

IMPROVEMENTS IN AUTOMATED DERIVATION OF OWL ONTOLOGIES FROM GEOSPATIAL UML MODELS

Knut Jetlund¹

¹Norwegian University of Science and Technology, Gjøvik, Norway, and The Norwegian Public Roads Administration -
knut.jetlund@vegvesen.no

Commission IV, WG IV/OGC

KEY WORDS: Ontologies, UML, Geospatial, MDA, Standards, OGC, ISO/TC 211

ABSTRACT:

Standards from ISO/TC 211 are the foundation for modelling a universe of discourse in a geospatial context. UML models based on the standards, and in particular based on the UML profile defined in ISO 19103, have been developed and implemented in applications and databases for a wide range of geospatial information, from international to national and agency level. Amounts of information has been collected, maintained and made available based on the models, but mainly through specific services and exchange formats for geospatial information. To make the models and the information available in The Semantic Web, the geospatial UML models need to be transformed from UML to OWL ontologies, and the information needs to be transformed from UML-based structures to RDF triples. This paper investigates methods for transforming UML models of geospatial information to OWL ontologies, identifies challenges, suggest improvements and identifies needs for further research. Several methods for automated transformation from geospatial UML models to OWL handle basic concepts, but some concepts and context-closed restrictions from UML cannot be directly transformed to the open world of The Semantic Web. None of the analysed methods handles all of these issues, and suggested improvements include combining and improving transformation rules, as well as modifications in the UML models. To what degree and how these issues need to be handled will depend on whether the scope of the ontologies is to simply present geospatial information on The Semantic Web, or if they shall be used in a bidirectional information exchange.

1. INTRODUCTION

Most of the information on the World Wide Web is available as documents and images in formats like HTML, PDF or JPEG, or in databases that are accessed by special applications. Humans can combine the information, make assumptions and extend knowledge by reading and understanding documents and looking at tables and maps, even if the documents and databases are structured in different ways, and even if different terms are used for the same phenomena. For processing by machines however, the information must have a formal structure and explicit semantic. The Semantic Web provides the framework for describing information in structures that machines can use to understand and share information, and reuse it independently of applications.

The basic framework for information modelling on The Semantic Web is the Resource Description Framework – RDF, in which the information is described with triples and graphs. A triple consists of an object, a predicate and a subject, where objects and subjects are resources that can be anything from a concrete physical phenomena to an abstract concept, and the predicate describes the connection between the object and the subject. An object in one triple may be a subject for another triple, and a set of triples form a graph of information. The Web Ontology Language – OWL is the main framework for describing ontologies, built on top of RDF and using the same principles with triples and graphs.

The geospatial aspect of information on the Web is important for many use cases, e.g. navigation, travel, advertising etc. The ISO Technical Committee 211 – ISO/TC 211 and the Open Geospatial Consortium – OGC have both been working on standardization of geospatial information since 1994,

individually and in cooperation. The work is based on the ISO/TC 211 approach to modelling information, described in the standards ISO 19103 (ISO/TC 211, 2015a) and ISO 19109 (ISO/TC 211, 2015b). The ISO/TC 211 approach is to perceive a part of the real world, known as a universe of discourse, limit the perception to a closed context of geographic¹ information, and classify feature types (classes) and properties (attributes) according to this perception. Figure 1, from ISO 19109, illustrates the ISO/TC 211 approach.

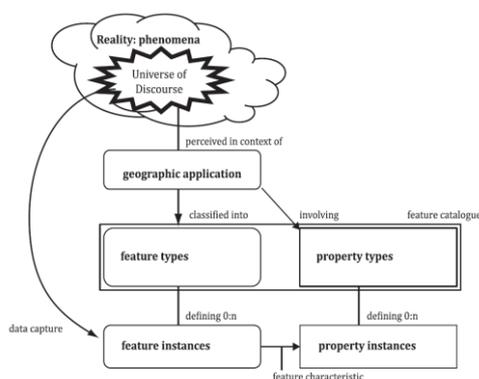


Figure 1 The process for modelling geographic information, from ISO 19109.

The foundation for information modelling in the ISO/TC 211 approach is the Unified Modelling Language – UML,

¹ ISO/TC 211 use the term “geographic information” (ISO/TC 211, 2014), while OGC use the term “geospatial information” (OGC, 2018). In this paper, the terms “geographic” and “geospatial” are considered equivalent.

formalized through the UML profile defined in ISO 19103 with extensions in ISO 19109, and the General Feature Model – GFM defined in ISO 19109. UML is both a graphical and a lexical modelling language, and has become the most common language used for modelling information and software applications (Miles and Hamilton, 2006). The graphical view presented in UML diagrams is very useful for human communication, while more semantics needed for machine processing is described lexically. ISO 19103 and ISO 19109 contain specific rules of how the mechanisms of UML shall be used to add semantics for automated generation of implementation schemas in e.g. XML, based on the principles of Model Driven Architecture – MDA.

The ISO/TC 211 approach for UML modelling of geospatial information has been used for a wide range of applications and information models in the domain of geospatial information. These are models at agency level, national level, regional level and international level, and large amounts of geospatial information have been collected and maintained based on these models. One important set of models and data is the European INSPIRE Directive, (European Commission, 2007), which defines several spatial themes that are described in information models according to ISO 19109. For this purpose, INSPIRE has also defined specific information modelling rules in the Generic Conceptual Model (INSPIRE, 2013). Another important set of models are the application schemas developed by OGC, such as CityGML and InfraGML.

