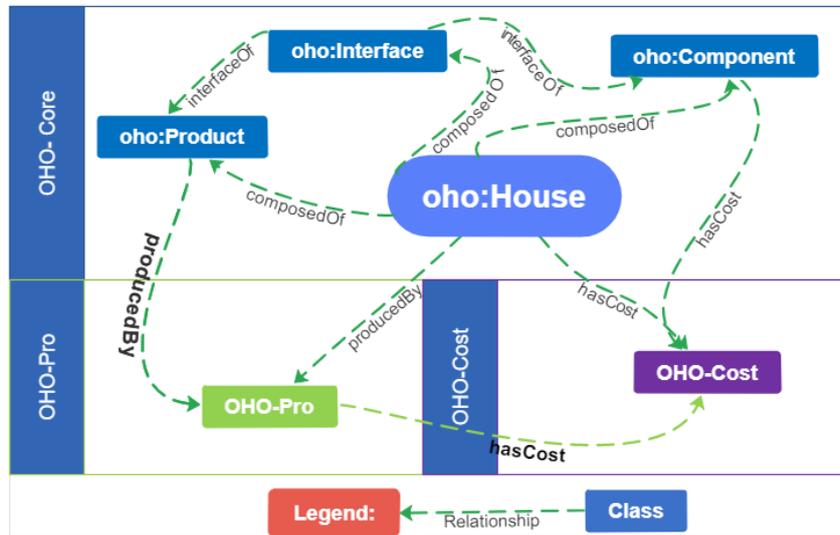


Fig. 2. Illustration of the OHO core classes.

relations as illustrated in Figure 2. These ontologies are defined in the namespaces and prefixes listed in Listing 1.

Listing 1: OHO prefix.

```
@prefix oho: <https://w3id.org/oho#> .
@prefix oho-pro: <https://w3id.org/oho
  ↪ -pro#> .
@prefix oho-cost: <https://w3id.org/
  ↪ oho-cost#> .
```

To support the detailed development of OHO, the scope and purpose of the OHO ontology was defined using the following competency questions.

- **CQ1:** What are production methods for a certain building product?
- **CQ2:** What are the stations for a production line method?
- **CQ3:** What are the activities carried out in each station of a production line method?
- **CQ4:** What type of material is required to produce a product?
- **CQ5:** What type of labour profile is required to work in each activity?
- **CQ6:** What is the cost of a DfMA house?
- **CQ7:** How long does it take to produce a Product / DfMA Product?
- **CQ8:** What are the components of a Product?
- **CQ9:** What are the resources needed for the production of a Product?

5.1. OHO Core

The OHO core module describes the core parts of a house constructed with the DfMA approach. Given that manufacturing is a major part of DfMA, this core module also includes production line aspects. The main classes of the OHO core ontology are:

- oho:House defines the house to be produced and built with the DfMA approach (oho:House);
- oho:Product categorizes the modular products produced for the scope of the DfMA house (e.g. wall panels are represented as oho:Product);
- oho:Component gives the details of finished modular components installed in a DfMA house (e.g. kitchens or bathroom are represented as oho:Component);
- oho:Interface is a generic class that defines the type of relationship that connects products or components together
- oho:Production defines the production line that produces an oho:product (e.g. panelized systems are represented as oho:Production);

5.1.1. House

In DfMA the design of the house and its individual components integrates production line concepts. In order to represent this semantically, the oho:House class functions as a central concept, combining all aspects that have to be defined in order for a house to be produced and built with the DfMA approach. The oho:House class can also have the object

properties `oho:composedOf`, pointing either to a `oho:Product` or `oho:Component`.

5.1.2. Product

The class `oho:Product` defines an object that is manufactured in factory using a DfMA production line. The definition of an `oho:Product`, and associated properties such as its cost, take into account the manufacturing concept of the production line. The `oho:Product` class is elaborated further with the subclasses such as `oho:Pod`, `oho:Panel` and `oho:PodFrame`. Details of the production line and product attributes are required to describe such a product, using the object property `oho:producedBy`, which relates the product to a `oho:Production` object.

5.1.3. Component

The `oho:Component` class represents a part used to compose a house or a product, which is directly procured in a finished state (i.e. outside the DfMA production line), either as discrete components (e.g. windows, doors), or complete modular components (e.g. a kitchen). The subclasses of `oho:Component` are listed in Table 3 and can be easily expanded given new product design. The composition of a DfMA house can be defined and inferred to a high level of granularity (e.g. up to the level of individual bolts in a connection), using transitive property `oho:composedOf` between OHO classes. The `oho:composedOf` property has `oho:House` and `oho:Product` classes as a definition domain and `oho:Component` and `oho:Product` classes as a range. For example, if a DfMA house is composed of panels, and the panels are composed of studs, windows, studs etc., then it can be inferred that the DfMA house is composed of the window, the studs etc. as well.

5.1.4. Interface

An interface in the OHO ontology represents a physical connection layer between a product and a component, between components or between products. The `oho:Interface` class allows to define interfaces between products and components as needed. For example the connection points between multiple components of a product (subassemblies) can be represented with an interface. At a higher level, an interface may also be used to define the connections between multiple products in an `oho:House`. Conceptually, this class draws from the `bot:Interface` class of the BOT ontology,

but adjusted to a DfMA context. Products and components are linked with interfaces using the `oho:isInterfaceOf` property. The domain of this is an `oho:Interface` class and the range is defined by `oho:Product` and `oho:Component` classes.

5.1.5. Production

The class `oho:Production` describes the methodology used in producing an `oho:Product`. As the aim was to develop a modular and extensible ontology, detailed subclasses and object properties were not introduced in the core ontology. Instead, this core ontology was extended with a dedicated ontology aimed at the representation of production methods (Section 5.2). This approach allows:

- the use of other ontologies to define production processes, while still relying on the OHO core,
- the use of the simpler and more general OHO core in case no production details are present,
- management of the OHO Production ontology with appropriate scope and focus.

5.2. OHO Production

The OHO Production module (`oho-pro:`) imports the OHO core and defines several dedicated classes and properties. It consists of four parts, namely Method, Station, Activity, and Resource (see Fig. 3). First, the `oho-pro:Method` class allows to categorise different types of production line methods e.g. automated production line; second, the `oho-pro:Station` class allows to define the stations of a production line, it has a starting station (`oho-pro:isStartingStation`) a next station (`oho-pro:hasNextStation`) and a final station (`oho-pro:isFinalStation`); third, the `oho-pro:Resource` class defines the resources needed (related to CQ9) for a successful operation handled by a station i.e. `oho-pro:Material` or `oho-pro:Labor`; and the `oho-pro:Activity` class represents distinct activities in the entire production line.

5.2.1. Production Line

The `oho-pro:Method` gives a semantic description of a production line. It is composed of stations (`oho-pro:Station`) and assigned to a `oho:Product` via a `oho-pro:hasMethod`. The semantic information is used to answer CQ1. Table 4 shows a number of classes available in the OHO Production ontology, with corresponding real-world

Table 3

A list of the subclasses of oho:Component class.

oho:Window	oho:KitchenCabinet
oho:Tape	oho:RecessedLight
oho:LightingCable	oho:Pipe
oho:Battery	oho:Socket
oho:LightSwitch	oho:SanitaryFitting
oho:SolarPanel	oho:BathroomCabinet
oho:Door	oho:PendantLight
oho:HeatRecoveryUnit	oho:Stair
oho:WhiteGood	oho:Radiator
oho:ElectricityCable	oho:Duck
oho:KitchenCable	

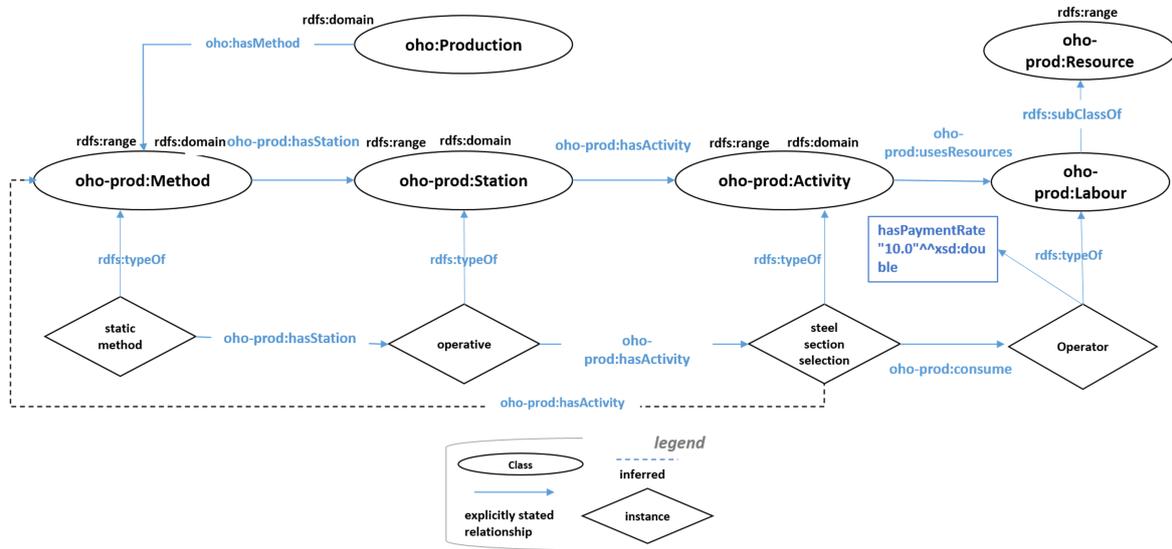


Fig. 3. Illustration of the oho:Production class, its subclasses and examples of instances e.g. static method as an instance of the class oho-pro:Method.

examples (i.e. implementation instances). This clearly indicates that a lot of diversity is present in stations, production lines, components and activities.

