

## AUTOMATIC ONTOLOGY GENERATION OF BIM AND GIS DATA

A. U. Usmani, M. Jadidi,\* G. Sohn

ESSE, Lassonde School of Engineering, York University, M3J 1P3 Canada - (usmani, mjadidi, gsohn)@yorku.ca

Commission IV, WG IV/2

**KEY WORDS:** Ontology, BIM, GIS, Semantic Web, OWL, FCA, Transformation Patterns, XSD

### ABSTRACT:

Data represented in the form of geospatial context and detailed building information are prominently nurturing infrastructure development and smart city applications. Bringing open-formats from data acquisition level to information engineering accelerates geospatial technologies towards urban sustainability and knowledge-based systems. BIM and GIS technologies are known to excel in this domain. However, fundamental level differences lie among their data-formats, which developed integration methods to bridge the gap between these distinct domains. Several studies have conducted data, process, and application-level integration, considering the significance of collaboration among these information systems. Although integration methods have narrowed the gap of geometric dissimilarity, semantic inconsistency, and information loss yet add constraints towards achieving interoperability. Integration using semantic web technology is more flexible and enables process-level integration without changing data format and structure. However, due to its developing nature and complex BIM-GIS data-formats, most approaches adapted requires human intervention. This paper presents a method, named OGGD (Ontology Generation for Geospatial Data), that implements a formal method for automatic ontology generation from XSD documents using transformation patterns following three extensive processes; first, formalization of XSD elements and transformation patterns; the second process identifies corresponding patterns explicitly, and the last process generates ontology for XSD schema. XSD elements from open-standard data models of BIM and GIS, ifcXML and CityGML, are manipulated and transformed into a semantically rich OWL model. The ontology models created can be applicable for information-based integration systems that will nurture knowledge-discovery and urban applications.

### 1. INTRODUCTION

The future of urban development relies on information collected from Architecture, Engineering, Construction and Facility Management (AEC/FM), and geospatial domains (Song et al., 2017). Building Information Modelling (BIM) and Geographic Information System (GIS) are conventional representations of these systems. Both fields have been developed for addressing respective problem domains. However, with technological advancements and user specifications, they have advanced, individually, with overlapping features. Data in each domain represent vital information, and thus bridging the gap between their heterogeneous data-sets without information loss becomes critical (Wang et al., 2019). Traditional methods for integrating BIM and GIS have surfaced issues of data incompatibility, misinterpretation, and absence of information (Liu et al., 2017). Integration using semantic web technology has shown promising results due to its natural ability towards heterogeneous data integration (Zhu et al., 2018). Though, interoperability between BIM and GIS using existing semantic integration is limited to the manual or semi-automatic process of generating ontology, one of the main building blocks of the semantic web.

Data integration systems require global schema generated from heterogeneous data sources and correspondence between schema elements based on names, data types, constraints, and semantic properties. Rich ontologies expressed by the semantics of data exploit such techniques. Thus, semantic web technique qualifies for achieving semantic interoperability by interlinking heterogeneous data sources, like BIM and GIS, and representing semantic information in a common RDF (Resource Description Framework) format (Hor et al., 2018).

Semantic web technologies based integration methods has enabled bidirectional information exchange for BIM and GIS (Karan et al., 2016). However, these approaches are ongoing processes, mainly manual and time-costly, to achieve interoperability between BIM and GIS (Liu et al., 2017). Especially enriching ontology definitions using data-formats for particular domain is a slow process at early stage. In contrast, multiple approaches (Hacherouf et al., 2015) are proposed for transformation of XML data-format to OWL (Web Ontology Language) schemas to find XSD (XML Schema Definition) constructs (i.e. *xs:element*, *xs:complexType*, *xs:simpleType*, etc.) matched with their valid OWL representations (*class*, *ObjectProperty*, *DataProperty*, etc.).

The proposed study presents a method named OGGD (Ontology Generation for Geospatial Data) that implements a formal method for automatic ontology generation from XSD documents using transformation patterns following three extensive processes. First, formalization of XSD elements and transformation patterns; the second process identifies corresponding patterns explicitly, and the last process generates ontology for XSD schema. XSD elements are extracted from the XML-based open-standard data model of BIM and GIS, ifcXML and CityGML respectively, and transformed into OWL model (Hitzler et al., 2012).