Geospatial information have been published on the Web in various forms for many years, both as web services and as download services. The ISO/TC 211 and OGC standards Web Map Service – WMS and Web Feature Service – WFS have been used extensively over the last 10-15 years, and Spatial Data Infrastructures – SDIs provide portals and catalogue services for searching and accessing information from several stakeholders. The services are mainly targeting geospatial applications and their users, and not so much The Semantic Web. However, mapping authorities in some countries have started to publish geospatial information for The Semantic Web, e.g. Ordnance Survey in the United Kingdom and Ordnance Survey in Ireland. Furthermore, OGC and W3C established The Spatial Data on the Web Working Group in cooperation in 2015 (Tand, van den Brink, and Barnaghi, 2017).

The purpose of this paper is to analyse methods for transformation of ISO/TC 211 conformant UML models of geospatial information to OWL ontologies for publication on The Semantic Web. Furthermore, based on the analysis, to identify possible challenges, and suggest improvements and further research.

2. LITERATURE REVIEW

The Object Management Group – OMG specification Ontology Definition Metamodel (Object Management Group, 2009) defines a metamodel and a UML profile for OWL, and describes transformations between UML and OWL in general. Transformations have been discussed in several articles as well, with similar approaches, but with differences in how complex UML characteristics they handle. Some articles, such as (Ferreira and Manuel, 2007, Gasevic et al., 2004, Gherabi and Bahaj, 2012) cover mainly classes, attributes and simple associations, while more complex methods described in e.g. (Bourahla and Belghiat, 2012b, a, Xu et al., 2012, Bahaj and

Bakkas, 2013, Hajjamy et al., 2016) also cover generalization, abstract classes, compositions and multiplicity.

The ISO/TC 211 standard ISO 19150, part 2 (ISO/TC 211, 2015c), defines standardized rules for transforming ISO 19103 and ISO 19109 compliant UML models to OWL. The ISO/TC 211 Group for Ontology Management – GOM (ISO/TC 211, 2018) have developed technologies for deriving ontologies from ISO/TC 211 compliant UML models, based on the rules in ISO 19150-2. For INSPIRE UML models, specific guidelines have been developed for transforming from UML to OWL (ARE3NA project, 2017).

Several research articles describe transformations of geospatial UML models to OWL. Some of the articles are early studies on the subject, written before the ISO/TC 211 and INSPIRE communities started to work on ontologies, and present some of the challenges and the benefits of preparing geospatial UML models for The Semantic Web. The use of ontologies for translation between data sources for land cover information was described already in 1999 in (Stuckenschmidt et al., 1999). Transformation of some core ISO and OGC UML models to ontologies are described in (Probst, Bibotti, and Pazos, 2004), while (Russomanno, Kothari, and Thomas, 2005) describe the transformation from UML to OWL for the OGC Specification SensorML. A similar transformation for the ISO/TC 211 and OGC standard for observation and measurement is described in (Probst, Gordon, and Dornelas, 2006) and later in (Cox, 2013, Cox, 2017). In (Buccella et al., 2011), an ontology based on the core ISO/TC 211 standards ISO 19107 and 19109 is described, while (Zedlitz and Luttenberger, 2012) discuss differences between UML and OWL and model transformation at a meta level, with reference to the UML Profile in ISO 19103.

One of the early articles on the use of geospatial information in The Semantic Web is (Egenhofer, 2002), where the concept of the Semantic Geospatial Web and research issues needed to enable it is described. In particular, two issues are pointed out: The need for a method for querying on geospatial characteristics, and the need for methods for enabling geospatial data sources for use in The Semantic Web. (Kolas, Hebel, and Dean, 2005) points out the way towards the Semantic Geospatial Web by suggesting an architecture of five types of geospatial ontologies: A base geospatial ontology, a geospatial service ontology, a filter ontology, domain ontologies and feature data source ontologies.

A state of art overview of methodologies for querying geospatial information on The Semantic Web is described in (Battle and Kolas, 2012), where the query language for geospatial information on The Semantic Web – GeoSPARQL (Perry and Herring, 2012) is introduced and described. Several articles show the practical use of geospatial information on The Semantic Web. The three articles (Aditya and Kraak, 2007, Klien, 2007, Lutz and Kolas, 2007) describe the potential of The Semantic Web for discovery in SDIs. Geospatial information is accessed as linked data from WFS in (Hietanen, Lehto, and Latvala, 2016), while (van den Brink et al., 2014) and (Patroumpas et al., 2015) describe a transformation of both UML models and data in GML format to OWL and RDF. (Karan, Irizarry, and Haymaker, 2015) describe and demonstrate how Semantic Web technologies can be used to integrate and query data sets from the GIS and BIM domains, while the ongoing INTERLINK Project (Luiten et al., 2017) uses Semantic Web technologies for combining geospatial information from different domains and stakeholders.

