An Improved Approach for Effective Describing Geometric Data in ifcOWL through WKT High Order Expressions

Dongming Guo[a], Erling Onstein[b] and Angela Daniela La Rosa[c]
Department of Manufacturing and Civil Engineering, Norwegian University of Science and Technology, Gjovik, Norway

Keywords: ifcOWL, Well-Known Text (WKT) Expressions, IFC Schema, Geometry Data, Shape Representation.

Abstract: Building Information Models (BIM) are considered as building digital representations, including comprehensive geometric and non-geometric information. For improving BIM interoperability, the semantic related technologies have been the one of main approaches for processing BIM data. Currently, ifcOWL is a recommended Web Ontology Language (OWL) representation of the Industry Foundation Classes (IFC) schema. When BIM geometric data is translated into ifcOWL representations, the excessive number of triples will be produced, and the generated Resource Description Framework (RDF) file will also be extremely bigger than the IFC original file. For generating concise geometric representation in Semantic Web context, Well-known text (WKT) has been widely used to describe BIM geometry data in ifcOWL. However, to avoid losing semantic information, only some simple pre-existing WKT expressions (Point or LineString) are used to describe BIM geometric aggregated data in semantics context. For solving this issue, we propose an improved approach that can represent BIM geometric data in ifcOWL ontology through WKT high order expressions. This representation can not only take full advantage of pre-existing WKT expressions to generate a more concise RDF representation, but also reduce the loss of semantic information.

1 INTRODUCTION

Recent years, Building Information Model (BIM) is widely used in Architecture, Engineering and Construction (AEC) industry as digital representations and repository of building information (Zhao 2017). To facilitate information sharing and interoperability in AEC industry, a data model with neutral platform and open file format, Industry Foundation Classes (IFC) (ISO 2013), is developed by buildingSMART organization. Along with the spread and development of IFC, IFC has already been a common data schema and try to cover the entire AEC industry (Laakso and Kiviniemi 2012). BIM cases based on IFC schema can contain geometry information of all building elements, such as 3D shape and the enclosed spaces, and non-geometric semantic information, such as the properties of the elements and the relationships between them.

Although IFC shows certain capabilities of implementing information sharing/exchange and improving interoperability in AEC industry, semantic clarity is not achieved in IFC that may result in non-efficient data exchange among different applications/stakeholders (Zhong et al. 2019). Additionally, semantic web and linked data technologies also promote the presentation of BIM data in a comprehensible form, especially in a machine-understandable form (ontological form). So, a recommended Web Ontology Language (OWL) representation of the IFC schema, ifcOWL, is also developed and standardized by buildingSMART (buildingSMART 2019a). The ifcOWL has been an open ontology for representing building data in Resource Description Framework (RDF) format. It also accelerates to process BIM data by semantic approaches in diversified engineering applications (Zhong et al. 2019).

Geometry data in BIM is one of important parts of building data, and the converting building geometry

[a] https://orcid.org/0000-0001-5357-8410
[b] https://orcid.org/0000-0002-5447-7563
[c] https://orcid.org/0000-0002-5739-6005

Guo, D., Onstein, E. and Rosa, A.
An Improved Approach for Effective Describing Geometric Data in ifcOWL through WKT High Order Expressions.
DOI: 10.5220/0010532302290236
In *Proceedings of the 7th International Conference on Geographical Information Systems Theory, Applications and Management (GISTAM 2021)*, pages 229-236
ISBN: 978-989-758-503-6

data into RDF format also needs to be considered for supporting related geometric processing and applications in Semantic Web context. Currently, most research mainly focuses on non-geometric data, and the description of construction-related geometric data in Semantic Web context is still a challenge and the uniform or general recommendation has not been achieved (Wagner et al. 2020). Hence, in the semantic web context, the processing of geometry data generally requires special attentions because diverse geometry descriptions may be used in different processing approaches based on specialized geometry ontologies (McGlinn et al. 2019). Wagner et al. (Wagner et al. 2020) summarized and analyzed approaches of geometry descriptions in Semantic Web context into four groups and evaluated the four groups with six aspects: semantic expressivity, flexibility, conciseness, simplicity, support, portability and extensibility. In their evaluations, the third group (using a Semantic Web approach for linking and storing geometry descriptions and other technologies for expressing geometry content and structure) showed out the more advantages in six aspects, in which Well-Known-Text(WKT) has been the most widely used to express geometry data as RDF literals (Wagner et al. 2020). WKT can represent several geometric objects, such as Point, MultiPoint, LineString, MultiLineString, Polygon, MultiPolygon, etc. However, in this kind of approaches, only some limited pre-existing WKT expressions are used in Semantic Web context to express the limited geometric data, because semantic information will be lost when some WKT high order expressions (e.g. MultiLineString, Polygon, MultiPolygon) are introduced to express BIM geometry data. For solving this issue, we propose an improved approach for effective describing geometric data in ifcOWL ontology through WKT. This representation can not only take full advantage of pre-existing WKT expressions to generate amore concise semantic representation, but also reduce the loss of semantic information. Our approach can be considered as an initial endeavour to explore the use of high order WKT to express building geometry data in OWL/RDF-environments, and possibly as one of feasible approaches for improving GIS and BIM interoperability.