### 2. METHODOLOGY

Our proposed methodology utilizes ontology development for XML based geospatial data by extending Janus (Bedini et al., 2011) and PIXCO (Pattern Identification for XSD Conversion to OWL) (Hacherouf et al., 2019) and proposing formal automatic ontology generation of BIM and GIS data. Janus

\* Corresponding author

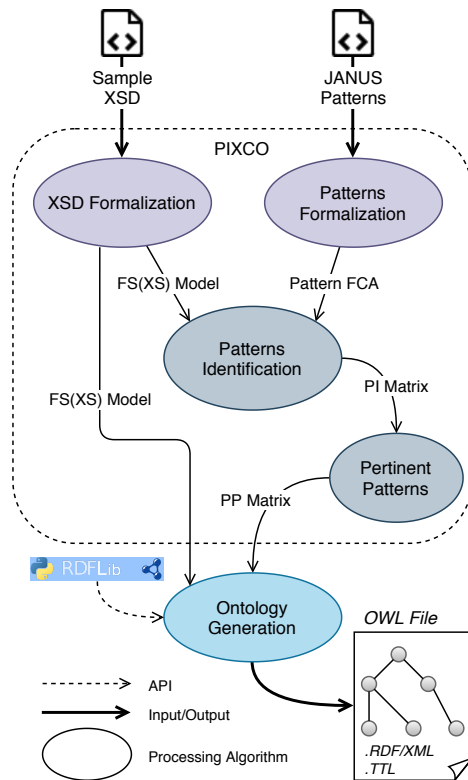


Figure 1. OGGD Architecture

defines sets of transformation patterns for correspondence rules between XML schema and OWL schema constructs with the OWL2-RL profile (Motik et al., 2012). Using these sets of patterns, PIXCO provides an extension to generate well-structured ontologies. However, in conjunction, these approaches provide incomplete population process and overlook XSD elements, comprising *list*, *length* etc., an XSD schema. The idea for extending these approaches in OGGD is, more the presented transformation process considers XSD components, the more ontology generated is semantically rich and exhaustive.

The proposed framework is illustrated in Fig. 1 for the maximum transformation of XSD constructions to OWL. The proposed methodology implements multiple algorithms and is based on three main processes. First process utilizes XSD schema from XML documents (ifcXML and CityGML) to develop a mathematical model, *FS(XS)* (Formal Structure of XML Schema) (Hacherouf et al., 2019), for formal modelling of XSD constructs. Further, we also perform the formalization of transformation patterns (Bedini et al., 2011, Hacherouf et al., 2019) using the context of FCA (Formal Concept Analysis) (Ganter, Wille, 1999). The second process identifies patterns for each XSD construction and further, it amends pertinent patterns for the respective constructs. Results from the second step along with corresponding rule from transformation patterns are utilized in the next step. In the third process, OWL models are generated for each XSD construct associated with appropriately identified transformation patterns, and a composite OWL file is created for OWL models.

OGGD is a python-based generalized solution for transforming XSD schemas to OWL models using RDF Library (Krech, 2020), and consumes BIM and GIS open-formats for its potential validation. In this study we conduct an experiment on an example ifcXML schema (Fig. 2) for prototyping. Ontology

