Creating Information Delivery Specifications using Linked Data

Léon van Berlo[1*] Peter Willems[1] and Pieter Pauwels[2]
[1]Netherlands organisation for applied scientific research TNO
[2]Ghent University, Dept. of Architecture and Urban Planning
* email: leon.vanberlo@tno.nl

Abstract

The use of Building Information Management (BIM) has become mainstream in many countries. Exchanging data in open standards like the Industry Foundation Classes (IFC) is seen as the only workable solution for collaboration. To define information needs for collaboration, many organizations are now documenting what kind of data they need for their purposes. Currently practitioners define their requirements often a) in a format that cannot be read by a computer; b) by creating their own definitions that are not shared. This paper proposes a bottom up solution for the definition of new building concepts a property. The authors have created a prototype implementation and will elaborate on the capturing of information specifications in the future.

Keywords: BIM, mvd, IFC, model checking, information delivery specification

1. Introduction

The use of Building Information Management (BIM) has become mainstream in many countries. Using BIM processes and tools, stakeholders in the Architecture, Engineering, and Construction (AEC) industry are able to model, design and engineer buildings and infrastructure in information-rich 3D models [Borrmann et al., 2018, Eastman et al., 2008, Sacks et al., 2018]. Further-more, they are able to exchange the associated information and achieve less error-prone and more efficient processes in the design and construction of assets in the built environment (buildings, bridges, etc.).

When using an open and neutral information exchange approach in this industry, the exchange of data is recommended to be modelled in Information Delivery Manuals [1][2] (IDMs), which specify Information Requirements (IRs) and Exchange Requirements (ERs) in Business Process Modelling Notation (BPMN)diagrams, text, and tables. These IDMs include specific exchanges, defined as Model View Definitions[3](MVDs). MVDs essentially represent what information needs to be delivered, when, and between which stakeholders.

The Industry Foundation Classes[4](IFC) serve as the de-facto industry standard for representing building information in a neutral format [ISO, 2013]. IFC is hence often recommended as a data format for the data representation of the information needs (what information) in an MVD. In other words, MVDs ideally represent a subset schema of the complete IFC schema in their purpose of defining exchanges (information needs).

This standardized way of working, starting from IDMs, is adopted in many countries, and definitely also in the Netherlands and Belgium. In practice, this often results in a PDF document that

[1] http://idm.buildingsmart.org
[2] https://technical.buildingsmart.org/resources/information-delivery-manual
[3] https://technical.buildingsmart.org/standards/mvd
[4] https://technical.buildingsmart.org/standards/ifc/

specifies the IDM. These IDMs often have different names, resulting in BIM Execution Plans, protocols, Delivery Manuals, and so forth, often depending on regions, languages, countries and local habits.

Yet, the main idea behind the IDM standard is often implemented in practice. Unfortunately, these IDMs seldom include tightly scoped MVDs, as specified in the original IDM and MVD standards, and they definitely seldom go in detail in terms of defining information needs with the IFC data model. Instead, most practitioners reside to defining (required) property sets with specific naming conventions for re-use within a project in a much more manual and ad-hoc manner. As a result, there is an overwhelming number of different properties and products defined within each project adopting an IDM workflow, leading to confusion and people lost in translation across projects.

This article looks into this situation and investigates to what extent the current situation may be addressed by the use of linked data technologies. Recently emerging linked data [Berners-Lee, 2006, Bizer et al., 2009, Heath and Bizer, 2011] and/or semantic web technologies [Berners-Lee et al., 2001, Domingue et al., 2011] enable decentralized information management over the web, and therefore, they might facilitate the bottom up definition and re-use of property sets over project borders. With linked data initiatives in the AEC industry aiming to make building concepts available in a modular, machine-readable, and decentralized fashion [Pan et al., 2004, Rezgui et al., 2011, Curry et al., 2013, Törmä, 2013,König et al., 2013, Pauwels, 2014, Pauwels et al., 2017, Rasmussen et al., 2018, Rasmussen et al., 2017], some of the aforementioned issues may be addressed using a linked data approach for the creation of information delivery specifications, which is the purpose of research reported in this article.

In Section 2, we provide a review of how MVDs and IDMs are specified in the Netherlands, thereby documenting the use of the 'BIM Base IDS' in practice. This section looks specifically in how diverse IDS documents are built, project per project. This leads to a number of identified issues in the process of creating IDMs and then requirements that should be met by any solution aiming to create IDMs. In Section 3, we document a state of the art review of technologies used in the creation of IDMs, including PDF, Object Type Libraries (OTLs), the buildingSMART Data Dictionary (bSDD), and Linked Building Data (LBD). Section 4 and 5 propose a solution for the creation of IDMs, including a prototype tool. The paper is finished with Results, Evaluation, Discussion, and Conclusions.