In this paper, we mainly focus on the improving the representation of geometry data in ifcOWL ontology. We introduce a new WKT representation approach for BIM geometry data in ifcOWL ontology. In the section 2, we briefly review the related work about RDF representation approaches of BIM geometry data in ifcOWL ontology. After that, we analyze the possible semantic loss in WKT representation with pre-existing WKT expressions in section 3. For solving this problem, we also introduce our approach in section 3. Section 4 shows our analyses and discussions about our approach. Finally, we make a brief conclusion in section 5.

2 RELATED WORK

For different engineering applications and geometric representations, some novel geometry ontologies have been developed. For example, Building Topology Ontology (BOT) can capture the topological logical information of a building structure and elements (Rasmussen et al. 2017). The boundary representation OntoBREP ontology can use a mathematical model to describe geometric properties of objects, including topological entities (e.g. solids, shells, wires, edges) and geometric entities (e.g. points, curves, surfaces) (Perzylo et al. 2015). The GEOM ontology aims at capturing geometry from different sources with minimal loss of expressiveness (RDF.Ltd. 2012). In this paper, we only focus on discussing the related research about geometry representation in ifcOWL ontology.

Beetz et al. (Jakob Beetz 2007) pointed out that an RDF representation of geometric information that contained little semantic information was fairly inefficient and provided little additional value when it cannot be used in a logical inference/reasoning process. The logic inference and semantic search functionalities are important features provided by OWL and desirable for AEC industry. So, the design of RDF representation of geometric data is limited by certain notations (e.g. compatible with Description logics (DL)) or data types. For example, RDF terms rdf:list - rdf:first - rdf:rest (Brickley and Guha 2014) cannot be used in ifcOWL ontology to represent ordered aggregation data types of BIM because the generated ontology based on these RDF terms cannot be used for logical inference (Pauwels et al. 2017).

In all data types for representing geometric information, aggregation data types (e.g. ordered lists of point in Cartesian point, ordered lists of Cartesian points in polylines) are commonly adopted in IFC schema. How to effectively represent these geometric aggregate data types in OWL ontology has become one of the main research challenges. Translating the LIST data types in IFC schema into OWL expression has been discussed by Pauwels et al. (Pauwels et al. 2017) and de Farias et al. (de Farias, Roxin, and Nicolle 2015). In these translation approaches, ordered lists or sequences have been received major attentions because RDF data based on a triple structure (subject–predicate–object) represents ordered aggregated data types as fairly complex expressions (Pauwels and Terkaj 2016; Hoang and Törmä 2015). A typical example was illustrated in Figure 1, in which a Cartesian point in IFC schema was converted into ifcOWL. The conversion was recommended by buildingSMART and implemented by Pauwels et al (Pauwels et al. 2020). It is clear in Figure 1 that several triples are required to represent a cartesian points (an ordered list), including triples for expressing connection relationships of axes (list:hasNext) and necessary triples for expressing the semantic information of Cartesian point and coordinate values (list:hasContents, express:hasDouble). This converted approach caused the converted RDF file to be much larger than the original IFC file. (Hoang and Törmä 2015).

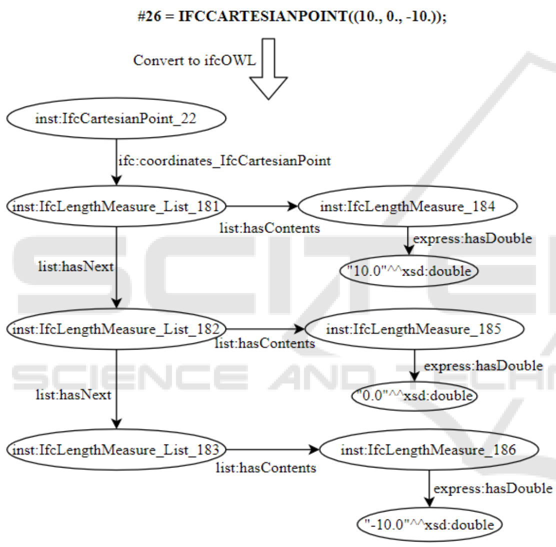


Figure 1: An example for converting an IfcCartesianPoint ((10.0, 0.0, -10.0)) into the ifcOWL representation, based on the converting approach in (Pauwels et al. 2020).