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <xs:schema
3     xmlns:xs="http://www.w3.org/2001/XMLSchema"
4     xmlns:ifc="http://www.buildingsmart-tech.org/ifcXML/IFC4/final"
5     <xs:element name="Entity" type="ifc:Entity"/>
6     <xs:complexType name="Entity">
7         <xs:attribute name="href" type="xs:anyURI"/>
8         <xs:attribute name="ref" type="xs:IDREF"/>
9     </xs:complexType>
10
11 </xs:schema>

```

Figure 2. An example of ifcXML XSD structure used in the experiment. Here line 6-9 represents XSD Constructs  $C_1$ - $C_4$  respectively.

model generated for Fig. 2 using OGGD architecture is validated by using Protégé (Musen, 2015) tool and shown in Fig. 5.

### 3. EXPERIMENT AND RESULTS

OGGD accepts XSD schema and performs series of operation including; extraction, formalization, identification and generating OWL model. We produced an example ifcXML schema (Fig. 2) to evaluate prototype operations, starting extracted XSD elements' formalization in *FS(XS)* model. Janus patterns modelled in FCA context are exploited by second process (i.e. *patternsIdentification* algorithm) for the *FS(XS)* model. Where each XSD construction block  $\{C_i \dots C_j\}$  is associated with set of potential transformation patterns  $\{P\}$  based on schema *elements*, resulting *PI* matrix (Table 1 for Fig. 2). Patterns identified in *PI* matrix can be validated from Table II-VII in (Bedini et al., 2011). Furthermore, for sets of XSD construction block in *FS(XS)* model, pertinent patterns are selected based on *attributes* and *attribute value types* of each construction block resulting *PP* matrix (Table 2). These relevant patterns are used to generate respective OWL models. Extracted elements of every XSD block (e.g.  $\{C_1\}$ ) are appropriately modelled to corresponding OWL pattern in list of transformation patterns (i.e.  $\{25\}$ ). Resulting OWL model generated of Fig. 2 that is exported in RDF/XML (Fig. 3) and RDF/Turtle (Fig. 4) formats, and can be visualized in Fig. 5.

Block	Pattern (P)
$\{C_1\}$	$\{15, 16, 21, 24, 25\}$
$\{C_2, C_3, C_4\}$	$\{27, 28\}$

Table 1. Patterns Identification algorithm resulting *PI* matrix

Block	Pattern (P)
$\{C_1\}$	$\{25\}$
$\{C_2, C_3, C_4\}$	$\{27\}$

Table 2. Pertinent Patterns algorithm resulting *PP* matrix

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <rdf:RDF
3     xmlns:owl="http://www.w3.org/2002/07/owl#"
4     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
5     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
6 >
7     <owl:DatatypeProperty rdf:about="http://example.org/autology#ref">
8         <rdfs:domain rdf:resource="https://www.buildingsmart-tech.org/ifcXML/IFC4/final#Entity"/>
9         <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#IDREF"/>
10    </owl:DatatypeProperty>
11    <owl:DatatypeProperty rdf:about="http://example.org/autology#href">
12        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#anyURI"/>
13        <rdfs:domain rdf:resource="https://www.buildingsmart-tech.org/ifcXML/IFC4/final#Entity"/>
14    </owl:DatatypeProperty>
15    <owl:Class rdf:about="https://www.buildingsmart-tech.org/ifcXML/IFC4/final#Entity"/>
16 </rdf:RDF>

```

Figure 3. RDF/XML generated for Fig. 2 example

```

1 @prefix ag: <http://example.org/autology#> .
2 @prefix dc: <http://purl.org/dc/elements/1.1/> .
3 @prefix owl: <http://www.w3.org/2002/07/owl#> .
4 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
5 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
6 @prefix xml: <http://www.w3.org/XML/1998/namespace> .
7 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
8 @prefix ifc: <http://www.buildingsmart-tech.org/
9       ifcXML/IFC4/final#> .
10
11 ag:href a owl:DatatypeProperty ;
12     rdfs:domain ifc:Entity ;
13     rdfs:range xsd:anyURI .
14
15 ag:ref a owl:DatatypeProperty ;
16     rdfs:domain ifc:Entity ;
17     rdfs:range xsd:IDREF .
18
19 ifc:Entity a owl:Class .

```

Figure 4. RDF/Turtle (.ttl) generated for Fig. 2 example

A single XSD construct can be identified with multiple transformation patterns (e.g.  $C_1$  in Table 1). In order to identify pertinent pattern for certain XSD construct, XSD attributes and attribute value's types orients the correct pattern.

The prototype method is also experimented for the schema specifications of IFC formats<sup>1</sup> and CityGML base schema<sup>2</sup> format. Initial results highlights that each IFC schema format have different number of XSD constructs (Fig. 6) and the time required to execute algorithm of identifying patterns (listed in Table 3) is not proportional to number of XSD constructs. Elapsed time for ontology generation of IFC schema formats, however, remains proximate. For the CityGML base schema, OGGD identifies 145 XSD constructs with minimal period of time to identify patterns and generate introductory ontology.

Preliminary results show ontologies generated from OGGD prototype exploits and evaluate Janus and PIXCO frameworks, highlighting axioms including *class*, *ObjectProperty* and *DatatypeProperty* to manifest OGGD implementation. However, significant effort is required to define an automated framework that articulates XSD schema elements to OWL, even if the transformation patterns are defined.