5.2.2. Method

The number and type of stations that constitute a production line are imposed by the production method. For example, a panelised system production line has a different production line compared to a pod system production line. A class oho-pro:Method is therefore defined to reflect the production line method and each method is connected with the different stations needed to complete the production (object property oho-pro:hasStation). However,

stations are defined as single instances and can be reused in different methods; e.g. a “Loading Station” can be defined once and the same instance reused in different production methods. A decision was made not to define production sub-methods directly in the OHO ontology. These, however, can be defined as subclasses of oho-pro:Method.

5.2.3. Activity

Capturing the activities that take place in a production line is an important part of a successful representation. The definition of those activities also enables the definition of constraints and checks on these activities. In this regard, OHO clearly divides

Table 4
OHO taxonomy of major elementary categories.

Entity Production Station	Example of Activity
Cladding-Assembly-Line	
CP-Boarding-Station	T10-Frame-transfer T11-Load-and-Place-CPB T12-Mechanical-Fixing
Adhesive-Station	T31-Feed-Adhesive T32-Dispense-Adhesive
Briquette-Apply-Station	T35-Place-Briquettes
Briquette-Load-Station	T34-Feeding-Briquettes
Frame-Assembly-Line	
Conveying-Station	T13-Return-Conveyor
Frame-Riveting-Station	T5-Rivet-Joints T6-Move-Frame-to-Lift T7-Lift-Frame
Frame-Transfer-Station	T9-Transfer-Frame
FrameBeam-Load-Station	T1-Deliver-Pallets T2-Select-and-Load-Beam T3-Clamp-Beam
Supporting-Activity	Quality-Inspection

production processes into activities that produce directly, e.g. `oho:CladdingAssembleyLine`, and activities that support production (e.g. loading, packaging and transporting). For example, a cladding assembly line is an automated activity that is defined as a production activity; a constraint placed on it is that it can only start after the frame assembly line has been completed; a property of it is that it consumes labour. The constraints in Listing 2 indicate how such restrictions can be included in the data.

Listing 2: Assembly line definition including constraints that can be set on activities and resources these activities consume.

```

oho-pro:CladdingAssemblyLineAutomated
↔ :=
oho-pro:CladdingAssemblyLine AND
    
```

```

oho-pro:beginsAfter ONLY oho-pro:
↔ FrameAssemblyLine AND
oho-pro:consumeLabour SOME oho-pro:
↔ Labour AND
oho-pro:hasSupportingActivity SOME (
oho-pro>Loading AND
oho-pro:consumeLabour SOME oho-pro:
↔ Labour )
    
```

5.2.4. Station

A `oho:Station` class defines a work station in the production line which covers **CQ2**. Each work station performs one or more production activities (e.g., cladding assembly line automation, frame assembly line) and needs time and resources (e.g., labour, material, component, overhead, etc.) to produce a product. These stations perform activities (related to **CQ3**), and those activities in turn use resources (define in a

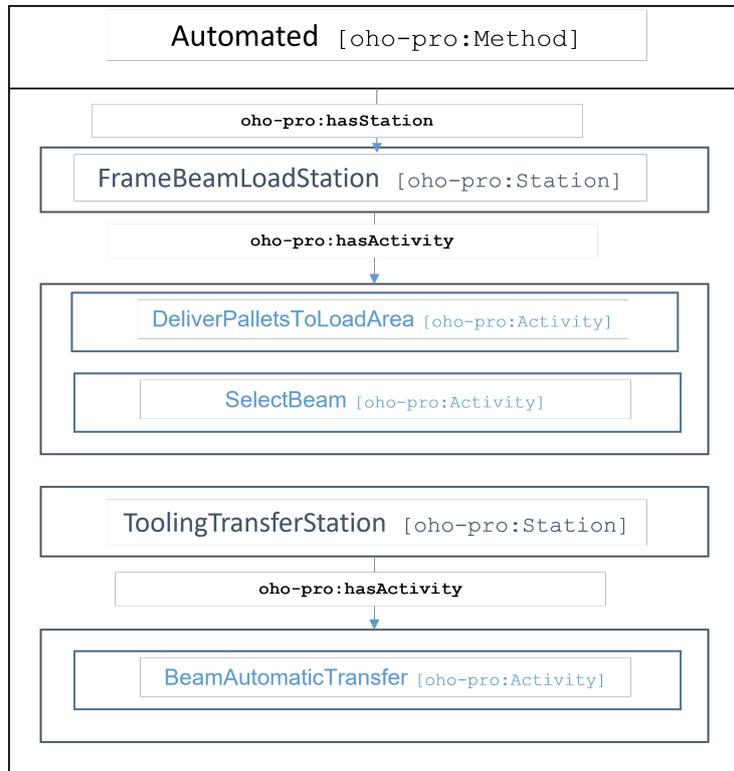


Fig. 4. Illustration of the class dependency of the automated production line.

oho-pro:Resources property) and time (defined in a (oho-pro:hasProcessingTime property - CQ7). These stations are crucial to the production system in the OHO ontology. The resulting structure is shown in Fig. 4.

A method of production (oho-pro:Method) has one or more stations (oho-pro:hasStation) and each station has one or more activities (oho-pro:hasActivity). The last property is used in a property chain axiom that formalises the following transitive relationship: if a method has a station that has an activity, then the method has that activity as well (Listing 3).

Listing 3: Property chain axiom for stations activities and methods represented in Turtle Syntax.

```

PREFIX owl: <http://www.w3.org
  ↪ /2002/07/owl#>
PREFIX oho-pro: <https://w3id.org/oho-
  ↪ pro#>
oho-pro:hasActivity owl:
  ↪ propertyChainAxiom
  
```

```

(oho-pro:hasActivity oho-pro:
  ↪ hasStation ).
  
```

Similarly, a property chain axiom for resources is defined: if an activity performing on a station of a production method uses some resources, then the station where the activity takes place, uses these resources as well (Listing 4).

Listing 4: Property chain axiom about resource activity and station presented in Turtle syntax.

```

PREFIX owl: <http://www.w3.org
  ↪ /2002/07/owl#>
PREFIX oho-pro: <https://w3id.org/oho-
  ↪ pro#>
oho-pro:usesResources owl:
  ↪ propertyChainAxiom
(oho-pro:usesResources oho-pro:
  ↪ hasActivity ).
  
```

5.2.5. Resource

Resources in OHO are classified in these subclasses: `oho-pro:Material`, `oho-pro:Labour`, `oho-pro:Plant` and `oho-pro:Overhead`. The `oho-pro:Material` class collects instances that represent the different types of material needed and used for the production of a product such as a wall panel. This class is described in more detail in the following section. The `oho-pro:Labour` class captures data about the type of labour that is engaged in the production process and is sub categorized as `oho-pro:Skilled`, `oho-pro:SemiSkilled` and `oho-pro:Unskilled`. This categorisation responds to **CQ5**. Different payment rates can be assigned to different labour types using the `oho-pro:hasPaymentRate` data property and the activity it is allocated to (`oho:workingOnActivity`) for a specified amount of time. The class `oho-pro:Plant` gives details about the plants (e.g., movable tools, static tools) used in the production process and `oho-pro:Overhead` defines activities of type cleaning, security etc.

The `oho-pro:Material` class defines the types of material used in different activities of the DfMA house production and assembly (related to **CQ4**). Each material is represented as an instance of this class and is semantically enriched with details for: its cost (`oho-pro:hasCost`); the source (`oho-pro:hasSource`) of information such as a reference database of rates; the conversion factor (`oho-pro:hasConversionFactor`); the quantity of the material (`oho-pro:hasQuantity`). Cost rates are affected by both the vendor and the time. In order to model this relationship, the `oho-pro:hasSource` property is used to capture the vendor, and the `state` property is used to capture the time. The latter is inspired by the OPM ontology [54].