When the expression of Cartesian coordinate values was not changed in RDF, the changing the expression of connection relationships of axes may be an alternative approach for improving the representation of Cartesian point in ifcOWL. It generally resulted in customized new concepts in ifcOWL ontology (Pauwels et al. 2017). In this kind of approaches, new properties were created to point directly to each item in a list of two or three coordinates and then to distinguish between 2D and 3D IfcCartesianPoint concepts (Pauwels et al. 2017), shown in Figure 2, in which an instance of a Cartesian point had three connected properties to express 3D coordinates.

When the expression of Cartesian coordinate values was also changed and combined with the semantic information of coordinate axis of a Cartesian point, a more concise representation was proposed and illustrated in Figure 3, in which three triples were required for representing three coordinate values and 3D coordinate properties of a Cartesian point (Pauwels et al. 2017). The new data type properties were required to be defined in this representation and the cons and pros of this approach was discussed in ref. (Pauwels et al. 2017). Although this is a concise and simple representation, a Cartesian point in IFC schema was still converted into three triples, which inevitably leaded to the size increase of a converted file.

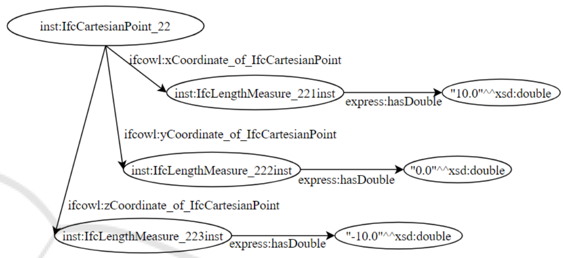


Figure 2: A representation of an IfcCartesianPoint ((10.0, 0.0, -10.0)) in ref. (Pauwels et al. 2017).

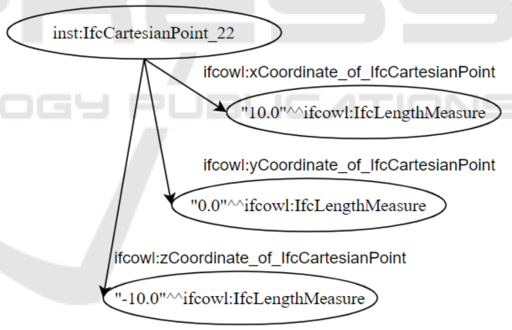


Figure 3: A representation of an IfcCartesianPoint ((10.0, 0.0, -10.0)), modified procedure 3b in (Pauwels et al. 2017).

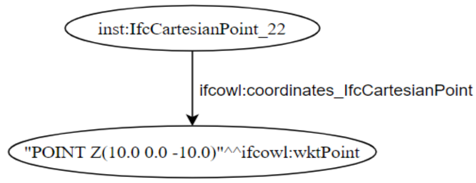


Figure 4: A representation of an ifc:CartesianPoint using the WKT approach.

Currently, it is accepted that using pre-existing WKT represents the BIM geometry data in ifcOWL and generates a concise RDF representation (Pauwels et al. 2017, McGlinn et al. 2019). For example, a POINT WKT string was adopted to represent a Cartesian point in IFC schema, shown in Figure 4. In this approach, only one triple in RDF can express an instance of IfcCartesianPoint in IFC schema. Similarly, the new formal definitions of data types were required in this approach. It was tested that the number of RDF triples can be decimated in several IFC4 cases after applying WKT to express triangulated geometries and Cartesian points in IFC4 (Pauwels et al. 2017). Additionally, WKT has been defined in Simple Feature Access (John R. Herring. 2011) and the re-use of existing vocabularies of WKT agreements was also recommended by W3C's Linked Data Best Practices (Bernadette et al. 2014). However, WKT expressions are limited to represent IFC schema, because WKT has been mainly applied in the geospatial domain and only focuses on geometrical information, non-geometric concepts in IFC schema cannot be represented by WKT. Furthermore, some special geometries in IFC schema that are not used in the geospatial domain cannot be directly represented by WKT.

Additionally, using some powerful WKT expressions to directly represent BIM geometric information will lose semantic information. For example, the geometric information of a wall containing an opening for a window can be expressed by WKT *PolyhedralSurface Z*, shown in Figure 5. This WKT expression (*PolyhedralSurface Z*) was largely compressed the number of RDF triples for representing the geometry data of the wall, whereas the loss of semantic information was not avoided in this representation (Pauwels et al. 2017). The other improvement approach may be to design new WKT expressions according to the IFC schema. However, the introducing new WKT expressions in ifcOWL are not recommended by Pauwels et al. (Pauwels et al. 2017), because the new WKT expressions require the additional and necessary effort to make data reasoners/query engines include and understand these expression strings. Currently, for achieving concise RDF representation and retaining semantic information, the geometric descriptions of products in IFC schema are commonly changed to apply to WKT expressions, such as WKT triangulated boundary representation, in which a solid surface is segmented into multiple triangular facets, and then WKT Point expression is used to record the vertices of triangular facets.

