# Strategies for Publishing Domain Ontologies as Linked Data

## Introduction

ELFIE has explored publishing environmental observations as linked data, with data interpretation supported by having unambiguous references to key concepts in the data represented by additional links (URIs). Such concepts cover the nature of the properties of a data object, but also the key Use Case of linking an observation to the Feature of Interest.

## Problem statement

Common semantics for such properties and the type of the Feature of Interest, are identified by using published models – such as HY\_Features, GWML. These models (currently) are available in a normative forms as UML diagrams, an in some cases as XML schema, neither of which is conducive to dereferencing a URI for a concept to get more detail about that specific concept, or even to confidently match the identity of any references to concepts.

The requirement therefore is to be able to publish such models, as stable URIS, dereferencing as fine-grained Linked Data views. This means that a predictable URI naming scheme for elements in a model is required. (this may not be strictly necessary for a single model, but when a body of such models are managed by a single authority – such as OGC and its delegated Specification Working Groups – then commonality is necessary to avoid revisiting the naming strategy for each case, and to allow users to become familiar and comfortable with a consistent product.)

Within ELFIE context the need has been identified for at least two models, HY\_Features and GWML, and their multiple component Application Schema. The GeoSciML community is also looking at the same issues. There is a need for both efficiency and consistency to harmonise and document a common approach.

ISO 19150-2 defines a set of rules for encoding UML as an OWL ontology referencing the suite of OWL artefacts created to model the ISO Harmonised Model.

This form of OWL has an unknown utility – by tying data types and concepts into the extensive ISO model this adds a high burden on clients to reason over the entire model (quite a large set of content) in order to identify fairly simple, but highly important, semantic baseline information – such as that a property may be treated as a xsd:integer for purposes of a calculation.) The feasibility of such reasoning is unknown, as there is no way to test the ISO models behave as expected under reasoning conditions.

Furthermore, for the purposes of Linked Data, much of the modelling of behavior of low-level data types is not relevant, when the main requirement is to identify concepts, access explanations, or potentially access information about implementations of these concepts that may be available.

Finally, from the perspective of infrastructure support, OGC provides a “definitions server” that is designed around SKOS meta-model and will provide dereferencing services and Linked Data representations.

## Background

BRGM have performed an initial analysis of the issues and questions that arise in the context of:

“

* Generating ontologies for a subset of GeoSciML: GeoSciML Basic, Borehole and Lite,
* Testing populating instances for Boreholes, Geologic Units pointing to vocabs when available (FR, EU, CGI GTWG) and testing some representations (maps, graph).

“

(<https://github.com/BRGM/GeoSciMLontology/blob/master/documents/ISO191xx_2_OWL_NoteBRGM.docx>)

These are relevant to the equivalent challenges for ELFIE, and collaboration with the GeoSciML, GWML and ELFIE activities has been initiated to develop a common solution.

The issues identified by BRGM have been further characterized from the perspective of definition publication process and governance. The following types of issues have been characterized:

1) weaknesses or issues with ISO 19150-2 rules

2) choices of options

3) non-canonical forms of additional constraints in UML and or OCL or tags

4) implementation choices for specific communities

5) meta-model issues (expressivity mismatches between OWL and UML)

6) bugs and limitations in software (or things too hard to configure)

7) best practices for documentation and annotation

The key meta-model issue I see here is the use of a character string (UML option) to hold an IRI in a particular implementation profile - and the trickiness of modelling this as an objectPRoperty or not. Option could be to model it as an rdfs:Property, and allow implementation profiles to constrain it to an owl:ObjectProperty.

## Proposed strategy

Resources, time and testing methodology for “hand building” optimal equivalent ontologies are not available (notwithstanding comments that these may lead to iterative refinement and improvement of the model).

The only automated pathway available to all participants currently is using Shapechange, which means developing a set of encoding rules from a wide range of choices and options from previous experiments in encoding different styles of UML model.

The ISO 19150-2 artefacts will be seen as an intermediate artefact (available by an annotation property reference from final product.)

“Manual interventions” to artefact to correct bugs or apply further rules will be encoded as RDF transformations or additional statements that can be applied in sequence.