A sample list of the directly procured materials for producing a Pod sub-assembly extracted from a BIM model is listed in Table 5.

The class `oho:Material` is useful in many way such as to get information about the material used in producing a product or in cost calculation or estimation.

5.3. OHO Costing module

While many ontologies in the LBD community are oriented towards the definition of the geometry

and semantics of a building or its elements, one of the advantages of the OHO ontology is that it combines this with the capacity to produce cost estimates. The second OHO ontology module that was developed facilitates the cost estimation of a DfMA house (related to **CQ6**). The main drivers of the cost estimation of a DfMA house (Figure 5) are defined in the following classes:

- `oho-cost:BusinessOperation`,
- `oho-cost:DesignAndEngineering`,
- `oho-cost:OffsiteProduction`,
- `oho-cost:Transportation`,
- `oho-cost:OnsiteProduction`,
- `oho-cost:InUse`.

The cost estimation for a DfMA house is activity-based, which accounts for the chain of activities in the offsite production. In order to capture this process, the `oho:OffsiteProduction` class is defined with a high level of granularity and detail.

6. Evaluation of OHO

According to the NeOn methodology, there are three evaluation criteria: i) *Domain coverage*, ii) *Quality of the modeling* and iii) *Suitability for an application or task*. An additional criterion suggested according to NeOn, i.e. *Adoption and use*, can only be evaluated over time, and thus is excluded from this study. OHO was assessed by the group of experts working in the AEC domain.

6.1. Domain coverage

The first version of the ontology was built by extracting knowledge from non-ontological resources and reviewing existing ontologies. Subsequent to the release of the first version, a few focus group meetings with the group of experts were conducted to refine OHO. The proposed CQs are covered with the ontology. Additional concepts and relationships were defined, ambiguous concepts were clarified and some terms were amended to facilitate common understanding.

6.2. Quality of the modeling

This evaluation criterion assesses the quality of the ontology based on a set of metrics and a list of attributes of the ontology development process. OHO contains 151 classes, with 61 object properties, and 77 data properties. **It was tested using Pellet reasoner in**

Table 5
Directly procured material for Pod sub-assembly.

Plasterboard	Hot Water Cylinder
Joint tape to internal walls	Electrical Installation
Insulation to external wall	Lamps
Stairs and balustrades	Basin
Timber studs	Battery System
Kitchen Cabinets	Solar PV
Bathroom Cabinets	Telephone and TV Distribution
Sockets	Energy and fault monitoring
Light switches	Bathtub
Pendant lights	Shower screen
Recessed lights	Taps
Lighting Cable	Water Cistern
Data Cable	Shower Thermostatic Mixer
Electricity Cable	Kitchen sink
Hot and cold water pipe	Kitchen mixer
Drainage pipe	Smoke detector
Panel Heaters	Heat detector
MVHR	CO2 detector

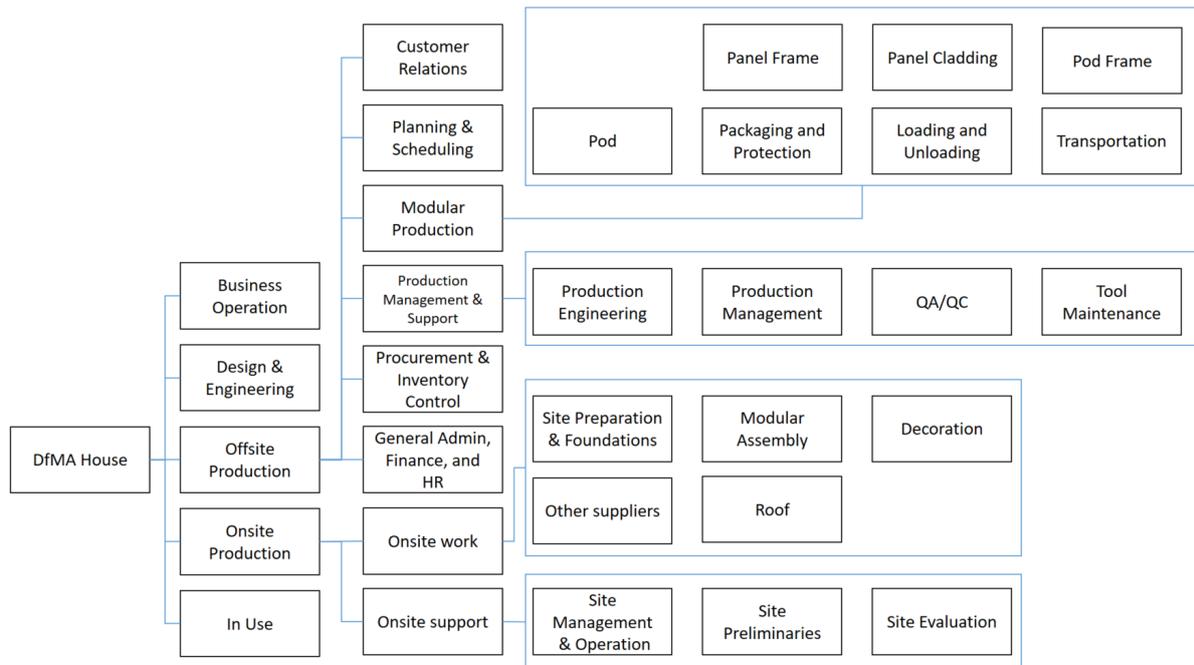


Fig. 5. Main drivers of a DfMA house project implementation.

Protégé to demonstrate that the schema is consistent. A comparison between OHO and other ontologies is presented later in this article.

The following attributes are used to evaluate the ontology development process [55]:

- **Accuracy:** The accuracy evaluates whether the ontology capture and represent correctly aspects of the real world. The ontology development process was assisted from the group of experts in the DfMA domain. The processes were supported by highly accurate non ontological resources such as the BIM model among other sources.
- **Adaptability:** The adaptability evaluates whether the ontology can be extended to offer a conceptual foundation for a range of anticipated tasks. OHO is a modular ontology and each module can be used independently and this provides reusability and extensibility. The OHO-Cost module can be used to estimate traditional construction project as well as offsite construction and OHO-Pro can be reused for other types of production lines.
- **Clarity:** The clarity evaluates whether the ontology communicate effectively the intended meaning of the defined terms. All the defined terms contain non-ambiguous names and are labelled with definitions and supplementary information to ensure common understanding.
- **Completeness:** The completeness evaluates whether the ontology defined all competency questions. The OHO ontology can answer all the competency questions defined in the ontology requirement specification.
- **Efficiency:** The efficiency evaluates how fast can the usual reasoning services be applied to the ontology. Querying the ontology using the SPARQL query language protocol was tested in the Protege framework and in GraphDB¹. The queries run in milliseconds in each of these environments.
- **Conciseness:** The conciseness evaluates whether the ontology include irrelevant axioms with regards to the domain to be covered. The knowledge modelled in the OHO ontology and its modules was captured from domain trusted sources, i.e. knowledge shared from the group of experts in this case.

- **Consistency:** The consistency evaluates whether the axioms lead to contradictions (logical consistency). Reasoning based on OHO was performed using the Pellet reasoner². No inconsistencies were found.

7. Demonstration of the OHO ontology

7.1. DfMA house wall panels and components of a wall panel

In order to demonstrate the functionality of the OHO ontology, a case study of a DfMA house composed of 32 wall panels was registered by instantiating the ontology. Different queries were executed (such as “what are the components of a wall panel?”, “what is the production time?”) in order to extract the captured and inferred data. The wall panels were modelled by describing their attributes, and the production line was detailed in terms of the activities required to produce a wall panel (Figures 6, 7).

All activities are related to each other, they are the first activity (`oho:hasStartingActivity`), after or before another activity (`oho:hasNextActivity`) or in parallel to another activity. The knowledge represented in the ontology was used to estimate cost per each activity and the overall cost of producing a sample product (e.g. a wall panel). By estimating the cost per each activity, a designer can get insights about the related activities and the assigned overhead costs. They can revise the design if needed.

The following are example SQWRL (Semantic Query-Enhanced Web Rule Language) queries that were executed with the Protégé 5 SQWRL plugin in relation to our example. The `sqwrl` prefix is used to denote the SQWRL operator and `swrlb` for identifying SWRL built-ins.

Q1: What is labor cost for each semi-skilled operative working on each activity of the wall panel production?