3 AN IMPROVED WKT REPRESENTATION APPROACH IN IFCOWL

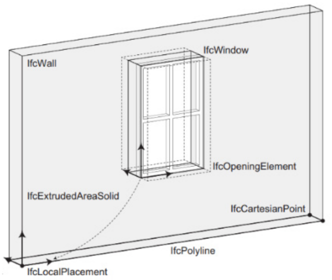
To overcome the limitation of pre-existing WKT expressions for representing IFC geometry data in ifcOWL ontology, we develop a new WKT representing approach that can further utilize pre-existing WKT geometry primitives and avoid the loss of sematic information. Here, we mainly explore two problems:

1. Which information will be lost when using WKT high order geometrical expressions in ifcOWL, such as *PolyhedralSurface Z*?

2. How to try to avoid the loss of semantic information when using WKT high order expressions?

3.1 Which Information Will Be Lost When using WKT Higher Order Geometrical Expressions in ifcOWL?

We take the case in Figure 5 as an example to discuss this problem. WKT expressions commonly use global spatial reference systems (the world coordinate system) (John R. Herring. 2011). However, a product defined in IFC schema generally has the relative coordinates/placement in relation to the placement of another product (BuildingSMART 2019b). So, the relative coordinates must be converted into the world coordinates in WKT expressions. In the Figure 5 (c), the first nine lines show the information about relative placement, and this kind of information can be used to convert the local placements/coordinates into world coordinates in WKT, including the following the segments #50-#52. When all products/elements are defined with world coordinates, the relative relationships of products in IFC can be implicitly contained in world coordinates of products. So, these information about *IfcLocalPlacement* and the related information cannot be considered to be lost.



(a) A wall with an opening element

```
GEOMETRYCOLLECTION Z(POLYHEDRALSURFACE Z(
((4    1   3, 4    1   0, 4    5.8 0, 4    5.8 3),
 (4    3.9 2, 4    2.9 2, 4    2.9 1, 4    3.9 1)),
((3.64 1   3, 3.64 1   0, 4    1   0, 4    1   3)),
((4    5.8 0, 4    1   0, 3.64 1   0, 3.64 5.8 0)),
((4    1   3, 4    5.8 3, 3.64 5.8 3, 3.64 1   3)),
((4    5.8 3, 4    5.8 0, 3.64 5.8 0, 3.64 5.8 3)),
((4    2.9 1, 4    3.9 1, 3.64 3.9 1, 3.64 2.9 1)),
((4    2.9 1, 3.64 2.9 1, 3.64 2.9 2, 4    2.9 2)),
((4    3.9 2, 4    2.9 2, 3.64 2.9 2, 3.64 3.9 2)),
((3.64 3.9 1, 4    3.9 1, 4    3.9 2, 3.64 3.9 2)),
((3.64 5.8 3, 3.64 5.8 0, 3.64 1   0, 3.64 1   3),
 (3.64 3.9 1, 3.64 2.9 1, 3.64 2.9 2, 3.64 3.9 2))
))
```

(b) The WKT description of the wall

```
#35=IFCLOCALPLACEMENT(#28,#34);
#34=IFCAXIS2PLACEMENT3D(#33,#7,#10);
#33=IFCCARTESIANPOINT((4.,1.,0.));
#7=IFCDIRECTION((0.,0.,1.));
#10=IFCDIRECTION((0.,1.,0.));

#28=IFCLOCALPLACEMENT($,#27);
#27=IFCAXIS2PLACEMENT3D(#8,#7,#6);
#8=IFCCARTESIANPOINT((0.,0.,0.));
#6=IFCDIRECTION((1.,0.,0.));

#48=IFCPRODUCTDEFINITIONSHAPE($,$,(#38,#47));

#38=IFCSHAPEREPRESENTATION(#11,'Axis','Curve2D
  ',(#37));
#36=IFCCARTESIANPOINT((4.8,0.,0.));
#37=IFCPOLYLINE((#8,#36));

#47=IFCSHAPEREPRESENTATION(#11,'Body','SweptSolid
  ',(#46));
#46=IFCEXTRUDEDAREASOLID(#45,#39,#7,3.);
#45=IFCARBITRARYCLOSEDPROFILEDEF(.AREA.,$,#44);
#39=IFCAXIS2PLACEMENT3D(#8,#7,#6);
#44=IFCPOLYLINE((#40,#41,#42,#43,#40));
#40=IFCCARTESIANPOINT((0.,0.));
#41=IFCCARTESIANPOINT((4.8,0.));
#42=IFCCARTESIANPOINT((4.8,0.36));
#43=IFCCARTESIANPOINT((0.,0.36));

#63=IFCOPENINGELEMENT('1wTXGF$$GHvQUJg67pSPh9',#5,$,
  $,$,#52,#62,$);
#52=IFCLOCALPLACEMENT(#35,#51);
#51=IFCAXIS2PLACEMENT3D(#50,#7,#6);
#50=IFCCARTESIANPOINT((1.9,0.,1.));

#62=IFCPRODUCTDEFINITIONSHAPE($,$,(#61));
#61=IFCSHAPEREPRESENTATION(#11,'Body','SweptSolid
  ',(#60));
#60=IFCEXTRUDEDAREASOLID(#59,#53,#7,1.);
#59=IFCARBITRARYCLOSEDPROFILEDEF(.AREA.,$,#58);
#53=IFCAXIS2PLACEMENT3D(#8,#7,#6);
#58=IFCPOLYLINE((#54,#55,#56,#57,#54));
#54=IFCCARTESIANPOINT((0.,-0.18,0.));
#55=IFCCARTESIANPOINT((1.,-0.18,0.));
#56=IFCCARTESIANPOINT((1.,0.54,0.));
#57=IFCCARTESIANPOINT((0.,0.54,0.));

#64=IFCRELVOIDSELEMENT('1wTXGG$$GHvQJZg67pSPh9',#5,$
  ,$,#49,#63);
#65=IFCWINDOW('1wTXGH$$GHvPr4g67pSPh9',#5,$,$,$,$,$,
  $,$,$);
#66=IFCRELFILLSELEMENT('1wTXGI$$GHvRTlg67pSPh9',#5,$
  ,$,#63,#65);
```