3. TRANSFORMING MODELS FOR GEOSPATIAL INFORMATION TO OWL

3.1 Fundamental differences

Several articles, e.g. (Noy and Klein, 2004, Kiko and Atkinson, 2008, ARE3NA project, 2017), point out differences between UML and OWL that are important to be aware of. One fundamental difference is the assumptions of an open or a closed world. The Open World Assumption – OWA and the possibility for anyone to say anything about anything is an important part of information modelling for The Semantic Web, while UML models based on the ISO/TC 211 approach are limited to the closed context of geographic information, and are assumed complete in that context. To preserve the Closed World Assumption – CWA of the original model, the transformation may need to include some restrictions. Furthermore, ontologies are reusing and extending other ontologies, including ontologies from other domains. UML models based on the ISO/TC 211 approach are reusing concepts from other models as well, but mainly limited to models from the domain of geospatial information. Finally, the logic in ontologies is based on set theories, with set-based class relations such as disjoint, union, intersect and equivalent. UML is not based on the same logic, but some restrictions from UML models must be translated to these kinds of relations in OWL.

3.2 Packages

UML packages correspond to OWL ontologies, and the transformation is described as straightforward, where the package name becomes the ontology name in (Object Management Group, 2009), (Bourahla and Belghiat, 2012a) and in ISO 19150-2. The INSPIRE Guidelines states that a package stereotyped as application schema according to ISO 19109 shall be converted into a single ontology with name and namespace derived from the tagged value “xmlns” on the UML package.

3.3 Classes

The concepts of *class*, *generalization* and *inheritance* exists in both UML and OWL. A UML class is simply transformed to an OWL class, while a UML generalization is transformed to an OWL *subclassOf* axiom.

The UML concept of *abstract classes* defines classes that shall not have instances, and is often used in generalizations where the abstract class is a *superclass* used for defining attributes, associations and operations that are common to all subclasses. The concept does not exist in OWL, and must be handled by using other mechanisms. ISO 19150-2 introduce an annotation property “*isAbstract*” for this purpose, and the INSPIRE Guidelines use this property as well. However, there are no rules connected to the property, so it may still be possible to create instances of the abstract class. (Zedlitz and Luttenberger, 2012) suggest to use the *DisjointUnion* axiom for abstract classes and subclasses, but as stated in the article, this will not prohibit creating instances of the abstract class directly.

Method	Described by
isAbstract	ISO 19150-2, INSPIRE
DisjointUnion	(Zedlitz and Luttenberger, 2012)

Table 1. Methods for transforming abstract classes

Multiple inheritance, where a class is a subclass of more than one class, is sometimes used in UML, but is a problematic issue for implementation in e.g. XML. The rules for conversion from UML models to implementation schemas in ISO 19136 (ISO/TC 211, 2007) do not support multiple inheritance, and ISO 19103 recommends that multiple inheritance is avoided unless really needed. However, multiple inheritance is still used in some UML models for geospatial information, e.g. some INSPIRE models (INSPIRE, 2010). Multiple *subclassOf* predicates is very common in OWL, and the INSPIRE guidelines states that multiple inheritance shall be handled with multiple *subclassOf* predicates. In (Hajjamy et al., 2016), a more specific representation is applied, with the subclass as an *intersection* of the superclasses, to make sure the subclass follows restrictions from all of its superclasses.

3.4 Data types

Data types for attributes in UML can be classified as primitive and complex. *Primitive data types* are types with atomic values, such as integer, string and boolean. A set of primitive data types are defined in ISO 19103, and according to both ISO 19150-2 and the INSPIRE Guidelines, these are mapped to equivalent XML Schema (XSD) data types and referred to as *DatatypeProperties*. The same approach is followed by (Hajjamy et al., 2016).

Complex data types have an internal structure. They have own attributes and associations, possibly with complex data types too. ISO 19103 defines a range of such data types: for measures (angle, length etc.) that require a unit of measure in addition to the value; name types; record types; and more. Other standards based on ISO 19103 defines other and more domain specific data types. There are some differences in how complex data types are transformed according to the rules in ISO 19150-2 and the INSPIRE Guidelines. ISO 19150-2 have a short list of data types from ISO 19103 that are mapped to equivalent XSD data types and referred to as *DatatypeProperties*, while all other data types are converted to OWL Classes, and referred to as *ObjectProperties*. The INSPIRE Guidelines focus more on reusing existing ontologies, and have a longer list of predefined mappings of commonly used data types, from both ISO 19103 and other ISO Standards. These are mapped to equivalent or similar datatypes from both XSD and other ontologies. As stated in the guidelines, this may lead to a loss of information when transforming from GML to RDF and back, because of differences between the UML data types and the external data types. Of the other mappings described in articles, only (Hajjamy et al., 2016) describe mapping of complex data types, and they are then converted to OWL classes referred to as *ObjectProperties*.

Method	Described by
Mapping a few, else new class	ISO 19150-2
Mapping all similar, else new class	INSPIRE

Table 2. Levels of mapping to existing data types

3.5 Enumerations

Enumerations in UML are data types that define complete lists of possible values. Attributes using the enumeration can only have values defined in the list, and no other value. OWL has the equivalent axiom *oneOf*, which can be used to restrict both *DatatypeProperties* (*DataOneOf*) and *ObjectProperties*

(*ObjectOneOf*). Both ISO 19150-2, the INSPIRE Guidelines, (Zedlitz and Luttenberger, 2012) and (Hajjamy et al., 2016) describe the transformation of UML enumerations to DatatypeProperties with the *oneOf* axiom and a collection of values from the UML enumeration. However, the INSPIRE Guidelines state that only enumerations with self-describing codes that have an obvious meaning shall be represented with the *oneOf* axiom. In other cases, the enumeration shall be handled as a separate *SKOS Concept Scheme*, and the range for the attribute shall refer to the generic class *skos:Concept*. A *seeAlso* statement is added with the URL to the SKOS Concept Scheme, to describe where the list is to be found, but without any actual binding. This is the same approach as used for code lists in the INSPIRE Guidelines.