Listing 5: Semi-skilled labour cost working on a wall panel production activity.

```
PREFIX swrl: <http://www.w3.org
↔ /2003/11/swrl#>
```

¹<https://graphdb.ontotext.com/>

²<https://www.w3.org/2001/sw/wiki/Pellet>

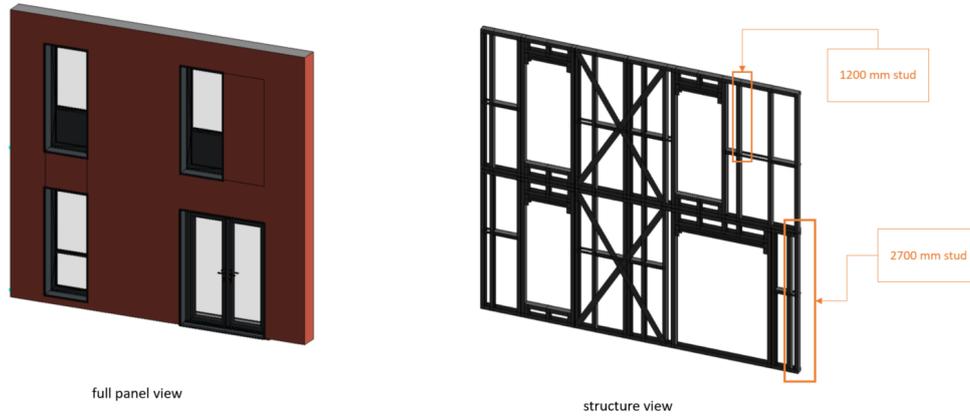


Fig. 6. The image of a wall panel in a DfMA house.

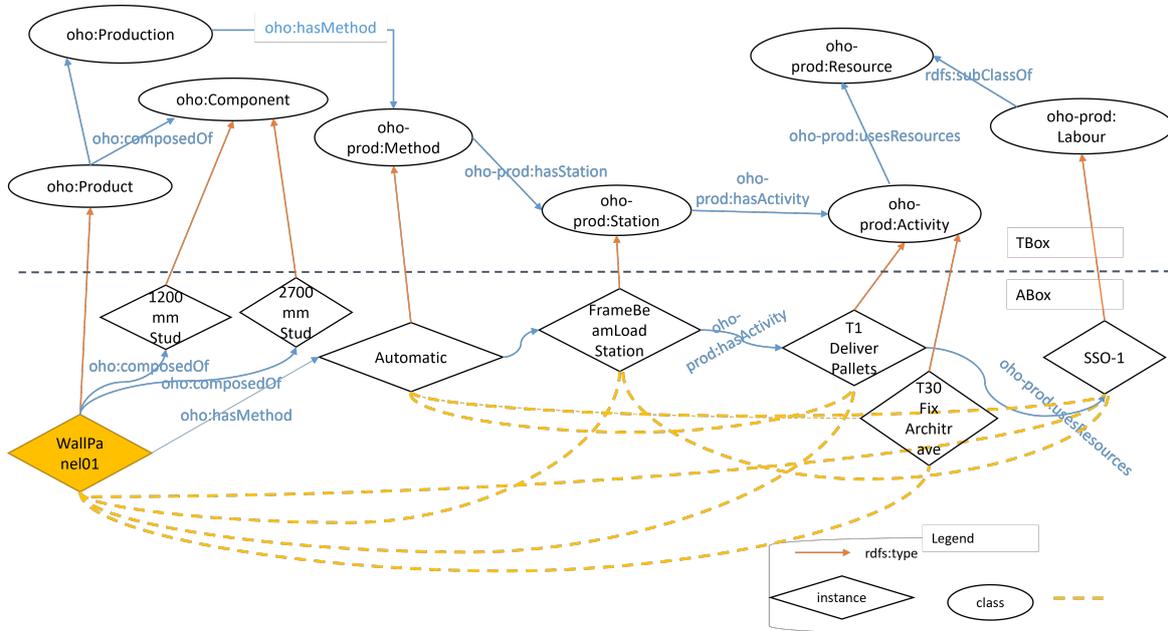


Fig. 7. The representation of a wall panel of a DfMA house in the OHO ontology.

```

PREFIX swrlb: <http://www.w3.org
  ↪ /2003/11/swrlb#>
PREFIX sqwrl: <http://sqwrl.stanford.
  ↪ edu/ontologies/built-ins/3.4/
  ↪ sqwrl.owl#>
Semi-Skilled_Operative(?s) ^
  ↪ Activities(?a) ^
  ↪ workingOnActivity(?s, ?a) ^
  ↪ hasProcessTime(?a, ?p) ^
  ↪ hasLabourHrRate(?s, ?r) ^ swrlb:
  ↪ multiply(?result, ?p, ?r) ->
  ↪ sqwrl:sum(?result) ^ sqwrl:
  ↪ select(?s)

```

Q2: What are the parts (related to **CQ8**) of the product *WallPanel01* wall panel?

Listing 6: List of the components of a wall panel.

```

Product(?p) ^ hasComponentPart(?p, ?
  ↪ Component) -> sqwrl:select(?
  ↪ Component)

```

Q3: What is the starting and upcoming activity for producing *WallPanel02* wall panel?

Listing 7: Starting and upcoming activities for producing a wall panel.

```
Product (LSF_3BED_01_LHS) ^
  ↪ hasStartingActivity (WallPanel02,
  ↪ ?StartActivity) ^
  ↪ hasNextActivity (?StartActivity,
  ↪ ?NextActivity) ->sqwrl:select (
  ↪ WallPanel02, ?StartActivity, ?
  ↪ NextActivity, )
```

7.2. Querying a Pod product

The level of granularity OHO enables many applications, including knowledge extraction using semantic search. Listings 8 to 10 show a number of relevant SPARQL queries to extract knowledge from a pod product. The queries are executed from Protégé 5 Snap SPARQL Query plugin and the inferred knowledge is leveraged, benefiting from the rules used.

Listing 8: Select the list of directly procured material needed for the pod production and their cost.

```
PREFIX rdf: <http://www.w3.org
  ↪ /1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org
  ↪ /2002/07/owl#>
PREFIX rdfs: <http://www.w3.org
  ↪ /2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/
  ↪ XMLSchema#>
PREFIX oho: <https://w3id.org/oho#>

SELECT *
WHERE { ?s a oho:
  ↪ DPMforPod_Sub_Assembly; oho:
  ↪ hasActivityCost ?c }
```

Listing 9: Select the method of production for pod frame therefore the stations and activities.

```
PREFIX rdf: <http://www.w3.org
  ↪ /1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org
  ↪ /2002/07/owl#>
```

```
PREFIX rdfs: <http://www.w3.org
  ↪ /2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/
  ↪ XMLSchema#>
PREFIX oho: <https://w3id.org/oho#>

SELECT ?m WHERE { ?p a oho:Product;
  ↪ oho:hasMethod ?m}
```

Listing 10: Select the 'parts' that compose a pod.

```
PREFIX rdf: <http://www.w3.org
  ↪ /1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org
  ↪ /2002/07/owl#>
PREFIX rdfs: <http://www.w3.org
  ↪ /2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/
  ↪ XMLSchema#>
PREFIX oho: <https://w3id.org/oho#>

SELECT ?m WHERE { ?p a oho:Product;
  ↪ oho:composedOf ?m}
```

7.3. Cost Estimation of DfMA using a Semantic Rule Language

Rules in the form of the Semantic Web Rules Language (SWRL) are used to provide more powerful deductive reasoning capabilities than OWL alone. For example, the rule in Listing 11 enables the calculation of the labour cost.

Listing 11: Labour cost per activity calculated from a SWRL based on the time spent on an activity and the labour rate.

```
Activities(?a),Labour(?s),
  ↪ hasLabourHrRate(?s, ?r),
  ↪ hasProcessTime(?a, ?p),
  ↪ workingOnActivity(?s, ?a),
  ↪ multiply(?result, ?p, ?r) ->
  ↪ hasActivityCost(?a, ?result)
```

Also, a dedicated set of SWRL rules has been created in addition to the presented OWL ontology modules, in order to evaluate their accuracy for cost estimation. A hypothetical project containing the offsite production of 200 houses of different types

Table 6
Project requirement specification.

Requirement	Value
Choice of offsite main frame sub-assembly systems:	Volumetric Type 1 (Pod with infill panel)
Production method	Automated panel
Batch size	200
Period to produce the batch	12
Project Assumptions	
Address (Post Code)	B12 0NW
<i>No of units for each house type</i>	
Detached	2
Semi-detached	2
Terrace	4
Average no. per row	4
<i>Service System choice</i>	
MVHR	Yes
Hot Water Cylinder type	Daikin
Electrical Installation Type	Extreme Low Energy
Lamps to be supplied	Yes
Energy and fault monitoring required	Yes
Battery System	Yes
Solar PV	Yes
Telephone and TV Distribution	Yes

(i.e. detached, semi-detached and terrace) that uses an automated production method were registered as input to the ontology using the terms of the OHO core, production and cost ontology modules and the full list of requirements defined as shown Table 6.