2. BIM Information Delivery Specification

To define information needs for collaboration, many organizations are now documenting what kind of data they need for their purposes. The 'Coordination View' is the most used MVD used in practice [van Berlo, 2019]. The Coordination View is very complex, and covers most of the 2x3 version of the full IFC schema. In practice, many of the mandatory elements are not modelled and therefore not available in the exported IFC data-set from authoring tools. This causes problems during coordination of a project, because information that project partners need, is missing in the data. To deal with this problem,13 contractors and buildingSMART Benelux started an initiative to create an MVD to deal with the most common basic elements of information exchange for coordination in IFC.

2.1 The BIM Base IDS

The 'Basic Information Delivery Manual' [5] (Base IDS) provides a solid base with basic requirements that are almost always necessary. Instead of including multiple smaller exchanges in individual MVDs, one main basic minimal MVD is thus provided by this Base IDS. At the time of writing, the Base IDS is adopted by hundreds of organizations and translated in almost 20 languages. The original Dutch title is 'BIM Base/Basic Information Delivery Specification'. During the translation, the current English term 'BIM Basic Information Delivery Manual' is used. During this translation two things happened:

1) The `Information Delivery Specification' was translated to `Information Delivery Manual'. Yet,

[5] https://www.bimloket.nl/BIMbasicIDM

the document is actually much more a Model View Definition then an Information Delivery Manual. The reason for this translation is that the use of Model View Definitions is still quite vague to end users.

2) The term `basis' was translated to `basic', while the Dutch term can also be translated to `base'. The original goal of the document was to create a `base' of requirements that are always in effect for almost every use-case of data exchange. Additional requirements would be built on top of this base. The Dutch term `basis' would have better been translated to `base' in the meaning of `foundation' to build specific information delivery specifications. Part 4.4 was intended to be the gateway to additional project specific requirements.

To represent the 'BIM Basis ILS' in a way it was originally intended, we will use the term 'BIM Base IDS' in this paper.

2.2 BIM Base IDS in practice

Since the launch and successful use of the BIM Base IDS, many organizations have adopted it in their data requirements. Its success sparked the creation of domain specific `Information Delivery Specifications' as well. The Dutch limestone suppliers created a `National Information Delivery Specification for Limestone in the Netherlands'. This limestone IDS was an extension of the BIM Base IDS. Other initiatives have built a specific IDS that conflicts with the requirements from the Base IDS. The authors will not elaborate on these initiatives but like to remark that this could harm the effectiveness and productivity of data exchange in the industry.

The Base IDS is meant as a base, and additional requirements are meant to be built on top of it. At the moment, there are a few issues arising from the use of this approach:

- Issue 1: Most of the additional requirements are defined in unstructured formats (e.g. PDF) or human-readable form with no tools to check data against requirements.
- Issue 2: Comparable requirements, like the capacity of an elevator, are not shared, so there is a welter of equal definitions that are being required in different formats.
- Issue 3: Already existing definitions in other domains (CityGML, gbXML, etc) are being introduced in IFC.

Many extensions to the Base IDS introduce various additional concepts and requirements, often in an unstructured format, and often already existing in other domains. This leads to concepts being defined multiple times, in the worst case in non-machine-readable formats, which in turn leads to confusion, debates and therefore considerable loss of efficiency in the AEC industry.

2.3 Requirements in the creation of an IDS

To tackle the issues identified above, a solution needs to adhere to the following requirements (in order of priority):

1. There is a need to formalize data requirements in a machine-readable way to be able to automate compliance checking;
2. There is a need to make existing building-related concepts and properties available to end users so that they can select and re-use what they need;
3. There is a need to be able to create new building-related concepts and properties when the currently defined are not suitable for the specific case of the end user;
4. There is a strong request to standardize and re-use common building-related concepts and properties in specific domains or regions.
5. There might be a good reason to re-use building-related concepts and properties from other domains (like CityGML, gbXML, etc) to avoid `re-inventing' or copying them into IFC.

These requirements and features will be the benchmark during the review of the current state of the art.

3. Review of current solutions and state of the art

In reviewing the state of the art, we intend to cover mainly the current tools and data models available and/or used for specifying information requirements that are part of an IDM. This includes PDF documents, which is the most commonly used format to represent the IDM itself, and which typically identify lists of custom PropertySets. Furthermore, the buildingSMART realm of IFC, mvdXML, and bSDD is briefly covered. Third and fourth, we discuss the use of Object Type Libraries (OTLs) and Linked Data technologies.

3.1 PDFs and lists of custom PropertySets

Currently the BIM Base IDS, the Information Delivery Specifications built on top of the BIM Base IDS, and the dialect IDS documents are all distributed as PDF documents. This makes the result very accessible for users, and therefore stimulates people to think about their data requirements. However, this approach also provokes fuzzy definitions and uncertainties. Requirements defined this way are also not machine-readable. Automated compliance checking of a data-set against the requirements set in the Base IDS needs a custom made solution every time.