3.6 Code lists

The ISO/TC 211 UML profile in ISO 19103 defines code lists as flexible enumerations, meaning that there can be other values than those described in the list. This is an important issue, as one cannot expect that all possible values are described in the model or the ontology, they may be described elsewhere. The INSPIRE Guidelines use the same approach for code lists as described for enumerations without self-describing codes. ISO 19150-2 also uses SKOS Concept Schemes for code lists, but with a closer binding than the approach in INSPIRE. The code list is defined both as a class, a concept scheme and a collection, where the class is a subclass of *skos:Concept*. The binding of the code list and the attribute is not described in the standard, but in the ISO/TC 211 official ontologies, the attributes are bound to the code list class with the *allValuesFrom* axiom. This close binding excludes the possibility for additional values described in other SKOS Concept Schemes.

(Zedlitz and Luttenberger, 2012) describe a different approach. They use a *UnionOf* axiom to define a union, and let the union include an *OneOf* list with the code list values, and any other value, defined with a standard XML Schema expression that is also used for code lists in GML.

Method	Described by
SKOS and <i>allValuesFrom</i>	ISO 19150-2
SKOS	INSPIRE
<i>DataUnionOf</i> and any value	(Zedlitz and Luttenberger, 2012)

Table 3. Handling of code lists

3.7 Union

The ISO/TC 211 UML profile in ISO 19103 defines unions as types with a list of several alternative datatypes, where one and only one shall be used for an attribute value. A union is similar to an enumeration, except that the values in the list are data types, not literals. This is a different meaning of union than the set-based union in OWL, where a union contains every individual contained in at least one of the classes in the union. (Zedlitz and Luttenberger, 2012) describe two possible methods for transforming a UML union to OWL. The first method cover the situation where all members in the UML union can be transformed to either *DataProperties* or *ObjectProperties*. They define a property (either data or object) for each member, and an additional property that all members are a subproperty of, and which has a cardinality of exactly one (*ExactCardinality*). By using this property as range for attributes from the UML model, only one of the members of the UML union can be

selected. However, because of the Open World Assumption, this method does not avoid the use of other properties that are not members of the union. The other method cover situations where some of the members in the union can be transformed to *DataProperties*, and some to *ObjectProperties*. A class is defined for each member of the union, and these classes are set as disjoint from each other. Each of the new classes is set to be equal to a set of exactly one of the current union member. The downside of this solution is that it gets much more complex with many axioms.

In ISO 19150-2, a UML union is simply transformed to an OWL union. As long as only one member is assigned to each UML attribute, this will give the correct representation. However, with the OWL Union, several members from the union can be assigned for the same instance of the UML attribute, which breaks the rules for a UML union. The INSPIRE Guidelines have a fourth approach, where each member of the union is transformed to a property. For each of these new properties, an intersection with all union members is defined, where cardinality expressions are used to define that only this property can have a value. A union expression combines all the intersections, but because of the cardinality restrictions in each intersection, choosing one of them will exclude the others. So only one of the transformed properties in the union may be used. Like the second method from (Zedlitz and Luttenberger, 2012), this method is quite complex, but it seems to maintain the purpose of a UML union.

Method	Described by
Union	ISO 19150-2
Intersection and union	INSPIRE
Subproperty and <i>ExactCardinality</i>	(Zedlitz and Luttenberger, 2012)
Disjoint classes	(Zedlitz and Luttenberger, 2012)

Table 4. Methods for transforming UML Unions

3.8 Attributes and association roles

UML has two ways of describing further characteristics of a class: As *attributes* with primitive or complex data types, or as *associations* to other classes. Both of these are similar to properties in OWL: In principle, attributes with simple (primitive) data types are equivalent to *DataProperties*, while attributes with complex data types and association roles are equivalent to *ObjectProperties*, as they refer to another class. However, there is one fundamental difference: While a UML class is the single owner of its attributes and associations, properties in OWL are globally defined and may be assigned to any class. Several classes in a UML model may have attributes or association roles that are identical on all classes, having identical name, data type and definition, or they may be almost identical. Furthermore, several classes may have an attribute or an association role with identical name, but different data type and/or definition. When UML attributes and association roles shall be transformed to OWL properties, these issues need to be handled. Identical attributes and roles should preferably be handled as one global property, assigned to the respective classes, while attributes and roles with identical names, but different data type and/or definition need to be handled as separate properties, with different identifiers. The attribute or role name alone may lead to duplicate properties with different meanings.

The referred articles, standards and guidelines handle these issues to various degrees, and with various approaches. (Gasevic et al., 2004, Gherabi and Bahaj, 2012, Bahaj and Bakkas, 2013, Hajjamy et al., 2016) do not refer to these issues at all, and just perform a simple transformation to properties. While (Xu et al., 2012, Zedlitz and Luttenberger, 2012, Cox, 2013, Cox, 2017) only reflect over the issues without proposing a method to solve them, and (Buccella et al., 2011) performs a similarity matching to identify global concepts. The rule in ISO 19150-2 is to add the class name as a prefix, and thereby create unique properties from all attributes and association roles.