As the OHO-Cost ontology module applies activity-based costing methodology, all activities are instantiated with their respective cost values. A starting activity is defined (`oho:isStartingActivity`) and associated with other cost activities using the `oho:hasNext` property. SWRL rules were set to chain the cost value instances together and at the same time sum them up to an accumulated cost (`oho-cost:hasActivityCost`) for each activity stage. Furthermore, a final overall inferred estimated cost is calculated at the end of the final activity (`oho-cost:hasActivityCostFromStart`). The rules that support the cost estimation of an offsite housing project are presented in Listings 12 and 13.

Listing 12: Assigning the cost to the starting activity of a process.

```
hasActivityCost(?a, ?c) ^
  ↪ isStartingActivity(?a, ?b) ->
  ↪ hasActivityCostFromStart(?a, ?c
  ↪ )
```

The rule on Listing 12 states that if an activity `a` is the starting activity of a chain of activities and has a given cost `c` then the value of `oho:hasActivityCostFromStart` will be `c` also. The main purpose of this rule is to initialize the `oho:hasActivityCostFromStart` data property.

Listing 13: Updating the overall cost estimation after each activity is added.

```
Rule 3: swrlb:add(?nptd, ?ptd, ?npd) ^
  ↪ hasNextActivity(?p, ?np) ^
  ↪ hasActivityCostFromStart(?p, ?
  ↪ ptd) ^ hasActivityCost(?np, ?npd
```

```

1   ↪ ) -> hasActivityCostFromStart(?
2   ↪ np, ?nptd)

```

The `oho-cost:hasActivityCostFromStart` is updated by adding the value of the current cost activity found in `oho-cost:hasActivityCost`. The new value is assigned to the current cost activity `?np` and is a result of the cost activity `np` and the cost carried on the `oho-cost:hasActivityCostFromStart` property from the start of this chain of cost activities (Listing 13).

Overall, the instantiated cost values along with the SWRL rules, and reasoned by the Pellet reasoner provided exactly the same result with the estimation from a Quantity Surveyor expert. Explanations for the semantic reasoner (e.g., Pellet) inferences is available in the justification of results that is provided as an advanced functionality in Protégé. For the expert that does the evaluation directly using the OHO ontology and the SWRL rules for cost estimations, an ontology-based approach can provide the insight into the reasons behind estimations. The available explanations can serve as a white-box proof of the results and hence, plays an important role in building trust of OHO.

7.3.1. Validating data integrity with SHACL

OWL and SWRL are based on logic, do not support non-monotonic reasoning and use the Open World Assumption (OWA) where the missing data is assumed as not identified yet. In cases where validating data integrity is crucial, there is a need to use the Closed World Assumption (CWA) in order to notify a constraint violation or take action by adjusting data to a standard format. SHACL shapes is one of the newest technologies that has filled this gap in the Semantic Web architecture stack. Listing 14 shows an example of a SHACL shape for validating the properties (`oho:hasHeight`, `oho:hasWidth` etc.) of a `oho:WallPanel` instance before it goes to production. SHACL validation will ensure that geometric inputs conform to the required standards.

Listing 14: Data integrity validation using SHACL model.

```

48 @prefix rdf: <http://www.w3.org
49   ↪ /1999/02/22-rdf-syntax-ns#> .
50 @prefix sh: <http://www.w3.org/ns/
51   ↪ shacl#> .

```

```

1 @prefix xsd: <http://www.w3.org/2001/
2   ↪ XMLSchema#> .
3 @prefix rdfs: <http://www.w3.org
4   ↪ /2000/01/rdf-schema#> .
5 @prefix owl: <http://www.w3.org
6   ↪ /2002/07/owl#> .
7 @prefix oho: <http://w3id.org/oho#> .
8
9 oho:WallPanelShape
10   a sh:NodeShape ;
11   sh:targetClass oho:WallPanel ;
12   sh:property [
13     sh:path oho:hasHeight ;
14     sh:minCount 1 ;
15     sh:datatype xsd:decimal ;
16   ];
17   sh:property [
18     sh:path oho:hasWidth ;
19     sh:minCount 1 ;
20     sh:datatype xsd:decimal ;
21   ];
22   sh:property [
23     sh:path oho-pro:hasLength ;
24     sh:minCount 1 ;
25     sh:datatype xsd:decimal ;
26   ];
27   sh:property [
28     sh:path oho:hasFrame ;
29     sh:minCount 1 ;
30     sh:class oho:Frame ;
31   ];
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

```

7.4. KBE tool Development

The OHO ontology was used to implement a Knowledge-Based Engineering (KBE) tool for cost estimation of housing projects using the DfMA approach. The application is implemented according to the Representational State Transfer (RESTful) architectural pattern, with an application programming interface (API) offering intermediary services to other web frontend endpoints and acts as a gateway to the OHO ontology and database servers. A web-based user friendly interface is provided for potential users such as architects, production engineers, structural engineers, steel suppliers, clients and cost consultants, etc.

7.4.1. Input of the KBE tool

DfMA houses are platform-based with standardised design prototypes. To evaluate the cost of a DfMA house, the following input is required:

1. Choice of sub-assembly system
2. Offsite sub-assembly approach
3. Batch size and expected batch delivery period
4. Project site location
5. Project size in terms of house numbers and types of houses
6. Design choice
7. Service installation choices

They are defined according to a set of entries in an evaluation form as shown in Figure 8.

The data used are based on a platform-based modular house developed in a UK government funded industrial research project by implementing DfMA design methodology. Detailed information was produced for the construction of a prototype of the modular house. The source of data includes a BIM model in an IFC format as well as cost data sheets and machining process data sheets. Built upon those data input, the KBE tool can be used for the cost estimate (Figure 9).

7.4.2. Output of the KBE tool

The cost estimation module produces cost estimation as well as the breakdown of the costs. In addition to the overall cost estimations, there are 4 levels of breakdown: activity group, activity, sub-activity, and resource. As the focus of the prototype is on Manufacturing and Assembly, all four levels of breakdown are available for the activities in relation to the manufacturing and assembly processes. The estimations and breakdown are shown in an interactive dashboard for the ease of visualisation. A screenshot of the dashboard is shown in Figure 10.

7.4.3. KBE Architecture

The KBE tool architecture (Figure 11) is designed with the capability to integrate and process heterogeneous data formats and to accommodate semantic web and web technologies. At the core of it is a Knowledge Graph defined as a knowledge bases that store factual information in form of relationships between entities [56] described with formal semantics of OHO.

The *Data layer* stores different file formats that contain information used for cost calculation, such as BIM model in IFC format of the house, productivity datasheets, and other sources of information. The *Transformation Layer* parses the different data sources and transforms the relevant information into the RDF format. At this stage the data is fed into the DfMA Knowledge Graph and is semantically

described with the OHO ontology concepts and relations creating a DfMA Knowledge Graph. The OWL API [57] library is used to facilitate the conversion of these semi structured data to a LBD format. The SPARQL query protocol or other Query API serves as an intermediate layer between the user Application layer and the other layers of the architecture. The user sends queries by entering filtering information and receives a visualized response.

7.4.4. Validation of the KBE tool

The KBE tool was presented and tested by the group of experts to calculate the total cost per house.. Besides, the cost was estimated by Quantity Surveyors. It is found that the outputs generated from the tool were the same as Quantity Surveyors' estimates.

8. Conclusion and Discussions

A new domain specific ontology, OHO, was designed and validated to represent and bring together disparate and isolated knowledge and data from various fields. The main contribution of the paper is the very detailed ontologies developed for offsite house construction that supports building design using DfMA. The concepts and relationships defined particularly about the production and costing enable users to retrieve knowledge that can support DfMA design. The use of it is demonstrated from the answers to the competence questions as well as the KBE cost estimation tool.

OHO expanded the ifcOWL ontology and used a real-life use case production lines to test and demonstrate the benefits of the semantic digital twin in obtaining data of the manufacturing for assessment. Analysing the specifics of the production processes we identified the need for OHO to grow as a separate domain ontology. Naturally, many aspects of a completed DfMA project, such as the building geometry or the material properties, fits ifcOWL concepts and can be represented accordingly. However, as a process with roots in industrial engineering, DfMA engages more with procedural and optimisation aspects, and introduces concepts, such as "assembly" or "sub-assembly", with different semantics from current BIM and construction technology practice. As such, a separate ontological domain was considered necessary in order to avoid semantic and ontological conflicts, as well as to

Fig. 8. Interface of the web evaluation form.