(c) The subset of the geometry information of the wall in IFC.

Figure 5: A case about using WKT to express a wall with an opening element provided in ref. (Pauwels et al. 2017).

The following segment #48=IfcProductDefinitionShape ($,$,(#38,#47)) defines two geometric shape representations of this product, which are a 2D representation (#38) and a "Body" 3D model (#47). The expression of Figure 5(b) only contains the 3D geometry information, excluding the 2D geometric information. Although the 2D geometric information can be deduced through 3D geometry information and a corresponding 3D engine, the part 2D semantic information are indeed lost, such as the semantic information of "Axis" and "Curve2D" and data value, which are retained in converting approaches recommended by buildingSMART (buildingSMART 2019a; Pauwels and Terkaj 2016). Additionally, these semantic descriptions are meaningless even if they are compulsively kept in the ifcOWL ontology, because these semantic descriptions have lost the direct relevant geometric data value.

The geometric information of "Body" solid model (#47) and the following geometric related descriptions have been expressed in Figure 5(b) with the world coordinates. After that, the opening element, described in *IfcOpeningElement* in #63, has the relative placement information and "Body" descriptions, which are also expressed in the WKT expression of the figure 5(b). However, the property *IfcIdentifier* of the *IfcOpeningElement* is lost in Figure 5(b), which can be used to link with the following lines: #64(*IfcRelVoidsElement*), #65(*IfcWindow*) and #66(*IfcRelFillsElement*). Moreover, because the geometric data of an opening element is merged into the whole geometric description of the wall, the other semantic information of an opening element can be hardly effectively correlated with the geometric data of an opening element in ifcOWL ontology, especially when multiple opening elements exist in a product.

Additionally, the current converting approaches in ref. (buildingSMART 2019a; Pauwels and Terkaj 2016) keep all semantic information about *IfcCartesianPoint* and every *IfcCartesianPoint* entity can be independently reused or linked with other ontologies. However, the semantic and geometric information about *IfcCartesianPoint* entities have been combined into an RDF literal in Figure 5(b), where every *IfcCartesianPoint* entity cannot be used independently without parsing the RDF literal. The combination of semantic information of *IfcCartesianPoint* entities can be viewed as the cost and key of compression of RDF, when using WKT expressions in ifcOWL.

3.2 How to Try to Avoid the Loss of Semantic Information in WKT High Order Expressions?

Based on the above analyses, several semantic information may be lost, mainly including 2D representation and the related semantic information of the opening element, while some information may be transferred from explicit expressions to implicit expressions, such as the relative relationships among products. The main reason of losing information is that the geometric representation is over-concentrated. To avoid losing the semantic information and introducing high order WKT expressions, we propose a new approach that can use pre-existing WKT expressions (including WKT high order geometric expressions) to be suitable for ifcOWL ontology. The main idea is to use multiple WKT expressions to describe geometric information of a product according to its descriptions in IFC.