A direct request from users that create Information Delivery Specifications in PDF, was to find a solution to re-use comparable definitions that are needed in different projects. For example, this happened with the definition of the load capacity of an elevator in two projects in the Netherlands. Both projects needed to work with a definition for the load capacity that was different from the standard definitions for *CapacityByWeight* and *CapacityByNumber* as defined in the IFC Schema, and the used BIM authoring tool could not export the IFC data-set with these standardized properties for elevators.

In the first project, the load capacity of an elevator is defined as a property `Capacity' in a separate PropertySet called `Facility Management' linked to the *IfcTransportElement* object. In the second project, the same property, with basically the same semantic meaning is defined as a combination of two separate properties called `Load capacity', in a different custom made PropertySet called `IDS'[6].

3.2 IFC, mvdXML, and bSDD

Using mvdXML, it is possible to define data exchanges in a machine-readably member, thereby picking from what is needed from IFC. Yet, while IFC is the de-facto standard for information exchange in the industry, it does not cover every element that the industry uses. The IFC Schema is focused on defining and standardizing elements that are most commonly being exchanged between different partners in the industry. It can be seen as the largest common denominator for exchange of data in the AEC industry. Yet, specific elements like gypsum board elements are not semantically defined and standardized in IFC, because these have not been identified as elements that need to be in the `largest common denominator' for data exchange.

It was identified that in almost every case, it should be possible to extend the IFC definitions with additional elements for specific cases. There are several ways in which this is currently possible to work with in practice:

1. Using IfcProxyElement
2. Using PropertySets
3. Referring to elements in the bSDD
4. Referring to elements in an external object type library

This paper will not elaborate on the several options. The conclusion of this section is that IFC is the standardized, accepted schema for most common elements, and there are several ways to extend the IFC Schema in practice with elements that are specific for a discipline or a region.

At the moment, buildingSMART is controlling the creation of new objects in the bSDD. It is a top

[6] it was actually named after the name of the building, which is changed for publication purposes.

down process with the intention to control the redundancy of definitions, manage overlap of definitions and check the inheritance structures. Besides the extension of IFC, the buildingSMART Data Dictionary (bSDD) is also used for different purposes. It is seen as a mapping between concepts, to identify objects in the built environment and their specific properties regardless of language.

3.3 Object Type Libraries (OTLs)

Because the definition of concepts and elements has a different meaning per region and discipline, many national organizations have started to create their own `Object Type Libraries' (OTLs). These object type libraries have the same intention as bSDD: to extend the definitions in IFC. In some cases, national object type libraries have a mapping to the bSDD, but in most cases they are separate and external to buildingSMART altogether. There are even cases where the libraries do not use IFC as a reference, but build all concepts from scratch, or based on another national data standard.

Similar to the bSDD, also these object type libraries are often managed in a top down approach. The concepts and properties in these initiatives are governed by a team of people and processes with the intention to create a controllable standard to refer to.

Looking at the requirements for our solution, the bSDD and most object type libraries do not comply with the need to `be able to create new building-related concepts and properties when the currently defined ones are not suitable for the specific case of the end user'. Our solution needs to be more bottom up, with the ability for users to define new concepts and properties that cannot be found in the current libraries.

3.4 Linked Building Data

As indicated in the introduction, linked data technologies are impacting on the AEC industry as a set of technologies that allows to specify vocabularies and data in a much more decentralized manner. The defined data is fully machine-readable, therefore, the linked data technologies might be perfect for addressing the above need(s). Therefore, an OWL version of IFC has been created, aiming to open up the use of these technologies for the AEC industry. As a result, alternative to the EXPRESS schema of IFC, also XML-based and RDF-based serializations of IFC are available, dubbed ifcXML and ifcOWL [Beetz et al., 2005, Pauwels and Terkaj, 2016] respectively.

Suggestions have been made to make the ifcOWL ontology (1) simpler [Mendes de Farias et al., 2015,Pauwels and Roxin, 2016], (2) more modular [Terkaj and Pauwels, 2017], and (3) more easily extensible [Beetz et al., 2014]. Of those three aims (simplicity, modularity, extensibility), one of the more central aims is to make building data available in a modular fashion. Therefore, there is an ambition to make building data available on the web, using modular ontologies that can be combined as wished [Schneider, 2017, Schneider et al., 2018]. This ambition is pushed most predominantly by the W3C LBD CG and closely follows well-known linked data best practices [Lóscio et al., 2016] (see also [Rector, 2003]). This inherently modular approach allows extensibility of the schema by anyone for any purpose. This might be a solution to the bottom up approach that is needed in our solution, and is missing in the current bSDD and known object type libraries.