The INSPIRE Guidelines takes a more advanced approach, striving to achieve globally scoped properties and reuse of existing properties when possible. Properties that have identical or close to identical meaning (but not necessarily identical name) shall be converted to properties with a global scope, and properties that are similar to properties already defined in other ontologies shall be converted to those properties. The guidelines recognize that the process of identifying these properties require a manual review. For all other properties, the class name is added as a prefix, following the rules from ISO 19150-2.

Method	Described by
Prefix	ISO 19150-2
Manual matching	INSPIRE
Similarity matching	(Buccella et al., 2011)

Table 5. Methods for attribute globalization

3.9 Associations

An association in UML is a relationship between two classes, is similar to an attribute, and is implemented in the same manner as attributes in e.g. XML. The transformation of a simple association from UML to OWL is done by creating an ObjectProperty with one class as the *domain* and the other class as the *range* of the property. This approach has been followed in all the referred articles, and also in ISO 19150-2 and the INSPIRE Guidelines.

A UML aggregation, also known as shared aggregation, is a more specific relationship, where the associated class is a part of the main class (the whole). This kind of association does not add any actual semantics to the model; it just describes a closer relationship. Aggregation is described as an association type in several articles, but (Bahaj and Bakkas, 2013) is the only article that describe a method for maintaining the aggregation in the OWL model. They have created a hierarchy of properties representing relationship types, with association at the top, and aggregation and composition as subproperties. Every association from the UML model is transformed to a subproperty of one of these properties. This way, possible semantics may be added to the different relationship types. ISO 19150-2 maintains information about the relationship type by adding an annotation with *aggregationType*.

A UML composition, also known as composite aggregation, is a stronger relationship between two classes. An instance of a class, related to another class through a composition, can only take part in one composition, i.e. the part instance can only be related to one whole instance at the same time. Often, but not always, the part will also be deleted if the related whole is deleted.

ISO 19150-2 handles compositions in the same manner as aggregations, with the *aggregationType* annotation. Composition is also described in several articles, but (Bahaj and Bakkas, 2013) and (Hajjamy et al., 2016) are the only articles that describe a way of maintaining it. In (Bahaj and Bakkas, 2013), this is done with the described hierarchy of relationship types, but no semantics is added to the composition property. (Hajjamy et al., 2016) describe a method for maintaining the restrictions by setting the property that represent the composition as *InverseFunctional*, which means that the associated class can only be linked to one class through this property. Furthermore, they also add restrictions saying that the composition cannot be from the related class itself (*Irreflexive*) and that the composition cannot be applied in the other direction (*Asymmetric*).

Method	Described by
<i>aggregateType</i>	ISO 19150-2, INSPIRE
Property hierarchy	(Bahaj and Bakkas, 2013)
<i>InverseFunctional</i>	(Hajjamy et al., 2016)

Table 6. Methods for transforming compositions

4. DISCUSSION

4.1 The scope of geospatial ontologies

Even though both The Semantic Web and digital geospatial information has existed for many years, the use of geospatial information in The Semantic Web is still limited. Large amounts of geospatial information are available on the Web, but almost solely in domain specific Web services or download services from the GIS domain. One example of this is the European INSPIRE Geoportal, with more than 58000 data sets available through WMS or WFS, and more than 37000 data sets available for download, but no information or information models available RDF or OWL. However, as described in several research articles, these services and the information provided by them can still be used as resources in The Semantic Web. Metadata may be converted to RDF and used for querying and discovery. The application schemas provided by the web services may be converted to ontologies, and data in GML format may be converted to RDF, on request or as complete exports. Both (van den Brink et al., 2014), (Hietanen, Lehto, and Latvala, 2016) and (Patroumpas et al., 2015) describe how such functionality can be built as extensions to existing SDIs, and thereby making the large amounts of geospatial information available for The Semantic Web.

(Noy and McGuinness, 2001) states that the first step in ontology development shall be to determine the scope of the ontology. The scope of geospatial ontologies is important for deciding how well the transformation of UML models to ontologies shall maintain the closed world-based concepts and restrictions from the UML model, and for deciding if similar but not identical concepts from existing ontologies can be reused. At least three possible main scopes are relevant for geospatial ontologies:

1. Enable geospatial information from GIS applications for The Semantic Web, by unidirectional information exchange.
2. Enable bidirectional exchange of information between GIS applications and The Semantic Web
3. Replace GIS databases and applications with triple stores and query engines in The Semantic Web.

The third scope is not a likely situation a short term, mainly because of the complexity and advanced functionality that exists in GIS applications and databases that have been developed specifically for handling complex geometries and topologies, and operations on these. As discussed in e.g. (Tand, van den Brink, and Barnaghi, 2017), it will not be convenient to replace all of this in triple stores and query engines. The second scope is applied in the INTERLINK project (Luiten et al., 2017), where information will be exchanged bi-directionally between application domains, stakeholders and lifecycle phases, but where the original information models are mainly developed directly in OWL. This is a more likely scope for the ontologies, in which case more restrictions will need to be included in OWL. However, the most discussed scope, e.g. in the INSPIRE Guidelines and in (Tand, van den Brink, and Barnaghi, 2017), is to store the information in GIS databases and transform it to RDF. With this scope, less strict transformations can be applied.