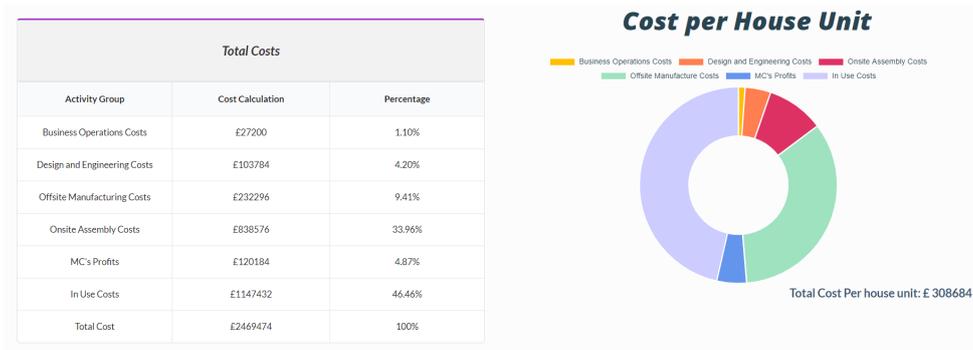


Fig. 9. Dashboard screenshot of cost estimation from KBE tool.

implement DfMA concepts appropriately. Ideally, OHO is able to facilitate a two-way conversation: enable AEC practitioners to apply DfMA design concepts in a BIM workflow, while simultaneously acting as an introduction to the DfMA concept for BIM-literate AEC practitioners.

Furthermore, such modular extension approach also entirely fits the recommendations put forward by the W3C Linked Building Data Community Group, as well as linked data communities at large. By aiming at a separate domain, such recent research initiatives are followed and implemented, thus moving away from a monolithic ontology approach. Attempting to capture all possible aspects of a building (or any other concept) in a single ontology, mapped to a super-schema, has innate limitations and lacks the flexibility to accommodate different design concepts (i.e. extensibility). DfMA is the future innovative

philosophies and practices for construction. It is likely to face similar challenges in BIM implementation.

The OHO ontology can be aligned with the IFC ontology by asserting that every `oho:Building` is an `IfcBuilding` and every `oho:Product` is an `IfcProduct`. An exploitation of OHO alignments with other standardised AEC ontologies such as BOT ontology are made such as an alignment in the instance level i.e. a DfMA product such as Pod (`oho:Product`) is a `bot:Element` and a DfMA house (`oho:House`) is a subclass of a `bot:Building`. The current version of OHO can be added as a module of a standardised ontology, and to make links on instance level (for web of linked data). A formal alignment approach will be developed further after OHO reaches a more mature stage. The instance linking proposed in the paper is recommended as a priority over more stringent ontology alignments due to practical implementations and research initiatives.

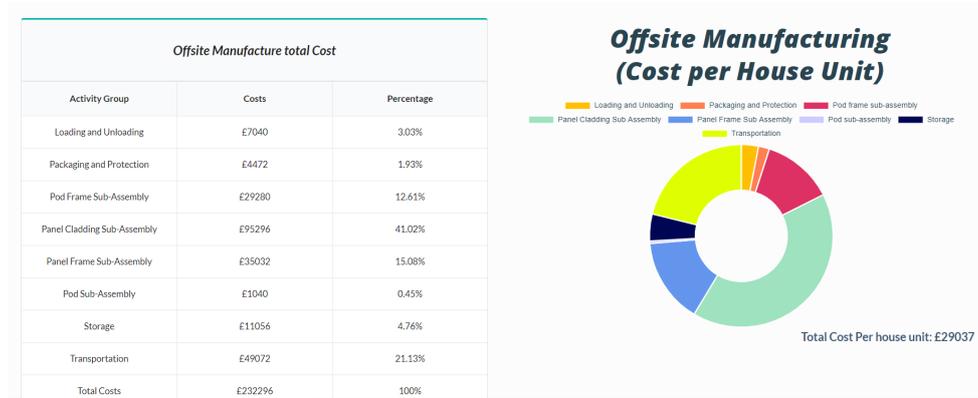


Fig. 10. Dashboard screenshot of production cost breakdown

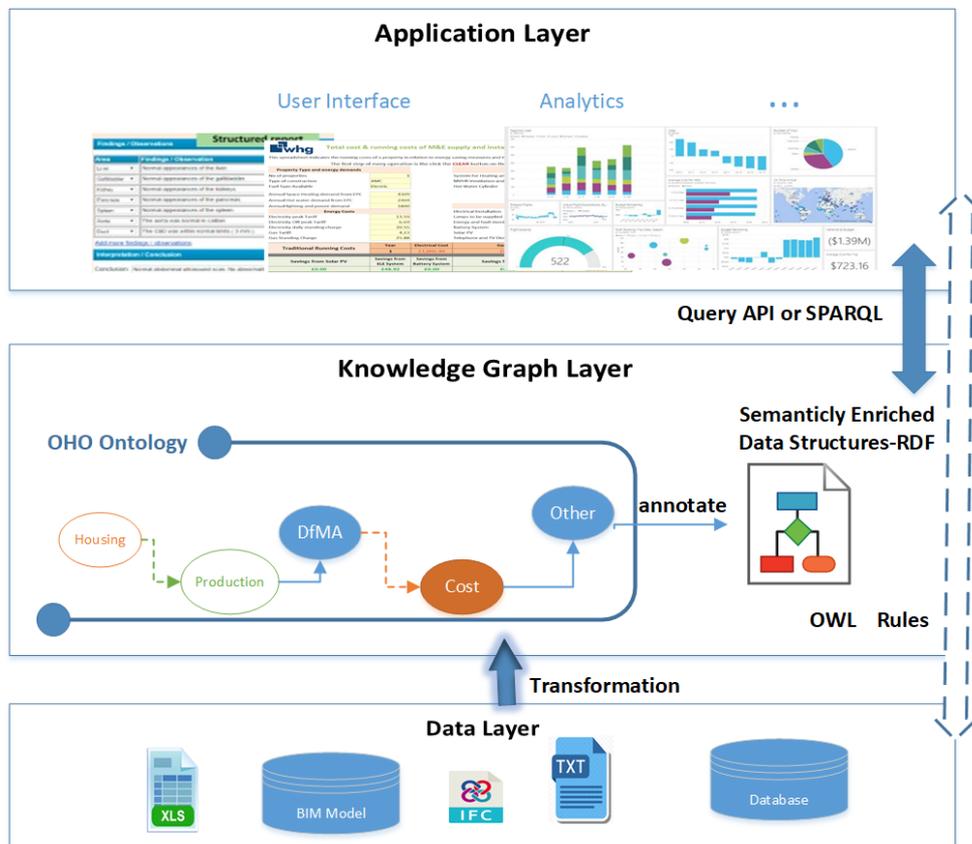


Fig. 11. Web-Application Architecture.

There are many directions that the base OHO ontology can be expanded to or used for when building DfMA applications. For example, in this paper, the OHO core semantic model was applied to develop the production and costing modules for cost estimation tool. The use of the OHO

ontology has also been experimented in different environments in order to accommodate diverse users' needs with varying levels of knowledge of the underlying semantic web technology including some users with a computer science background and most from the AEC community. Users with knowledge of

using an ontology editor (e.g., Protégé) are able to directly interact with the semantic model, while less experienced users required a frontend application that uses an API (e.g. OWL API) to connect with the OHO ontology as discussed in the *OHO Evaluation* Section.

A deep analysis of the best approach to follow in terms of efficiency, complexity and ease of use will be part of future works. Finally, the OHO ontology paves the way to the implementation of Digital Twins that integrates production information of houses with BIM models. Future developments of this work will focus on further iterations of the OHO ontology development life cycle, by expanding and improving the ontology based on more cases of production processes for offsite construction.

9. Acknowledgments

This research was supported by an Innovate UK funded project (Project no. 104798) “Collaborative Knowledge-Based DfMA approach to building cost efficient, low impact and high performance houses”. The authors would like to thank the participating industrial partners of the research project including Walsall Housing Group, Hadley Group, QM systems, and Northmill Associates and Energy Systems Catapult.