Of central importance to the current aim to be able to define custom product types and associated properties, is the Ontology for Property Management(OPM), which has been proposed by Rasmussen et al., 2018.This ontology allows to represent properties in general and manage the changes to those properties over time. The same topic has been discussed at length within the W3C LBD CG, under the PROPS topic (e.g. [Bonduel, 2018]). Al-though the W3C LBD CG now does not formally support or acknowledge any Properties (PROPS) or OPM ontology, the overall approach used for representing properties and their management over time, is well-known and informally very well accepted. This overall approach includes a number of ‘levels’. As indicated in Rasmussen et al., 2018, these levels refer to the number of steps/relations between a product and the node (literal or individual)that encodes the value of its property. Listing 1 shows what this means for Level 1, as also explained at length in Rasmussen et al., 2018.

Listing 1: Property Level 1 (adapted from Rasmussen et al 2018)

```
ex:thermalTransmittance a owl:DatatypeProperty .
ex:material a owl:ObjectProperty .
inst:wallA ex:thermalTransmittance "0.27 W/(m2.K)" .
inst:wallA ex:material ex:Concrete .
```

Level 2 and Level 3 properties allow to add more metadata to the property definitions. For example, storing properties in a Level 2 design pattern allows to store units, value, timestamp, author, and so forth, directly with a generic node that represents the property value (see Listing 1). Property definitions in level 3 allow to define even more metadata, which allows the detailed property management with property states and changes over time, which is out of scope here.

Listing 2: Property Level 2 (adapted from Rasmussen et al 2018)

```
ex:thermalTransmittance a owl:ObjectProperty .
ex:material a owl:ObjectProperty .
inst:wallA ex:thermalTransmittance inst:PropertyX .
inst:wallA ex:material inst:PropertyY .
inst:PropertyX ex:value "0.27" .
inst:PropertyX ex:unit "W/(m2.K)" .
inst:PropertyY ex:name "Concrete"@en .
```

The outlined methods for defining properties in Level 1 and Level 2 property definitions can also be used to define the properties and property sets used in information delivery specifications. Essentially, these are generic methods which can also be used in the definition of a bSDD and/or OTLs. A bottom-up approach is also possible. This leads to the following key questions:

- Should properties be defined in Level 1 or 2?
- Where does property (set) naming happen (e.g. thermal transmittance)?
- Are end users able to define their own properties and property sets?

3.5 Conclusions State of the art

The authors conclude that there is no integral solution for end users to define an Information Delivery Specification that adheres to the mentioned requirements (Section 2.3). The current approach to define PDF documents with requirements is not machine readable. The current solutions surrounding object type libraries (and bSDD) are focused on a top down approach. The state of the art in Linked Data has potential, but is still very experimental. Since linked data is the most obvious technology for extending data schemas, in a bottom up approach, with the ability to combine definitions from different domains, this technology is chosen to build a new solution. The fact that these technologies can also be used in the definition of properties and products in the bSDD and in OTLs, makes this approach usable not only in a bottom-up, but also a top-down approach.

4. The proposed solution

Our proposed system to define requirements using a linked data approach consists of two parts:

- A library to create and share property(set) definitions, with a SPARQL [Steve Harris and Andy Seaborne, 2013]and GraphQL[7] interface.
- A tool to combine definitions from IFC, self-created properties and external resources (Section 5). The tool generates SPARQL [Steve Harris and Andy Seaborne, 2013],mvdXML, JSON and PDF [Manola et al., 2015] to share information requirements.

[7] https://graphql.org

4.1 Creating a user IDS

The envisioned use of the solution for creating a user Information Delivery Specification (IDS) is as follows. A user logs into the system. The user decides to define a new IDS. The system suggests some base modules that are popular. These modules consist of a list of required concepts, properties and modelling guidelines. These modules could be those of the `BIM Base IDS', but also base modules like `2nd order space boundaries' or some other requirement modules. The user selects a couple of these modules to start her own IDS.

In the current prototype, when creating a new user defined IDS from selected existing IDS modules, the requirements defined in those modules are copied to the newly created user IDS. In this way, the user can change the contents of the chosen existing modules, without affecting specifications from other users. It has to be discussed to what extent this is desirable, because changes in the imported modules, e.g. the BIM Base IDS might be counter-effective, or conflicts with other requirements might arise.

4.2 Adding required properties to the IDS

After selecting requirements into a new user IDS, a user can add additional requirements.

End users should be able to log in to a system where they can search for already defined properties, user defined properties, and properties in domains other than IFC.

The first prototype of the proposed system will focus on defining Properties for the already available concepts in IFC.

A user can search for properties in a library. This library consists of the buildingSMART PSet properties and available user-defined properties. The list of search results shows known buildingSMART properties at the top, and existing user-defined properties below. User-defined properties are sorted based on popularity, where popularity is defined by the number of times the property definition is used in any user-created IDS. Finding existing property set definitions follows the same process as finding existing properties. Ideally, the system differentiates between property sets that can be extended and property sets that are fixed (e.g. by the IFC schema).

When the user finds an existing (IFC) property, it can be added to the IDS.