4.2 Transformation issues

The studies indicate that the transformation methods handle the basic UML concepts *classes*, *generalizations*, *primitive data types*, *enumerations* and *simple associations* similarly and acceptable. For other UML concepts, several approaches have been used, and the level of complexity needed will depend on the scope of the ontologies. If all restrictions shall be maintained, the transformation need to be more complex.

The concept of *abstract classes* is widely used in UML models, but only briefly handled in the transformation methods. None of the methods fully maintain the concept of abstract classes; it may still be possible to create instances of the classes in RDF. A solution that may maintain the concept better is to transform only the properties of the abstract class to OWL and not the class itself, and then assign the properties to each subclass in OWL instead. This would also be closer to implementations in databases, where the abstract class itself will not be implemented. This approach has not been discussed in any of the articles.

Reusing data types and classes from existing ontologies is a fundamental part of The Semantic Web. All *primitive data types* from ISO 19103 and some *complex data types* can be transformed directly to XML Schema data types. This is a simple mapping for primitive data types, while for complex data types there is a question on whether or not the data types are identical, and how identical they need to be. The INSPIRE Guidelines have specified mapping to more existing data types than ISO 19150-2, including mappings to similar but not identical types, which may lead to information loss due to minor differences. An approach that may improve this in general is to reuse more existing data types in the original UML model. The method described by (van den Brink et al., 2014), where elements that shall be linked to existing ontologies are tagged in the UML model, may be used to support this approach.

The methods for handling *Code lists* according to ISO 19150-2 and the INSPIRE Guidelines both describe the use of complete SKOS Concept Schemes. However, there are some weaknesses in the methods. The use of the *allValuesOf* axiom in the official ISO/TC 211 ontologies excludes values that are not in the defined code lists. This breaks the concept of a code list as an open enumeration. The solution described in (Zedlitz and Luttenberger, 2012), with a union of defined values and any other value, defined with a standard XML Schema expression,

seems to solve this better. The INSPIRE Guidelines recommend to use separate SKOS Concept Schemes also for *enumerations* that do not have obvious meaning. This seems good for being able to describe the values in the enumerations, but it removes the restriction that lies in the concept of an enumeration as a closed list.

The handling of a UML *union* in OWL is a complex issue that is solved with different methods in ISO 19150-2, the INSPIRE Guidelines and in (Zedlitz and Luttenberger, 2012). The most complete method seems to be the one described in the INSPIRE Guidelines, but it is also quite complex. One method that may simplify this is to use the *ObjectOneOf* axiom, similar to the way enumerations are handled.

For *attributes and association roles*, a main question is how to handle attributes or association roles that are identical for several classes, how to handle those that have almost identical names and meaning, and those that have identical names but different meaning. The approach in ISO 19150-2, with class name as prefix for all properties, makes all properties globally unique, but excludes reuse of properties globally. The INSPIRE Guidelines suggests to harmonize properties internally in the ontology, and with external ontologies, to enable reuse, which will make the ontologies differ from the UML model. Once again, this is acceptable with the scope of unidirectional information exchange. However, a weakness is that the harmonization must be done manually, but string-matching algorithms, as described in (Buccella et al., 2011) may support the process. An additional approach, as used in (Hietanen, Lehto, and Latvala, 2016), would be to improve the UML models from the ontologies, and thereby achieve harmonization between the UML and OWL representations. Including tagging of attributes and association roles in the UML model that shall be linked to existing or similar internal attributes or association roles, as discussed for complex data types and in (van den Brink et al., 2014), would also be an improvement. Furthermore, a use of properties and subproperties for almost identical attributes is also a method that may be used.

Of the described methods for transformation of *associations*, only (Hajjamy et al., 2016) include a solution for maintaining the semantics of a composition. However, both the *aggregationType* annotation that is used in ISO 19150-2 and the INSPIRE Guidelines, and the association type hierarchy that is used in (Bahaj and Bakkas, 2013) may be extended to include such semantics. Combining one of these approaches with the method from (Hajjamy et al., 2016) is a possible solution that may improve the handling of association restrictions.

5. CONCLUSIONS AND RESEARCH RECOMMENDATIONS

5.1 Conclusions

In this paper, methods for transforming UML models of geospatial information has been studied and discussed. This subject, and methodologies for enabling geospatial information from SDIs for The Semantic Web, has been studied for many years and in several projects. Still, geospatial information is mainly available in domain specific Web services for geospatial information in SDIs, and not as OWL and RDF. Research indicates that a method for enabling models and information from SDIs for The Semantic Web can be to extend existing Web services and download services with transformation functionality to OWL and RDF.

The information models for geospatial information are mainly developed using UML, based on standards from ISO/TC 211 and OGC. An important part of enabling geospatial information for The Semantic Web will be to transform these UML models to OWL ontologies. Methods and rules for transforming from UML to OWL in general, and in particular for UML models of geospatial information, has been studied in several research articles, and standardized rules for the transformation have been developed in ISO/TC 211, as well as guidelines in INSPIRE. The review in this paper indicates that most of the transformation is straight forward, but some fundamental differences between UML and OWL must be handled. Three fundamental differences are particularly important: First, the UML models represents a closed world in a geospatial context, while OWL operates in an open world. The UML concepts of abstract classes, enumerations, unions and aggregations represent restrictions in the model that need special handling if they shall be maintained in the ontology. This has been done in diverse ways in the methods that has been studied, but none of them fully maintain all restrictions. Second, attributes and associations in UML are uniquely owned by each class, while properties in OWL are globally scoped, and one property may be used for many classes. Several classes in UML may have identical attributes and association roles; almost identical attributes and association roles; or attributes and association roles with identical name but different meaning. All of these need to be handled in the transformation to OWL, and several approaches have been used here as well, each with strengths and weaknesses. Third, an important principle in ontology development is reusing elements from existing ontologies, while UML models mainly use elements internally in the model. For this matter, mapping to existing ontologies have been defined in several ways.