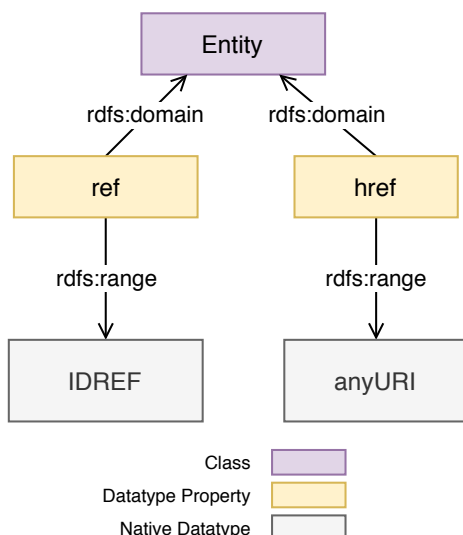


Figure 5. Model of ontology generated from example in Fig. 2

<sup>1</sup> <https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/>

<sup>2</sup> <http://schemas.opengis.net/citygml/>

## 4. DISCUSSION

Indeed, generating OWL/RDF graph is a tortuous approach with critical edge cases for its syntax and reasoning. OWL API itself has support for few platforms as semantic web technologies are still developing and maturing; thus, it depends on related libraries like RDFLib. Python-based RDFLib also has a limitation of supporting OWL profiles in RDF format. For example, a list of data values as collection needs to be specified as basic list constructs (Klyne, Carroll, 2004). Hence the process of defining concepts (subjects) and properties (objects) linked with relationships (predicates) for corresponding pattern transformation from XSD structure to OWL becomes challenging. Nevertheless, the library provides you a root-level access to generate a very refined ontology with a detailed OWL profile that unfolds potential rich semantics of subjects, predicates, and objects.

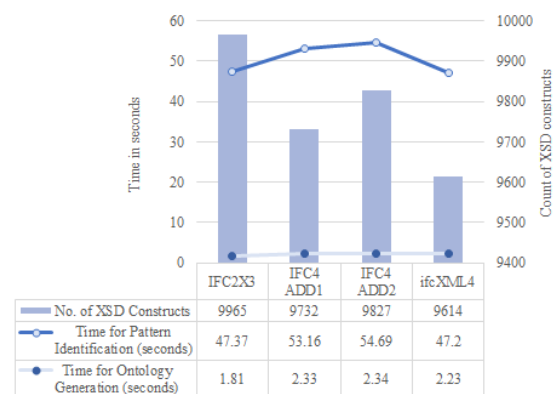


Figure 6. Results comparing execution time for pattern identification and ontology generation of IFC schemas for listed transformation patterns in Table 3

There are limited formal methods defined for ontology generation, and prototype work shows potential in the automation of OWL modeling. The approach of the semantic web is promising but time-consuming and still developing. Janus and PIXCO methods have relatively processed XSD schemas from the B2B (Business to Business) and UBL (Universal Business Language) domain for validation. OGGD progressively implements these approaches with focus on extending XSD elements that are not processed (like *list*, *maxLength* and *minLength* etc.) though are components of geospatial schemas. Additionally, there are limitations with XSD notations and no direct mapping for XML constraints into OWL. Thus, OGGD proposes to consider maximum XSD components by participating further in supplementary transformation patterns. Currently, the algorithm complexity of the OGGD implementation is polynomial, and it depends on the number of XSD constructs to be processed by each algorithm for OWL transformation. If  $n$  is the number of XSD constructs and  $k$  is a number of the nested algorithm, then the complexity of algorithms can be denoted as:

$$O(n^k) \quad (1)$$

where  $1 \leq k \leq 5$

The complexity for automation of the ontology generation can be accessed as at early stage ontology development itself requires human intervention, and the process is time-costly. It involves the development, transformation, and validation of OWL

Transformation Pattern	XSD Element
Attribute type	attribute
Derived type	complexContent restriction
Simple and Complex type	simpleType complexType
XS elements	element sequence
Grouped entities	minOccur maxOccur

Table 3. Preparatory patterns implemented by OGGD for transforming XSD elements into to OWL

schema sub-structures and further integrated with a complete model. Implementation of the proposed methodology is an ongoing process and proposed is an innovative study for implementing transformation patterns with outlined XSD elements for a prototype in Table 3. The latter feature provides an opportunity to deliver adaptive results. OGGD is designed as generalized architecture to support XSD documents with a valid schema. The architecture is also scalable for more transformation patterns and implementing or modifying corresponding rules between XSD constructs and OWL representations to generate semantically exhaustive ontologies. However, newly developed ontology models require a certain level of revision to be used further for integration knowledge generation.