Any property needs to be part of a property set. The definition of properties occurs in the Level 1 and Level 2 ontology design patterns that were documented in Section 3.4. In defining her IDS requirements, an end user is able to search for, generate, and store properties and property sets in a library.

When selecting a property that is part of a property set, the user can define what properties of that property set are mandatory or optional in her IDS.

4.3 Defining new properties

When a user cannot find an existing property, a new one can be created. When creating a new property this property can also be set to mandatory or optional in the IDS.

Newly created definitions are defined in a public library. Each newly created property has a specific URI that is related to the user that created it. After generating a new property, it is stored in the library component. The properties and property sets in this library are publicly available through a SPARQL and GraphQL interface, and of course through the IDS generation tool. These properties can be selected to be used in a (user defined) IDS. User created IDSs are stored separately, and only available for the logged in user.

4.4 Using the IDS

After selecting properties and other requirements, the IDS will be saved and linked to the user profile. A user can export the IDS to PDF, mvdXML, SparQL or a self-defined JSON format.

The vision is to be able to load these machine readable formats into BIM Authoring tools, to make sure all information requirements and properties are defined before exporting to IFC. This will help end-users that are modelling BIM data to make sure they comply with the information requirements, and that the IFC export is valid.

Future development could include the ability to select concepts or properties from other domains like gbXML or CityGML. This will enable interoperability over domains. This will complicate the potential feature to load the IDS in authoring tools to create a valid IFC export.

5. The Development

The prototype was developed in several steps:

1. Translating the IFC PSet Definition XML-schema to OWL
2. Converting the IFC PSet Definition XML documents to RDF
3. Harness a SPARQL engine with the generated PSet Definition RDF triples
4. Embed the SPARQL endpoint in a GraphQL interface
5. Develop the IDS definition tool in a web browser application

The next subsections will describe these steps in more detail.

5.1 Translating the IFC PSet Definition XML-schema to OWL

Since the IFC property set definitions are XML documents specified conform an XML schema[8], the first step was to translate the schema into an OWL ontology with equivalent semantics and references to the ifcOWL ontology. The code snippet below shows the OWL class specification of a property set definition[9].

Listing 3: PSet Definition example

```
PSD:PropertySetDef
  rdf:type owl:Class ;
  rdfs:comment "Top node element of PSD."@en ;
  rdfs:label "Property set definition"@en ;
  rdfs:subClassOf owl:Thing ;
  rdfs:subClassOf [
      rdf:type owl:Restriction ;
      owl:allValuesFrom ifc:IfcProduct ;
      owl:onProperty PSD:applicableClass ;
    ] ;
  rdfs:subClassOf [
      rdf:type owl:Restriction ;
      owl:maxCardinality "1"^^xsd:nonNegativeInteger ;
      owl:onProperty PSD:definition ;
    ] ;
  rdfs:subClassOf [
      rdf:type owl:Restriction ;
      owl:maxCardinality "1"^^xsd:nonNegativeInteger ;
      owl:onProperty PSD:name ;
    ] ;
.
```

The information delivery specification entities have been added to this ontology as well to be able to group the required property sets and marking the mandatory properties of those property sets (Figure 1).

[8] http://www.buildingsmart-tech.org/xml/psd/PSD_IFC4.xsd
[9] https://app.informationdeliveryspecification.org/psets/psetdef.ttl