An important question to be answered for the transformations is whether the ontologies shall be used for unidirectional or bidirectional information exchange. In the case of unidirectional information exchange, the need for maintaining restrictions from the UML model is not so important, as the information shall only be transformed from UML-based databases to RDF. In the case of bidirectional information exchange, where information shall also be transformed back to UML-based databases, instances may possibly be created in RDF. Maintaining the restrictions from the UML models is then more important, or else instances may be created that are illegal according to the UML model.

5.2 Research recommendations

Based on the discussion and the conclusions, two main topics have been identified as candidates for further research:

The semantics of some UML concepts cannot be directly transformed to OWL concepts, in particular abstract classes; code lists; unions; and aggregations. Possible approaches and improvements have been discussed in this paper, and should be studied further and tested in implementations, by extending the technologies developed by ISO/TC 211 GOM (ISO/TC 211, 2018).

For attributes and associations, and for complex data types, there is a question of how to handle singular versus globally owned properties, and the reuse of elements from existing ontologies. Some improvements have been discussed in this paper, including methods for reusing and marking elements

from external ontologies in the UML models, and methods for defining global properties in the UML models. Experiences from the work on ontologies may be brought into the work on UML models, to overcome some of the challenges. Further research on these issues should include methods for defining global and external elements in the UML models, and methods for similarity matching with internal and external elements.

ACKNOWLEDGEMENTS

The author would like to acknowledge the support from Rune Hjelsvold, Erling Onstein and Lizhen Huang with this article.

REFERENCES

- Aditya, Trias, and Menno-Jan Kraak, 2007. A Search Interface for an SDI: Implementation and Evaluation of Metadata Visualization Strategies. *Transactions in GIS*, 11 (3):413-435, doi: 10.1111/j.1467-9671.2007.01053.x.
- ARE3NA project, 2017. Guidelines for the RDF encoding of spatial data. accessed 2018-07-04. <http://inspire-eu-rdf.github.io/inspire-rdf-guidelines/>.
- Bahaj, Mohamed, and Jamal Bakkas, 2013. Automatic Conversion Method of Class Diagrams to Ontologies Maintaining Their Semantic Features. *International Journal of Soft Computing and Engineering (IJSCE)*:65-69.
- Battle, R., and D. Kolas, 2012. Enabling the geospatial Semantic Web with Parliament and GeoSPARQL. *Semantic Web*, 3 (4):355-370, doi: 10.3233/Sw-2012-0065.
- Bourahla, Mustapha, and Aissam Belghiat, 2012a. Transformation of UML models towards OWL ontologies. *2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications, SETIT 2012*.
- Bourahla, Mustapha, and Aissam Belghiat, 2012b. UML Class Diagrams to OWL Ontologies: A Graph Transformation based Approach. *International Journal of Computer Applications*, 41 (3):41-46, doi: 10.5120/5525-7566.
- Buccella, A., A. Cechich, D. Gendarmi, F. Lanubile, G. Semeraro, and A. Colagrossi, 2011. Building a global normalized ontology for integrating geographic data sources. *Computers & Geosciences*, 37 (7):893-916, doi: 10.1016/j.cageo.2011.02.022.
- Cox, S. J. D., 2017. Ontology for observations and sampling features, with alignments to existing models. *Semantic Web*, 8 (3):453-+, doi: 10.3233/SW-160214.
- Cox, Simon, 2013. An explicit OWL representation of ISO/OGC Observations and Measurements. *SSN@ ISWC*:1-18.
- Egenhofer, Max J, 2002. Toward the semantic geospatial web. *Proceedings of the 10th ACM international symposium on Advances in geographic information systems*.
- European Commission, 2007. DIRECTIVE 2007/2/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 14 March 2007 establishing an Infrastructure for Spatial Information in the European Community (INSPIRE). *Official Journal of the European Union*.
- Ferreira, David De Almeida, and Alberto Manuel, 2007. UML to OWL Mapping Overview An analysis of the translation process and supporting tools. *7th Conference of Portuguese Association of Information Systems (CAPSI'06)*.