The *IfcShapeRepresentation* in IFC schema can describe different geometric representations of a product or different product components (BuildingSMART 2019b). So, in our approach, basic WKT point and line expressions will continue to be used, such as expressing the position information of *IfcSite* or *IfcBuidling*. Additionally, an important criterion in our approach is to use one appropriate WKT expression for every *IfcShapeRepresentation* in IFC schema if the geometric information of the products can be expressed by WKT. Except the geometric information, the other information in IFC, including semantic descriptions, properties and relationships, can still be converted into RDF by approaches recommended by buildingSMART (buildingSMART 2019a; Pauwels and Terkaj 2016). In our approach, high order WKT expressions can be considered to be simplified to adapt to IFC schema, because expressing a complex structure of a product can require several *IfcShapeRepresentation* instances in IFC schema and every *IfcShapeRepresentation* instance has a WKT expression in our approach. That means multiple WKT expressions can split the expression of the complex structure of a product and the split structure can also simplify complex expression in WKT. The semantic relationships among the several WKT expressions of a product are the semantic relationships among corresponding *IfcShapeReresentation* instances in IFC schema.

We still use the case in Figure 5 to explain our proposal. The case in Figure 5 only used one WKT expression to represent geometric information of the wall and it will inevitably lead to the loss of semantic information. In our approach, the 2D geometric information of the wall in *#38=IfcShapeRepresentation(#11,'Axis','Curve2D',(#37))* and the related geometric information are retained and expressed as a *LINESTRING WKT*. The *#47=IfcShapeRepresentation(#11,'Body','SweptSolid ',(#46))* and the following lines until #43 segment are still expressed as *PolyhedralSurface Z*. Because there is an opening in the wall, an extra WKT expression *PolyhedralSurface Z* is needed to express the *#61=IfcShapeRepresentation(#11,'Body','SweptSolid ',(#60))* and following related geometric information. So, three WKT expressions are required to describe the geometric information of the wall in Figure 5 in our proposal. Our approach can avoid the loss of semantic information and generate a concise RDF representation. The results of our approach are show in Figure 6, in which some spaces are added for improving legibility in WKT and they can be removed to further condense. It is noted that we only change the expression of geometry information of a product in our approach. The other semantic information of a product and relationships of different components of a product are retained into RDF to avoid the loss of semantic information.

4 ANALYSES AND DISCUSSIONS

In our approach, every *IfcShapeRepresentation* instance is specified a corresponding WKT expression to describe related geometric information. Although the WKT expression form of IFC geometric information is different with IFC schema, the structure of the WTK geometry expression in ifcOWL ontology is similar with IFC schema in our approach. The geometric data of every *IfcShapeRepresentation* instance is converted into WKT, while the other semantic information still be retained in an *inst:IfcShapeRepresentation* instance in ifcOWL. In this way, properties of a product can be easy to be linked with the related shape representations. Furthermore, that all shape representations of a product are kept in ifcOWL ontology can lessen the loss of semantic information. Meanwhile, the relationships of multiple WKT expressions of a product can be clearly expressed in ifcOWL based on their relationships in IFC. Additionally, in engineering applications, the required WKT expressions can be independently extracted. For example, when comparing the opening element size with the window size, WKT expressions of the opening element and the window can be separately extracted and compared.

Generally, WKT can represent some complex geometric objects, such as using *PolyhedralSurface Z*

```
#38=IFCSHAPEREPRESENTATION(#11,'Axis','Curve2D
  ',(#37));
#36=IFCCARTESIANPOINT((4.8,0.,0.));
#37=IFCPOLYLINE((#8,#36));
```

```
LINESTRING (4 1 0, 4 5.8
0)^^ifcOWL:WKTLineString
```

(a) WKT LineString expression for 2D

```
#47=IFCSHAPEREPRESENTATION(#11,'Body','SweptSolid
  ',(#46));
#46=IFCEXTRUDEDAREASOLID(#45,#39,#7,3.);
#45=IFCARBITRARYCLOSEDPROFILEDEF(.AREA.,$,#44);
#39=IFCAXIS2PLACEMENT3D(#8,#7,#6);
#44=IFCPOLYLINE((#40,#41,#42,#43,#40));
#40=IFCCARTESIANPOINT((0.,0.));
#41=IFCCARTESIANPOINT((4.8,0.));
#42=IFCCARTESIANPOINT((4.8,0.36));
#43=IFCCARTESIANPOINT((0.,0.36));
```

```
GeometryCollection Z(PolyhedralSurface Z(
((4    1   3, 4    1   0, 4    5.8 0, 4    5.8 3)),
((4    1   3, 3.64 1   3, 3.64 1   0, 4    1   0)),
((4    1   0, 3.64 1   0, 3.64 5.8 0, 4    5.8 0)),
((4    1   3, 4    5.8 3, 3.64 5.8 3, 3.64 1   3)),
((4    5.8 3, 4    5.8 0, 3.64 5.8 0, 3.64 5.8 3)),
((3.64 5.8 3, 3.64 5.8 0, 3.64 1   0, 3.64 1   3))))
```