## 5. CONCLUSION

In this study, we presented the OGGD method that provides potentially generalized solutions for the automatic generation of OWL from XSD schema using transformation patterns for deriving maximum XSD attributes of BIM and GIS data. We extended the three-step method for ontology development using geospatial data; XSD formalization, pattern identification, and ontology generation. The contemporary OGGD implementation results unfold rich semantics information that can be extracted by extending supplementary transformation patterns, which promises semantically rich geospatial ontologies. Future work for the study focuses on the implementation of extended transformation patterns and perform validation with completeness, consistency, and correctness of generated ontology models. Furthermore, OWL models will be investigated, and performance tests conducted to devise the complexity of algorithms and apply recommended optimizations. Generated RDF from ontology models can further nurture knowledge discovery for smart city applications like facility management and urban environment analysis, by retrieving information from cross-domain integrated RDF graphs.

## REFERENCES

Bedini, I., Matheus, C., Patel-Schneider, P. F., Boran, A., Nguyen, B., 2011. Transforming xml schema to owl using patterns. *2011 IEEE Fifth International Conference on Semantic Computing*, 102–109.

Ganter, B., Wille, R., 1999. *Formal Concept Analysis*. Springer Nature.

Hacherouf, M., Bahloul, S. N., Cruz, C., 2015. Transforming XML documents to OWL ontologies: A survey. *Journal of Information Science*, 41(2), 242–259. <https://doi.org/10.1177/0165551514565972>.

Hacherouf, M., Nait-Bahloul, S., Cruz, C., 2019. Transforming XML schemas into OWL ontologies using formal concept analysis. *Software & Systems Modeling*, 18(3), 2093–2110. <https://doi.org/10.1007/s10270-017-0651-4>.

Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P. F., Rudolph, S., 2012. OWL 2 Web Ontology Language Primer. W3C Recommendation <https://www.w3.org/TR/owl2-primer/> (4 May 2020).

Hor, A.-H., Sohn, G., Claudio, P., Jadidi, M., Afnan, A., 2018. A Semantic graph database for BIM-GIS integrated information model for an intelligent urban mobility web application. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-4, 89–96. [doi.org/10.5194/isprs-annals-IV-4-89-2018](https://doi.org/10.5194/isprs-annals-IV-4-89-2018).

Karan, E. P., Irizarry, J., Haymaker, J., 2016. BIM and GIS Integration and Interoperability Based on Semantic Web Technology. *Journal of Computing in Civil Engineering*, 30(3), 04015043.

Klyne, G., Carroll, J. J., 2004. Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation [www.w3.org/TR/2004/REC-rdf-concepts-20040210](http://www.w3.org/TR/2004/REC-rdf-concepts-20040210) (4 May 2020).

Krech, D., 2020. RDFLib Python Library, Version 4.2.2. GitHub repository. [github.com/RDFLib/rdfli](https://github.com/RDFLib/rdfli) (05 February 2020).

Liu, X., Wang, X., Wright, G., Cheng, J. C. P., Li, X., Liu, R., 2017. A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and Geographic Information System (GIS). *ISPRS International Journal of Geo-Information*, 6(2). <https://www.mdpi.com/2220-9964/6/2/53>.

Motik, B., Grau, B. C., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C., 2012. OWL 2 Web Ontology Language Profiles. W3C Recommendation <https://www.w3.org/TR/owl2-profiles/> (4 May 2020).

Musen, M. A., 2015. The protégé project: a look back and a look forward. *AI Matters*, 1(4), 4–12. <https://doi.org/10.1145/2757001.2757003>.

Song, Y., Wang, X., Tan, Y., Wu, P., Sutrisna, M., Cheng, J., Hampson, K., 2017. Trends and Opportunities of BIM-GIS Integration in the Architecture, Engineering and Construction Industry: A Review from a Spatio-Temporal Statistical Perspective. *International Journal of Geo-Information*, 6, 397. [doi.org/10.3390/ijgi6120397](https://doi.org/10.3390/ijgi6120397).

Wang, H., Pan, Y., Luo, X., 2019. Integration of BIM and GIS in sustainable built environment: A review and bibliometric analysis. *Automation in Construction*, 41 - 52.

Zhu, J., Wright, G., Wang, J., Wang, X., 2018. A Critical Review of the Integration of Geographic Information System and Building Information Modelling at the Data Level. *ISPRS International Journal of Geo-Information*, 7(2). <https://www.mdpi.com/2220-9964/7/2/66>.