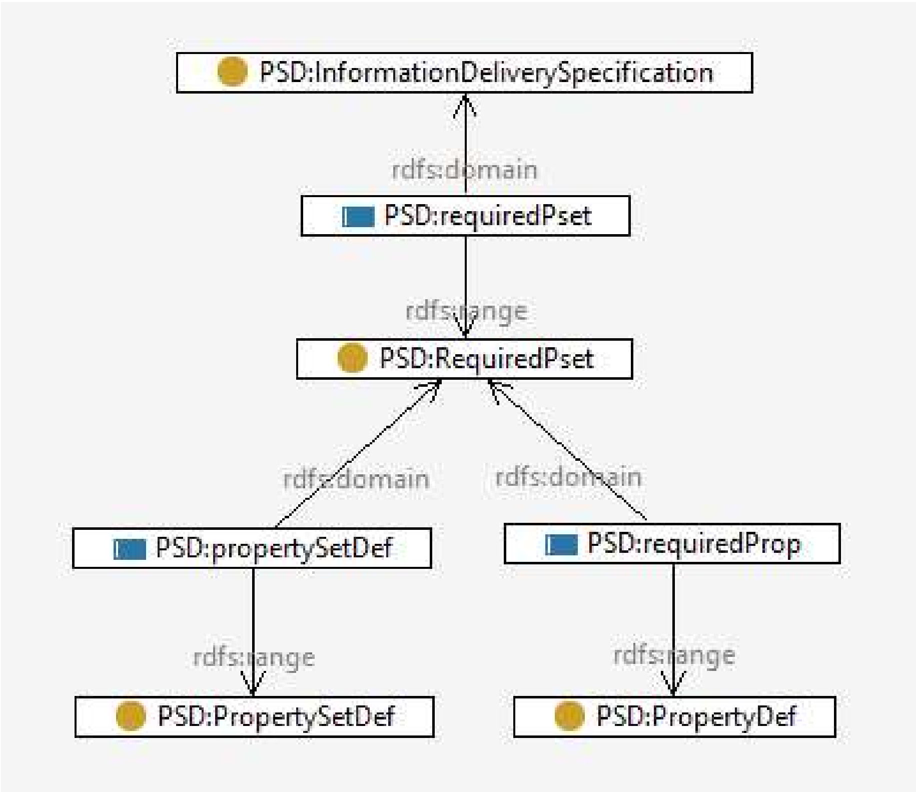

Figure 1: The Information delivery specification ontology

5.2 Converting the IFC PSet Definition XML documents to RDF

In this step, the IFC Pset definitions[10] are serialized to RDF conforming the ontology of the previous subsection. This results in property (set) definitions as displayed in Listing 4 for the BeamCommon PSET example.

Listing 4: PSet BeamCommon example

```
:Pset_BeamCommon
  rdf:type PSD:PropertySetDef ;
  PSD:applicableClass ifc:IfcBeam ;
  PSD:applicableTypeValue "IfcBeam" ;
  PSD:definition "Properties common to the definition of all occurrence and type
objects of beam." ;
  PSD:ifcVersion [
    rdf:type PSD:IfcVersion ;
    PSD:version "IFC4" ;
  ] ;
  PSD:name "Pset_BeamCommon" ;
  PSD:propertyDef :p04abf900d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p0970ad00d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p19888c80d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p1ee5d700d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p23aa8b00d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p286f3f00d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p2f964d00d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p33c26a80d1c411e1800000215ad4efdf ;
  PSD:propertyDef :p38871e80d1c411e1800000215ad4efdf .
```

[10] http://www.buildingsmart-tech.org/ifc/IFC4/Add2/html/psd/

5.3 Harness a SPARQL engine with generated Pset Definition RDF triples

Each IFC PropertySet definition is loaded into a triple store as a separate named graph. Named graphs facilitate configuration management especially for the class of custom property set definitions. This RDF triple store forms the public library of properties used by the proposed tool (part 1 defined in Section 4).

5.4 Embed the SPARQL endpoint in a GraphQL interface

Although a SPARQL endpoint is available for querying the properties in the library, many applications rely on alternative means to query for data, such as GraphQL. In this step, a GraphQL interface is added on top of the SPARQL endpoint, thus simplifying the development of web clients. Using GraphQL, a web client can precisely specify its data needs while preventing to receive unnecessary data or, on the other hand, repeatedly has to query for potentially missing data. Figure 2 shows a number of web services that can be executed to receive the necessary data based on GraphQL, e.g. allPSDs, allPDs, searchPD, etc.

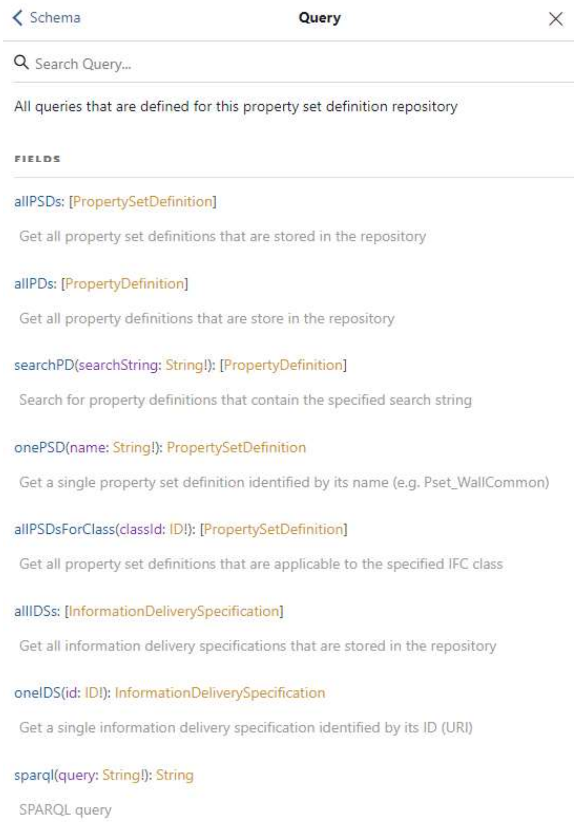

Figure 2: GraphQL query schema

5.5 Develop the IDS definition tool in a web browser application

A web browser application was developed to demonstrate the idea of sharing information delivery specifications that refer to both standard IFC and user defined property sets. Figure 3 shows an example IDS that includes IFC-based property sets only.