References

- [1] R. Hairstans and R.E. Smith, Offsite HUB (Scotland): establishing a collaborative regional framework for knowledge exchange in the UK, *Architectural Engineering and Design Management* **14**(1–2) (2018), 60–77.
- [2] G. Boothroyd, Product design for manufacture and assembly, *Computer-Aided Design* **26**(7) (1994), 505–520.
- [3] W. Lu, T. Tan, J. Xu, J. Wang, K. Chen, S. Gao and F. Xue, Design for manufacture and assembly (DfMA) in construction: The old and the new, *Architectural Engineering and Design Management* **17**(1–2) (2021), 77–91.
- [4] H.W. Stoll, Design for manufacture: an overview (1986).
- [5] R. Bogue, Design for manufacture and assembly: background, capabilities and applications, *Assembly automation* (2012).
- [6] F.J. Emmatty and S. Sarmah, Modular product development through platform-based design and DFMA, *Journal of Engineering Design* **23**(9) (2012), 696–714.
- [7] K.G. Swift and N.J. Brown, Implementation strategies for design for manufacture methodologies, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* **217**(6) (2003), 827–833.
- [8] M. Arif, N. Blismas and R. Wakefield, Drivers, constraints and the future of offsite manufacture in Australia, *Construction innovation* (2009).
- [9] K.O. Ayinla, F. Cheung and A.-R. Tawil, Demystifying the concept of offsite manufacturing method: Towards a robust definition and classification system, *Construction Innovation* **20**(2) (2020), 223–246.
- [10] A.-Q. Gbadamosi, A.-M. Mahamadu, L.O. Oyedele, O.O. Akinade, P. Manu, L. Mahdjoubi and C. Aigbavboa, Offsite construction: Developing a BIM-Based optimizer for assembly, *Journal of Cleaner Production* **215** (2019), 1180–1190.
- [11] M.B. Murtaza, D.J. Fisher and M.J. Skibniewski, Knowledge-based approach to modular construction decision support, *Journal of Construction Engineering and Management* **119**(1) (1993), 115–130.
- [12] C. Pasquire, A. Gibb, N. Blismas et al., What should you really measure if you want to compare pre-fabrication with traditional construction?, in: *13th International Group for Lean Construction Conference: Proceedings*, International Group on Lean Construction, 2005, p. 481.
- [13] T. Tan, W. Lu, G. Tan, F. Xue, K. Chen, J. Xu, J. Wang and S. Gao, Construction-Oriented Design for Manufacture and Assembly Guidelines, *Journal of Construction Engineering and Management* **146**(8) (2020). doi:10.1061/(ASCE)CO.1943-7862.0001877.
- [14] J.J. Shi, D.-E. Lee and E. Kuruku, Task-based modeling method for construction business process modeling and automation, *Automation in Construction* **17**(5) (2008), 633–640.
- [15] K. Farghaly, F.H. Abanda, C. Vidalakis and G. Wood, BIM-linked data integration for asset management, *Built Environment Project and Asset Management* (2019).
- [16] J.E. Taylor and P.G. Bernstein, Paradigm trajectories of building information modeling practice in project networks, *Journal of management in engineering* **25**(2) (2009), 69–76.
- [17] A.A. HIJAZI and H.A. OMAR, Level of detail specifications, standards and file-format challenges in infrastructure projects for BIM level three, *WIT Transactions on The Built Environment* **169** (2017), 143–154.
- [18] B. Dave, A. Buda, A. Nurminen and K. Främling, A framework for integrating BIM and IoT through open standards, *Automation in Construction* **95** (2018), 35–45.
- [19] M. Mayouf, M. Gerges and S. Cox, 5D BIM: an investigation into the integration of quantity surveyors within the BIM process, *Journal of Engineering, Design and Technology* (2019).
- [20] J. Domingue, D. Fensel and J.A. Hendler, *Handbook of semantic web technologies*, Springer Science & Business Media, 2011.
- [21] L. Ding, K. Li, Y. Zhou and P.E. Love, An IFC-inspection process model for infrastructure projects: Enabling real-time quality monitoring and control, *Automation in Construction* **84** (2017), 96–110.
- [22] R. Masood, J.B. Lim and V.A. González, Performance of the supply chains for New Zealand prefabricated house-building, *Sustainable Cities and Society* **64** (2020), 102537.
- [23] E. Curry, J. O’Donnell, E. Corry, S. Hasan, M. Keane and S. O’Riain, Linking building data in the cloud: Integrating cross-domain building data using linked data, *Advanced Engineering Informatics* **27**(2) (2013), 206–219.
- [24] P. Pauwels, S. Zhang and Y.-C. Lee, Semantic web technologies in AEC industry: A literature overview, *Automation in Construction* **73** (2017), 145–165.

- [25] S. Gao, R. Jin and W. Lu, Design for manufacture and assembly in construction: a review, *Building research & information* **48**(5) (2020), 538–550.
- [26] L. Patil, D. Dutta, A.D. Bhatt, K. Jurrens, K. Lyons, M. Pratt, R.D. Sriram et al., Representation of heterogeneous objects in ISO 10303 (STEP), in: *ASME International Mechanical Engineering Congress and Exposition, Orlando, Florida, 2000*.
- [27] S. Grimm, A. Abecker, J. Völker and R. Studer, Ontologies and the semantic web, in: *Handbook of semantic web technologies*, 2011.
- [28] P. Pauwels and W. Terkaj, EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology, *Automation in Construction* **63** (2016), 100–133.
- [29] P. Pauwels and A. Roxin, SimpleBIM: From full ifcOWL graphs to simplified building graphs, in: *Proceedings of the 11th European Conference on Product and Process Modelling (ECPPM)*, 2016, pp. 11–18.
- [30] W. Terkaj and P. Pauwels, A Method to generate a Modular ifcOWL Ontology, in: *8th International Workshop on Formal Ontologies meet Industry*, Vol. 2050, 2017.
- [31] I. UK, Infrastructure and P. Authority, Creating a Digital Built Britain: what you need to know (2017).
- [32] J. Werbrouck, P. Pauwels, J. Beetz and L. van Berlo, Towards a decentralised common data environment using linked building data and the solid ecosystem, in: *Advances in ICT in Design, Construction and Management in Architecture, Engineering, Construction and Operations (AECO) : Proceedings of the 36th CIB W78 2019 Conference*, B. Kumar, F. Rahimian, D. Greenwood and T. Hartmann, eds, 2019, pp. 113–123. ISBN 9781861354877. <http://cibw78.northumbria-eee.co.uk/>.
- [33] M.H. Rasmussen, M. Lefrançois, G.F. Schneider and P. Pauwels, BOT: the Building Topology Ontology of the W3C Linked Building Data Group, *Semantic Web Journal* **0**(0) (2019).
- [34] B. Balaji, A. Bhattacharya, G. Fierro, J. Gao, J. Gluck, D. Hong, A. Johansen, J. Koh, J. Ploennigs, Y. Agarwal et al., Brick: Towards a unified metadata schema for buildings, in: *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, 2016, pp. 41–50.
- [35] L. Daniele, F. den Hartog and J. Roes, Created in close interaction with the industry: the smart appliances reference (SAREF) ontology, in: *International Workshop Formal Ontologies Meet Industries*, Springer, 2015, pp. 100–112.
- [36] T.E. El-Diraby, Domain ontology for construction knowledge, *Journal of Construction Engineering and Management* **139**(7) (2013), 768–784.
- [37] Y. Zheng, S. Törmä and O. Seppänen, A shared ontology suite for digital construction workflow, *Automation in Construction* **132** (2021), 103930.
- [38] A. Wagner and U. Ruppel, BPO: The building product ontology for assembled products, *CEUR Workshop Proceedings* **2389** (2019), 106–119.
- [39] S. Lemaignan, A. Siadat, J.-Y. Dantan and A. Semenenko, MASON: A Proposal For An Ontology Of Manufacturing Domain, in: *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, 2006, pp. 195–200. doi:10.1109/DIS.2006.48.
- [40] S. Lemaignan, A. Siadat, J.Y. Dantan and A. Semenenko, MASON: A proposal for an ontology of manufacturing domain, *Proceedings - DIS 2006: IEEE Workshop on Distributed Intelligent Systems - Collective Intelligence and Its Applications* **2006**(June 2014) (2006), 195–200. ISBN 076952589X. doi:10.1109/DIS.2006.48.
- [41] S. An, P. Martinez, R. Ahmad and M. Al-Hussein, Ontology-based knowledge modeling for frame assemblies manufacturing, *Proceedings of the 36th International Symposium on Automation and Robotics in Construction, ISARC 2019* (2019), 709–715. doi:10.22260/isarc2019/0095.
- [42] S. Staub-French, M. Fischer, J. Kunz and B. Paulson, An ontology for relating features with activities to calculate costs, *Journal of Computing in Civil Engineering* **17**(4) (2003), 243–254.
- [43] M. Zhiliang and W. Zhenhua, Framework for automatic construction cost estimation based on BIM and ontology technology, in: *Proceedings of the CIB W*, Vol. 78, 2012, p. 2012.
- [44] H. Abanda, J.H.M. Tah, M. Manjia, C. Pettang and F. Abanda, An ontology-driven house-building labour cost estimation in Cameroon, *Journal of Information Technology in Construction* **16**(March 2010) (2011), 617–634.
- [45] N. Li, Q. Li, Y.-S. Liu, W. Lu and W. Wang, BIMSeek++: Retrieving BIM components using similarity measurement of attributes, *Computers in Industry* **116** (2020), 103186. doi:10.1016/J.COMPIND.2020.103186.
- [46] S.K. Lee, K.R. Kim and J.H. Yu, BIM and ontology-based approach for building cost estimation, *Automation in Construction* **41** (2014), 96–105. doi:10.1016/j.autcon.2013.10.020.
- [47] H. Liu, M. Lu and M. Al-Hussein, Ontology-based semantic approach for construction-oriented quantity take-off from BIM models in the light-frame building industry, *Advanced Engineering Informatics* **30**(2) (2016), 190–207. doi:10.1016/j.aei.2016.03.001.
- [48] F. Ning, Y. Shi, M. Cai, W. Xu and X. Zhang, Manufacturing cost estimation based on a deep-learning method, *Journal of Manufacturing Systems* **54** (2020), 186–195.
- [49] H.V. Nguyen, I.D. Tommelein and G. Ballard, Process-based cost modeling to support lean project delivery, in: *16th annual conference of the international group for lean construction*, 2008, pp. 577–588.
- [50] N.F. Noy, D.L. McGuinness et al., Ontology development 101: A guide to creating your first ontology, Stanford knowledge systems laboratory technical report KSL-01-05 and . . . , 2001.
- [51] M. Rasmussen, P. Pauwels, C.A. Hviid and J. Karlshøj, Proposing a central AEC ontology that allows for domain-specific extensions, in: *Proceedings of the Joint Conference on Computing in Construction (JC3)*, 2017, pp. 237–244.
- [52] P. Doran, V. Tamma and L. Iannone, Ontology module extraction for ontology reuse: an ontology engineering perspective, in: *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2007, pp. 61–70.
- [53] A. Algergawy, S. Babalou, F. Klan and B. König-Ries, Ontology Modularization with OAPT, *Journal on Data Semantics* **9**(2) (2020), 53–83.
- [54] M.H. Rasmussen, M. Lefrançois, P. Pauwels, C.A. Hviid and J. Karlshøj, Managing interrelated project information in AEC Knowledge Graphs, *Automation in Construction* **108** (2019),