Rules to extract basic OWL and xsd: datatype equivalence, without embedded reference to ISO datatypes will be developed and applied (Rob Atkinson as part of OGC-NA infrastructure)

Rules to extract SKOS equivalent glossary/taxonomy from OWL class models will be developed and applied (Rob Atkinson as part of OGC-NA infrastructure)

URIs under in [www.opengis.net/def](http://www.opengis.net/def) will be assigned, and content will be marked as draft, as an exercise in publication governance by OGC-NA, to make content and process available for wider discussion.

## Characterisation of issues

(starting with BRGM notes)

### Weaknesses or issues with ISO 19150-2 rules

* « The rules of ISO 19150-2 are restrictive on many aspects if we respect them all. Respecting all of them means we don’t take into account the open world assumption when working with ontologies”
* The transformation rules are consistent but limits the resulting ontologies to the UML paradigm. Some additional work may be done on the resulting ontologies to add semantics between classes (disjunctions, subsumption, equivalence, etc) and within or between properties (functional properties, transitive properties, symmetric properties, inverse of, etc).
* No specific indications about association classes are mentioned in the norm. It is obvious that an association class is translated as an OWL class. No rule for linking this class to the related class(es) appear

### Choices of options

* Properties naming when translating attributes: dots in properties identifiers could be interpreted somehow that they are still scoped to classes, while in ontologies, properties are scoped to a name space instead. Properties are independent entities that may or may not have a specific class as a domain. This is one major structural difference between UML and OWL.
* Basic types (from SWE types for example) must be modified if needed by specialized Classes from other ontologies or by defining new ones
* Use general (non-scoped to class) property names when the name of the attribute or association is unique. Thus, leave the domain of the properties open (or typed as owl:Thing). The restrictions on the properties values in the class definition can be used for this purpose.
* Allow scoped names for properties (class.Property) then verify whether automatically created properties can be merged into one (eg. GeologicFeature.purpose and EarthMaterial.purpose)
* Use GSML\_QuantityRange instead of swe:QuantityRange as recommended
* Rename swe:Category to skos:Concept or mdl:Lineage (depending on the case) and swe:Quantity to the relevant class in the context (ts:TimePosition, mdq:PositionalAccuracy, etc.)
* For the properties of GeoSciML Basic and Borehole to be reused in GeoSciML Lite, we activate the ShapeChange rule “rule-owl-prop-globalScopeByUniquePropertyName” that scopes unique name property to global use, and thus not specify the domain of these properties. The scoping of the properties to their classes in Basic and Borehole is done using restrictions on the values that these properties can take for their corresponding classes. This can be done thanks to the ShapeChange rule “rule-owl-prop-range-local-withUniversalQuantification”

### non-canonical forms of additional constraints in UML and or OCL or tags

* The requirements of the model cannot be all respected in the ontology representation (eg. “QuantityRange properties that must report a single value SHALL assign both lower and upper value as equal to that single value.”). This should be checked and translated manually as restrictions (when possible) afterwards.

Implementation choices for specific communities

* ShapeChange “Map entries” provide a flexible way to choose recommended names for properties and classes. This would enable one to reuse existing specialized classes and properties from external ontologies.
* GeologicUnitView contains mixed information from Both GeologicUnit and MappedFeature. A decision must be made to which entity the view must be associated (using the same URI as the GeologicUnit or MappedFeature )

### Meta-model issues (expressivity mismatches between OWL and UML)

* The placeholder attribute “any” becomes useless property in owl 🡪 delete it.
* Choice made to replace the “character string” data properties by object properties from GeoSciML Basic, borehole and other ontologies when possible (using the XPath mapping detailed in GeoSciML specification).

### Bugs and limitations in software (or things too hard to configure)

* Association classes must be handled differently: ShapeChange transforms an association class into separate class and properties. Thus, no link is created between the association class and the classes that are initially related by it in the UML. No rule is found in ShapeChange to handle that.

🡪 this must be defined afterwards with two properties: associationSource and associationTarget (exactly as in passing from conceptual model to a logical schema). As a solution, this could be locally defined as [association name]+”Source” and [association name]+”Target”. These two properties must have the right domain and range. The direct property between the source and the target automatically created by Shapechange must be deleted.

### Annotation practices

* Version the ontology: use owl:priorVersion and owl:versionInfo properties to describe the ontology, and owl:deprecatedClass and owl:deprecatedProperty annotation properties to specify the version status of a class or a property when deprecated.

# Outputs

## HY\_Features encoding example:

(ideally with live links to deployed content)

## Summary of choices made

## Configuration references