(b) WKT PolyhedralSurface Z expression for "SweptSolid"

```
#63=IFCOPENINGELEMENT('1wTXGF$$GHvQUJg67pSPh9',#5,$,
  $,$,#52,#62,$);
#52=IFCLOCALPLACEMENT(#35,#51);
#51=IFCAXIS2PLACEMENT3D(#50,#7,#6);
#50=IFCCARTESIANPOINT((1.9,0.,1.));

#62=IFCPRODUCTDEFINITIONSHAPE($,$,(#61));
#61=IFCSHAPEREPRESENTATION(#11,'Body','SweptSolid
  ',(#60));
#60=IFCEXTRUDEDAREASOLID(#59,#53,#7,1.);
#59=IFCARBITRARYCLOSEDPROFILEDEF(.AREA.,$,#58);
#53=IFCAXIS2PLACEMENT3D(#8,#7,#6);
#58=IFCPOLYLINE((#54,#55,#56,#57,#54));
#54=IFCCARTESIANPOINT((0.,-0.18,0.));
#55=IFCCARTESIANPOINT((1.,-0.18,0.));
#56=IFCCARTESIANPOINT((1.,0.54,0.));
#57=IFCCARTESIANPOINT((0.,0.54,0.));
```

```
GeometryCollection Z(PolyhedralSurface Z(
((4    3.9 2, 4    2.9 2, 4    2.9 1, 4    3.9 1)),
((4    2.9 1, 4    3.9 1, 3.64 3.9 1, 3.64 2.9 1)),
((4    2.9 1, 3.64 2.9 1, 3.64 2.9 2, 4    2.9 2)),
((4    3.9 2, 4    2.9 2, 3.64 2.9 2, 3.64 3.9 2)),
((3.64 3.9 1, 4    3.9 1, 4    3.9 2, 3.64 3.9 2)),
((3.64 3.9 1, 3.64 2.9 1, 3.64 2.9 2, 3.64 3.9 2))))
```

(c) WKT PolyhedralSurface Z expression for "SweptSolid"

Figure 6: The new WKT representation approach for a wall with an opening element.

to express 3D polyhedral surface with holes (openings) and using *Polygon* to express 2D polygon with interior linear rings etc. However, IFC schema adopts multiple *IfcShapeRepresentation* entities to respectively describe polyhedral surface and opening elements of a product. So, WKT *PolyhedralSurface Z* or *Polygon* used in our approach (without holes or interior linear rings, because holes and interior linear rings are expressed in other WKT expressions) will be simpler than used in geospatial domain. Furthermore, our approach doesn't require to change the definitions of pre-exist WKT expressions in RDF and may use these high order WKT expressions in ifcOWL ontology.

In terms of the number of triples in RDF, our approach has more concise RDF representation than ifcOWL ontology (buildingSMART 2019a) and conservative WKT serialization proposed by Pauwels et al. (Pauwels et al. 2017), because some high order WKT expressions can be used in our approach, which can combine multiple Cartesian points into one expression. The results of triple count for expressing the wall contained a window in three approach are shown in Figure 7.

However, curves cannot be serialized in WKT that can only describe points, linear-segments, polygon and the high order combinations of them in 2D or 3D (John R. Herring. 2011). The construction-related geometric information contained curves cannot be expressed by WKT, which limits applications of WKT in ifcOWL. Meanwhile, it is also a limitation of our approach. Additionally, when using WKT to expression geometric information in our approach, some extra calculations are necessary for the converting from IFC schema into RDF, such as the converting calculation from relative coordinates to world coordinates and some geometry shape expressing calculations from IFC schema to high order WKT expressions, etc. Additionally, the converting IFC geometry data into WKT expressions may be a complex task, because a whole building can have hundreds/thousands of *IfcShapeRepresentation* entities, and the converting task is hardly implemented manually. The curve expression in WKT and the automatic converting program (from IFC schema to WKT expressions) need to be further researched. If these issues will be overcome, geometric data in building/construction and geospatial domain can be expressed in WKT, and WKT may be a feasible approach for improving GIS and BIM interoperability.

5 CONCLUSIONS

In this paper, a new expression approach is proposed for using WKT expressions in ifcOWL. It cannot only provide concise RDF representation with high order WKT expressions, but also avoid the loss of semantic information in ifcOWL. In our approach, an appropriate WKT expression is used for every *IfcShapeRepresentation* instance in IFC schema if the geometric information of a product can be expressed by WKT. It means that every *IfcShapeRepresentation* instance has its independent geometric expression and the related property semantic information of every *IfcShapeRepresentation* instance can be linked with its geometric expression. Based on geometry descriptions in IFC schema, multiple WKT expressions may be used to express the geometric information of a product. So, WKT high order expressions can be used to generate the more concise RDF representation in our approach. Because the WKT high order expressions are used, the limitations of WKT also become the limitations of our approach.