102956. doi:<https://doi.org/10.1016/j.autcon.2019.102956>.
<http://www.sciencedirect.com/science/article/pii/S0926580519300378>.
- [55] D. Vrandečić, Ontology evaluation, in: *Handbook on ontologies*, Springer, 2009, pp. 293–313.
- [56] M. Nickel, K. Murphy, V. Tresp and E. Gabrilovich, A review of relational machine learning for knowledge graphs, *Proceedings of the IEEE* **104**(1) (2015), 11–33.
- [57] M. Horridge and S. Bechhofer, The owl api: A java api for owl ontologies, *Semantic web* **2**(1) (2011), 11–21.
- [58] M. Rasmussen, P. Pauwels, M. Lefrançois, G.F. Schneider, C. Hviid and J. Karlshøj, Recent changes in the building topology ontology, 2017.
- [59] F. Barbosa, J. Woetzel, J. Mischke, M.J. Ribeirinho, M. Sridhar, M. Parsons, N. Bertram and S. Brown, Reinventing construction through a productivity revolution, *McKinsey Global Institute* (2017).
- [60] I.C. No, Specification for information management for the capital/delivery phase of construction projects using building information modelling (2013).
- [61] P. Pauwels, T. Krijnen, W. Terkaj and J. Beetz, Enhancing the ifcOWL ontology with an alternative representation for geometric data, *Automation in Construction* **80** (2017), 77–94.
- [62] M. Hepp, GoodRelations: An ontology for describing products and services offers on the web, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **5268 LNAI** (2008), 329–346. ISBN 3540876952. doi:10.1007/978-3-540-87696-0-29.
- [63] E.V. Kalemi, F. Cheung, A.-r. Tawil and P. Patlakas, ifcOWL-DfMA a new ontology for the offsite construction domain, *Proceedings of the 8th Linked Data in Architecture and Construction Workshop - LDAC2020 ifcOWL-DfMA* (2020), 105–117.
- [64] T. Steiner, R. Troncy and M. Hausenblas, How Google is using linked data today and vision for tomorrow, in: *LDFI 2010, Workshop on Linked Data in the Future Internet at the Future Internet Assembly, Volume 700, December 15-17, 2010, Ghent, Belgium, ISSN 1613-0073*, Ghent, BELGIUM, 2010. <http://www.eurecom.fr/publication/3284>.
- [65] J. Ashraf, R. Cyganiak, S. O’Riain and M. Hadzic, Open ebusiness ontology usage: Investigating community implementation of goodrelations, *CEUR Workshop Proceedings* **813** (2011), 1–11.
- [66] A. Borrmann, M. König, C. Koch and J. Beetz, *Building Information Modeling*, Springer International Publishing, 2018. doi:10.1007/978-3-319-92862-3.
- [67] *ISO 10303-11 Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual*, International Standardisation Organisation, Geneva, Switzerland, 2004.
- [68] C. Boje, A. Guerriero, S. Kubicki and Y. Rezgui, Towards a semantic Construction Digital Twin: Directions for future research, *Automation in Construction* **114** (2020), 103179.
- [69] I. Yitmen, The challenge of change for innovation in construction: A North Cyprus perspective, *Building and environment* **42**(3) (2007), 1319–1328.
- [70] A. Omigbodun, Value engineering and optimal building projects, *Journal of architectural engineering* **7**(2) (2001), 40–43.
- [71] S. Ashley, Cutting costs and time with DFMA, *Mechanical Engineering* **117**(3) (1995), 74–77.
- [72] T. Tan, W. Lu, G. Tan, F. Xue, K. Chen, J. Xu, J. Wang and S. Gao, Construction-Oriented Design for Manufacture and Assembly Guidelines, *Journal of Construction Engineering and Management* **146**(8) (2020), 04020085.
- [73] S. Kamaruddin, M. Mohammad, R. Mahbub and R. Yunus, Perception towards cost implication of mechanisation and automation approach in IBS projects in Malaysia, in: *29th Annual ARCOM Conference*, 2013.
- [74] Y. Jung and S. Woo, Flexible work breakdown structure for integrated cost and schedule control, *Journal of construction engineering and management* **130**(5) (2004), 616–625.
- [75] S.-L. Fan, C.-H. Wu, C.-C. Hun et al., Integration of cost and schedule using BIM, *Journal of Applied Science and Engineering* **18**(3) (2015), 223–232.
- [76] HM Government, 3-Digital Built Britain Level 3 Building Information Modelling - Strategic Plan, *UK Government* (2015), 1–47. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/410096/bis-15-155-digital-built-britain-level-3-strategy.pdf.
- [77] N.F. Noy, D.L. McGuinness et al., BCA, BIM design for design for manufacturing and assembly essential guide, Building and Construction Authority, 2016.
- [78] W. Pan and R. Sidwell, Demystifying the cost barriers to offsite construction in the UK, *Construction Management and Economics* **29**(11) (2011), 1081–1099.
- [79] K. Otto and K. Wood, *Techniques in Reverse Engineering, Systematic Design and New Product Development*, New York: Prentice Hall, 2001.
- [80] A. Wagner, L. Möller, C. Leifgen and U. Rüppel, SolConPro: Describing multi-functional building products using semantic web technologies, in: *EWork and eBusiness in architecture, engineering and construction*, CRC Press, 2018, pp. 447–455.
- [81] R.V. Guha, D. Brickley and S. Macbeth, Schema.org: evolution of structured data on the web, *Communications of the ACM* **59**(2) (2016), 44–51.
- [82] M. Bonduel, J. Oraskari, P. Pauwels, M. Vergauwen and R. Klein, The IFC to linked building data converter: current status, in: *6th Linked Data in Architecture and Construction Workshop*, Vol. 2159, 2018, pp. 34–43.
- [83] G. Boothroyd, *Assembly automation and product design*, CRC Press, 2005.
- [84] D.A. Steinhardt and K. Manley, Adoption of prefabricated housing—the role of country context, *Sustainable cities and society* **22** (2016), 126–135.
- [85] R. Masood, J.B.P. Lim and V.A. González, Performance of the supply chains for New Zealand prefabricated house-building, *Sustainable Cities and Society* **64** (2021), 102537. doi:<https://doi.org/10.1016/j.scs.2020.102537>. <http://www.sciencedirect.com/science/article/pii/S2210670720307538>.
- [86] A. Hajare and E. Elwakil, Integration of Life Cycle Cost Analysis and Energy Simulation for Building Energy-Efficient Strategies Assessment, *Sustainable Cities and Society* (2020), 102293.
- [87] A. Darko, A.P. Chan, Y. Yang and M.O. Tetteh, Building information modeling (BIM)-based modular integrated construction risk management—Critical survey and future needs, *Computers in Industry* **123** (2020), 103327.

- [88] T. Greif, N. Stein and C.M. Flath, Peeking into the void: Digital twins for construction site logistics, *Computers in Industry* **121** (2020), 103264.
- [89] M.C. Suárez-Figueroa, A. Gómez-Pérez and M. Fernández-López, The NeOn methodology for ontology engineering, in: *Ontology engineering in a networked world*, Springer, 2012, pp. 9–34.
- [90] K. Ayinla, E. Vakaj, F. Cheung and A.-R.H. Tawil, A Semantic Offsite Construction Digital Twin- Offsite Manufacturing Production Workflow (OPW) Ontology, 2021.
- [91] T. Berners-Lee, Linked data-design issues, <http://www.w3.org/DesignIssues/LinkedData.html> (2006).

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51