Basic information delivery manual

ID

http://openbimstandards.org/information-delivery-specification/Basic_IDM#Basic_IDM

PSET	mandatory properties
Pset_BeamCommon	FireRating, IsExternal, LoadBearing
Pset_ColumnCommon	FireRating, IsExternal, LoadBearing
Pset_DoorCommon	FireRating, IsExternal
Pset_RoofCommon	FireRating, IsExternal, LoadBearing
Pset_SlabCommon	FireRating, IsExternal, LoadBearing
Pset_StairCommon	FireRating, IsExternal, LoadBearing
Pset_WallCommon	FireRating, IsExternal, LoadBearing
Pset_WindowCommon	FireRating, IsExternal

Figure 3: Simple Information delivery manual

6. Results and Evaluation

The result of our work is two-fold: on the one hand, a preset library with properties and property set definitions is available, in the L1 and L2 property definition patterns established within the LBD Community Group. This library mostly covers the properties that are part of the property sets of IFC specifications. Second, a tool is available that allows to re-use existing and define new properties, so they can be used in IDS specifications.

Unfortunately, the tool is not stable enough for full beta testing with end users; only alpha testing (unit tests) has been performed. In addition, the prototype was presented to groups of industry experts, linked building data professionals, and potential end-users. Responses have been enthusiastic and all respondents see the potential. Based on the responses, the authors conclude that the chosen solution approach is one with potential. To fully test the solution, further developments are needed.

In terms of evaluation results, the demonstrations started a debate about the top down definitions of properties and concepts, versus the bottom up approach of the solution. Industry experts that are currently involved in standardization initiatives, are resistant to the bottom up approach of the chosen solution. Linked data experts and end users praise the bottom up approach and see much potential for the solution. Linked data experts focus on the distributed character of the solution. End users focus on the feature that makes it possible to find `the most often used properties in other IDSs'. This feature is only possible because this system centralizes the creation of different Information Delivery Specifications. If the same PSET ontology remains to be used, and the L1 and L2 property definition patterns are maintained as well, it should however be possible to implement the same system in a more distributed manner, including at least a federated query architecture.

During the observations, a big confusion was noticed about the definitions and boundaries of what an IDM and an MVD is, and what an IDS is. This confusion is most likely caused by a lack of knowledge about the definitions of IDMs and MVDs in the buildingSMART context, and a lack of open implementations of both which have a direct impact on the end user. The authors advise that this has to be resolved to facilitate the use of IFC in an effective way in the future, and with our approach, we make an important step forward in bringing MVDs, IDMs, and IDSs to the end user in a practical format.

7. Conclusions

This paper looked into the creation of Information Delivery Specifications (IDSs) in a bottom up and structured format. It hereby investigated the use of linked data technologies for the creation of property set and property definitions. Eventually, a library of those definitions is made available, together with an IDS creation tool which allows the creation of IDSs that include the mentioned property (set) definitions. With these definitions and formally correct IDSs and a referenceable library, a big step forward can be made in the formal specification of work processes and exchanges in the AEC industry.

In our work, we indicated both top-down and bottom-up approaches towards the definition of properties and property sets. The bottom up approach that allows end users to define their own properties is praised as the best way forward by many respondents in our evaluation. The library of newly defined properties is published as linked data and available for everyone to re-use. As such, it can be considered as a public bottom-up data dictionary based on linked data principles and formalisation with SPARQL, JSON, and GraphQL. Publishing the library with a SPARQL and GraphQL interface makes it very accessible.

Because of its bottom-up approach, this solution approach delivers interoperability for the industry without the need for long (top down) standardization procedures, yet it does not stand in the way of top down approaches either. In fact, the tool can be used to collect community feedback on often used and needed properties and property sets, which can inform top-down approaches. As such, the tool is intended to stimulate re-use, and potentially standardize, concepts and properties that are often used in practice.

Acknowledgements

This work was part of the BIM4REN project, which has received funding from the European Union's Horizon2020 Research and Innovation Program under grant agreement No 820773.

References

ISO 16739 - 2013: Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries. Technical report, International Organization for Standardization, Geneva, CH.

Beetz, J., van den Braak, W., Botter, R., Zlatanova, S., and 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 eWork and eBusiness in Architecture, Engineering and Construction, pages 135–140. CRC Press.

Beetz, J., van Leeuwen, J., and de Vries, B. (2005). An ontology web language notation of the industry foundation classes. In Proceedings of the 22nd CIB W78 Conference on Information Technology in Construction, pages 193–198.

Berlo, L. van (2019). BIM Analytics http://bimsupporters.com (Last Accessed May 2019)

Berners-Lee, T. (2006). Linked data - design issues. http://www.w3.org/DesignIssues/LinkedData.html (Last Accessed 24 Au-gust 2016).

Berners-Lee, T., Hendler, J., and Lassila, O. (2001).The Semantic Web. Scientific American, 284(5):34–43.

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data-the story so far. International journal on Semantic Web and Information Systems, 5(3):1–22.[Bonduel, 2018] Bonduel, M. (2018). Towards a PROPS ontology.