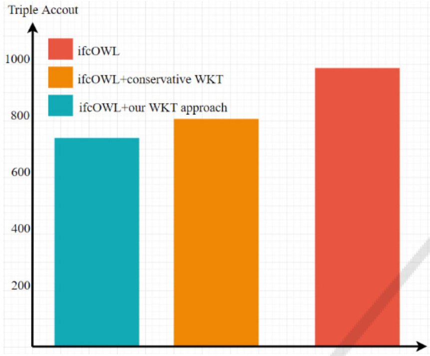


Figure 7: Triple counts for the ifcOWL, conservative WKT representations and our approach.

REFERENCES

Bernadette Hyland, Ghislain Atemezing, and Boris Villazón-Terrazas. 2014. 'Best Practices for Publishing Linked Data - W3C Working Group Note 09 January 2014', Accessed on 04 Jan. 2020. https://www.w3.org/TR/ld-bp/.

Brickley, Dan, and R.V. Guha. 2014. 'RDF Schema 1.1 - W3C Recommendation 25 February 2014', Accessed on 27 Dec. 2020. https://www.w3.org/TR/rdf-schema/.

buildingSMART. 2019a. 'ifcOWL', Accessed on 17 December 2020. https://technical.buildingsmart.org/standards/ifc/ifc-formats/ifcowl/.

buildingSMART. 2019b. 'Industry Foundation Classes Release 4 (IFC4)', Accessed 21 July, 2020. https://standards.buildingsmart.org/IFC/RELEASE/IFC4/FINAL/HTML/.

de Farias, Tarcisio Mendes, Ana Roxin, and Christophe Nicolle. 2015. 'IfcWoD, semantically adapting IFC model relations into OWL properties', arXiv preprint arXiv:1511.03897.

Hoang, Nam Vu, and Seppo Törmä. 2015. "Implementation and Experiments with an IFC-to-Linked Data Converter." *In 32nd CIB W78 Conference*, 285-94. Eindhoven, Netherlands.

ISO. 2013. 'ISO 16739:2013 Industry Foundation Class (IFC) for Data Sharing in the Construction and Facility Management Industries', Accessed on 12 December. https://www.iso.org/standard/51622.html.

Jakob Beetz, Bauke de Vries, Jos van Leeuwen. 2007. "RDF-based distributed functional part specifications for the facilitation of service-based architectures." *In 24th CIB W78 Conference,* 183-88. Maribor, Slovenia.

John R. Herring., ED. 2011. "*OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture*" In, edited by John R. Herring, 51-62. Open Geospatial Consortium, Inc.

Laakso, Mikael, and AO Kiviniemi. 2012. 'The IFC standard: A review of history, development, and standardization, information technology', *ITcon*, 17: 134-161.

McGlinn, K., A. Wagner, P. Pauwels, P. Bonsma, P. Kelly, and D. O'Sullivan. 2019. 'Interlinking geospatial and building geometry with existing and developing standards on the web', *Automation in Construction*, 103: 235-50.

Pieter Pauwels, and Walter Terkaj. 2016. 'EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology', *Automation in Construction*, 63: 100-133.

Pieter Pauwels, Thomas Krijnen, Walter Terkaj and Jakob Beetz. 2017. 'Enhancing the ifcOWL ontology with an alternative representation for geometric data', *Automation in Construction*, 80: 77-94.

Pieter Pauwels, Lewis John McGibbney, Benjamin Thurm, and Giovanni Velludo. 2020. 'IFC to RDF Tool', Accessed on 08 September 2020. https://github.com/pipauwel/IFCtoRDF.

Perzylo, A., N. Somani, M. Rickert, and A. Knoll. 2015. "An ontology for CAD data and geometric constraints as a link between product models and semantic robot task descriptions." *In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4197-203.

Rasmussen, Mads Holten, Pieter Pauwels, Jan Karlshøj, and Christian Hviid. 2017. "Proposing a Central AEC Ontology That Allows for Domain Specific Extensions." *In Proceedings of the Joint Conference on Computing in Construction*, pp. 237–244.

RDF.Ltd. 2012. 'geometry.ttl', Accessed on 27. Dec. 2020 http://rdf.bg//geometry.ttl.

Wagner, Anna, Mathias Bonduel, Pieter Pauwels, and Uwe Rüppel. 2020. 'Representing construction-related geometry in a semantic web context: A review of approaches', *Automation in Construction*, 115: 103-130.

Zhao, Xianbo. 2017. 'A scientometric review of global BIM research: Analysis and visualization', Automation in Construction, 80: 37-47.

Zhong, B. T., H. T. Wu, H. Li, S. Sepasgozar, H. B. Luo, and L. He. 2019. 'A scientometric analysis and critical review of construction related ontology research', *Automation in Construction*, 101: 17-31.