- Gasevic, Dragan, Dragan Djuric, Vladan Devedzic, and Violeta Damjanovi, 2004. Converting UML to OWL ontologies. *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters - WWW Alt '04*.
- Gherabi, N., and M. Bahaj, 2012. A New Method for Mapping UML Class into OWL Ontology. *International Journal of Computer Applications*:5-9.
- Hajjamy, O. E., K. Alaoui, L. Alaoui, and M. Bahaj, 2016. Mapping UML to OWL2 ontology. *Journal of Theoretical and Applied Information Technology*, 90 (1):126-143.
- Hietanen, E., L. Lehto, and P. Latvala, 2016. Providing Geographic Datasets as Linked Data in Sdi. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B2:583-586, doi: 10.5194/isprs-archives-XLI-B2-583-2016.
- INSPIRE, 2010. INSPIRE Data Specification on Transport Networks – Guidelines.
- INSPIRE, 2013. INSPIRE Generic Conceptual Model.
- ISO/TC 211, 2007. ISO 19136:2007 Geographic information — Geography Markup Language (GML).
- ISO/TC 211, 2014. ISO 19101-1:2014 Geographic Information — Reference model — Part 1: Fundamentals.
- ISO/TC 211, 2015a. ISO 19103:2015 Geographic Information — Conceptual schema language.
- ISO/TC 211, 2015b. ISO 19109:2015 Geographic information — Rules for application schema.
- ISO/TC 211, 2015c. ISO 19150-2:2015 Geographic information — Ontology — Part 2: Rules for developing ontologies in the Web Ontology Language (OWL).
- ISO/TC 211, 2018. ISO/TC 211 Group for Ontology Management. accessed 2018-07-04. <https://github.com/ISO-TC211/GOM>.
- Karan, Ebrahim P, Javier Irizarry, and John Haymaker, 2015. BIM and GIS integration and interoperability based on semantic web technology. *Journal of Computing in Civil Engineering*, 30:4015043.
- Kiko, K, and Colin Atkinson, 2008. A detailed comparison of uml and owl. *Technical Report 4, Department of Mathematics and Computer Science, University of Mannheim*:1-58.
- Klien, Eva, 2007. A Rule-Based Strategy for the Semantic Annotation of Geodata. *Transactions in GIS*, 11 (3):437-452, doi: 10.1111/j.1467-9671.2007.01054.x.
- Kolas, D, J Hebler, and M Dean, 2005. Geospatial semantic web: Architecture of ontologies. *Geospatial Semantics, Proceedings*.
- Luiten, Bart, Aonghus O'Keefe, Sander Stolk, Lars Wikström, and Matthias Weise, 2017. INTERLINK D4. Principles for a European Road OTL.
- Lutz, Michael, and Dave Kolas, 2007. Rule-Based Discovery in Spatial Data Infrastructure. *Transactions in GIS*, 11:317- 336.
- Miles, Russ (Russell), and Kim Hamilton, 2006. *Learning UML 2.0*. First edit ed: O'Reilly.
- Noy, Natalya F, and Michel Klein, 2004. Ontology evolution: Not the same as schema evolution. *Knowledge and information systems*, 6:428-440.
- Noy, Natalya F., and Deborah L. McGuinness, 2001. Ontology Development 101: A Guide to Creating Your First Ontology. *Stanford Knowledge Systems Laboratory*:25, doi: 10.1016/j.artmed.2004.01.014.
- Object Management Group, 2009. Ontology Definition Metamodel, Version 1.0.
- OGC, 2018. Glossary of terms. accessed 2018-07-10. <http://www.opengeospatial.org/ogc/glossary>.
- Patroumpas, K., N. Georgomanolis, T. Stratiotis, M. Alexakis, and S. Athanasiou, 2015. Exposing INSPIRE on the Semantic Web. *Journal of Web Semantics*, 35:53-62, doi: 10.1016/j.websem.2015.09.003.
- Perry, Matthew, and John Herring, 2012. OGC GeoSPARQL-A geographic query language for RDF data.
- Probst, Florian, Fernando Bibotti, and Andres Pazos, 2004. Connecting ISO and OGC Models to the Semantic Web | OGC Network. *The Third International Conference on Geographic Information Science: Extended Abstracts and Poster Summaries, Adelphi, MD, USA*.
- Probst, Florian, Andrew Gordon, and Ivan Dornelas, 2006. OGC Discussion paper: Ontology-based representation of the OGC observations and measurements model. In *Institute for Geoinformatics (ifgi)*: Citeseer.
- Russomanno, David J., Cartik R. Kothari, and Omoju a. Thomas, 2005. Building a Sensor Ontology: A Practical Approach Leveraging ISO and OGC Models. *Proceedings of International Conference on Artificial Intelligence*.
- Stuckenschmidt, H, U Visser, G Schuster, and T. Vögele, 1999. Ontologies for geographic information integration. *Proceedings of the workshop intelligent methods for handling environmental information: Special aspects of processing space and time*.
- Tand, Jeremy, Linda van den Brink, and Payam Barnaghi, 2017. Spatial Data on the Web Best Practices. In *W3C Working Group Note*.
- van den Brink, Linda, Paul Janssen, Wilko Quak, and Jantien Stoter, 2014. Linking spatial data: automated conversion of geo-information models and GML data to RDF. *International Journal of Spatial Data Infrastructures Research*, 9:59-85, doi: 10.2902/.
- Xu, Z. M., Y. Y. Ni, W. J. He, L. L. Lin, and Q. Yan, 2012. Automatic extraction of OWL ontologies from UML class diagrams: a semantics-preserving approach. *World Wide Web-Internet and Web Information Systems*, 15 (5-6):517-545, doi: 10.1007/s11280-011-0147-z.
- Zedlitz, Jesper, and Norbert Luttenberger, 2012. Transforming Between UML Conceptual Models And OWL 2 Ontologies. *Proceedings of the Terra Cognita Workshop on Foundations Technologies and Applications of the Geospatial Web at the 11th International Semantic Web Conference ISWC 2012*.