Borrmann, A., König, M., Koch, C., and Beetz, J.(2018).Building Information Modeling:

Technology Foundations and Indus-try Practice. Springer, 1 edition.

Curry, E., O'Donnell, J., Corry, E., Hasan, S., Keane, M., and O'Riain, S. (2013). Linking building data in the cloud: Integrating cross-domain building data using linked data. Advanced Engineering Informatics,27(2):206–219.

Domingue, J., Fensel, D., and Hendler, J. A. (2011).Handbook of semantic web technologies. Springer.

Eastman, C. M., Teicholz, P., Sacks, R., and Liston, K.(2008).BIM handbook: a guide to building information modeling for owners, managers, architects, engineers, contractors, and fabricators. John Wiley &Sons, Hoboken, NJ, USA.

Heath, T. and Bizer, C. (2011). Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web: Theory and Technology, 1(1):1–136.

König, M., Dirnbek, J., and Stankovski, V. (2013). Architecture of an open knowledge base for sustainable buildings based on linked data technologies. Automation in Construction, 35:542–550.

Lóscio, B. F., Burle, C., and Calegari, N. (2016). Data on the web best practices.W3C Working Draft, 19.

Manola, F., Miller, E., and McBride, B. (2015). RDF 1.1 primer, 2014. Working group note, World Wide Web Consortium.

Mendes de Farias, T., Roxin, A., and Nicolle,C. (2015). IfcWoD, semantically adapting IFC model relations into OWLproperties. In Proceedings of the 32nd CIB W78 Conference on Information Technology in Construction.

Pan, J., Anumba, C., and Ren, Z. (2004). Potential application of the semantic web in construction. In Proceedings of the 20th Annual Conference of the Association of Researchers in Construction Management(ARCOM), pages 923–929.

Pauwels, P. (2014). Supporting decision-making in the building life-cycle using linked building data. Buildings, 3:549–579.DOI:10.3390/buildings4030549.

Pauwels, P. and Roxin, A. (2016). SimpleBIM: From full ifcOWL graphs to simplified building graphs. In eWork and eBusiness in Architecture, Engineering and Construction, pages 11–18, Limassol, Cyprus. CRC Press.

Pauwels, P. and Terkaj, W. (2016). EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology. Automation in Construction, 63:100–133.

Pauwels, P., Zhang, S., and Lee, Y.-C. (2017). Semantic web technologies in AEC industry: A literature overview. Automation in Construction, 73:145–165.

Rasmussen, M. H., Lefrancois, M., Bonduel, M., Hviid,C. A., and Karlshøj, J. (2018). OPM: An ontology for describing properties that evolve over time. In 6th Linked Data in Architecture and Construction Workshop (LDAC), CEUR Workshop Proceedings, volume 2159, pages 23–33, UK.

Rasmussen, M. H., Pauwels, P., Hviid, C. A., and Karlshøj, J. (2017). Proposing a Central AEC Ontology That Allows for Do-main Specific Extensions. In Lean and Computing in Construction Congress- Volume 1: Proceedings of the Joint Conference on Computing in Construction, number July, pages 237–244, Edinburgh. Heriot-Watt University.

Rector, A. L. (2003). Modularisation of domain ontologies implemented in description logics and related formalisms including owl. In Proceedings of the 2nd international conference on

Knowledge capture, pages121–128. ACM.

Rezgui, Y., Boddy, S., Wetherill, M., and Cooper, G.(2011). Past, present and future of information and knowledge sharing inthe construction industry: Towards semantic service-based e-construction Computer-Aided Design, 43(5):502–515.

Sacks, R., Lee, C. M. E. G., and Teicholz, P. (2018). BIMhandbook: a guide to building information modeling for owners, managers, architects, engineers, contractors, and fabricators. John Wiley & Sons, Hoboken, NJ, USA, 3 edition.

Schneider, G. F. (2017). Towards aligning domain ontologies with the Building Topology Ontology. In5th Linked Data in Architecture and Construction Workshop, University of Burgundy, Dijon, France.

Schneider, G. F., Rasmussen, M. H., Bonsma, P.,Oraskari, J., and Pauwels, P. (2018). Linked Building Data for Modular Building Information Modelling of a Smart Home. In eWork and eBusiness in Architecture, Engineering and Construction, pages 407–414, Copenhagen, Denmark. CRC Press.

Steve Harris and Andy Seaborne(2013). SPARQL 1.1 Query Language.

Terkaj, W. and Pauwels, P. (2017). A method to generate a modular ifcOWL ontology. In Proceedings of the 8th International Workshop on Formal Ontologies meet Industry, volume 2050 of CEUR Workshop Proceedings, Bolzano, Italy.

CEUR-WS.org. Accessed September 2018.

Törmä, S. (2013). Semantic linking of building information models. Proceedings - 2013 IEEE 7th International Conference on Semantic Computing, ICSC 2013, pages 412–419.