European Commission

ISA Action 1.17: A Reusable INSPIRE Reference Platform (ARE3NA)

Study on RDF and PIDs for INSPIRE

Deliverable D.TD.03

Guidelines on methodologies for the creation of RDF vocabularies representing the INSPIRE data models and the transformation of INSPIRE data into RDF

Diederik Tirry

Danny Vandenbroucke

Robin S. Smith

Michael Lutz

Joint Research Centre

This publication is a Deliverable of Action 1.17 of the Interoperability Solutions for European Public Administrations (ISA) Programme of the European Union, A Reusable INSPIRE Reference Platform (ARE3NA), managed by the Joint Research Centre, the European Commission's in-house science service.

Bibliographic Information:

Tirry D, Vandenbroucke D., Smith R.S. and Lutz, M. Study on RDF and PIDs for INSPIRE (D.TD.03) Guidelines on methodologies for the creation of RDF vocabularies representing the INSPIRE data models and the transformation of INSPIRE data into RDF. European Commission; 2015. JRC96797

Preface

The current default encoding recommended in the INSPIRE Technical Guidelines for most INSPIRE themes is based on GML. The GML schemas are automatically generated from the UML data models based on explicitly defined encoding rules. While GML is widely known within the geospatial information domain, many e-government applications and tools start to adopt 'Linked Data' to publish their data using RDF. This document describes a first attempt to establish guidelines on methodologies for the creation of RDF vocabularies representing the INSPIRE data models and the transformation of INSPIRE data into RDF. It should be emphasized, however, that this study is not an attempt to replace GML as the current default encoding in INSPIRE.

In this study an empirical approach was followed and resulted in an overview of transformation challenges that are described in detail. An executive summary is included to provide an overview of the methodology in general, and targets those that are not familiar with the INSPIRE process and/or the Resource Description Framework (RDF). People with limited or no expertise in these knowledge domains are also encouraged to read another outcome of this study i.e. a state-of-play report that collects shared evidence about the current status in Europe of linked (geospatial) data related to INSPIRE [1].

The document is organized as follows: first, an introduction is given to the overall context, motivation and scope of this study. In chapter 2 it is explained how a pilot experiment has been set up and how the results were assessed. Chapter 3 and 4 compile, consolidate and synthesize the output and major findings of the experiment, divided into 2 distinct but interrelated aspects: (1) the transformation of INSPIRE conceptual models, and (2) the transformation of INSPIRE data into RDF. To clarify possible approaches and solutions, examples are inserted using Turtle (Terse RDF Triple Language). Next, in chapter 5 an overview is provided of tools that are potentially useful for schema and instance transformation. Chapter 6 lists a number of potential implications to other INSPIRE components, when transforming INSPIRE models and data into RDF. Chapter 7 puts forward the main conclusions of this experiment and further steps to be taken for transforming INSPIRE into RDF. The generated RDF vocabularies for each of the investigated INSPIRE Annex Themes are published at https://ies-svn.jrc.ec.europa.eu/attachments/download/483/ARE3NA_D.TD.03_AnnexB_Vocabularies.zip. More details on the methodology can be found in the individual reports of the experts ([2], [3], [4], [5], [6] and [7]).

Executive summary

Scope

The INSPIRE Directive –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– entered into force in May 2007. The European Commission initiated the INSPIRE initiative to enhance the sharing of spatial data and services between public authorities in Europe and between the Member States and the European Institutions in particular. INSPIRE is focusing on addressing the interoperability of geospatial data sets and services through the creation of application schemas (using UML) and geospatial encodings mechanisms (using GML, GeoTIFF and other formats), for the exchange of data related to one of the 34 spatial data themes defined in the INSPIRE Directive. At the same time, e-Government applications and tools start to use the Linked Data paradigm, based on Semantic Web languages and technologies, such as the Resource Description Framework (RDF) of W3C. This document provides a first set of guidelines and recommendations on methodologies for the creation of RDF vocabularies representing the INSPIRE data models and the transformation of INSPIRE data into RDF.

Empirical approach

This study sought to have an in-depth exploration of the application of RDF for INSPIRE by setting up a pilot experiment with three domain experts, using different methodologies and tools for the requested tasks. The aim of the pilot experiment was to compare the different methodologies taken by the experts and to try to establish a common methodology for the transformation of INSPIRE UML models into RDF. The experts were requested to propose a methodology based on their experience and knowledge of the topic and to test their methodology by applying it to three INSPIRE Annex Themes that have cross-sector relevance, i.e. they are not only relevant to environmental policies. An analysis framework based on a modified Delphi research method was set up for assessing the different methods. The analysis of the outcome of the experiment comprises 2 distinct but interrelated aspects:

- The transformation of INSPIRE conceptual models, and
- The transformation of INSPIRE data into RDF.

Transformation of INSPIRE conceptual models

From the beginning of the study it became clear that future guidelines first should elucidate the general context before proposing a set of specific rules for the conversion of INSPIRE UML models and data into RDF. Two major factors constitute the general context for making INSPIRE data available as Linked Data: the scope of the transformation and existing standards and specifications that are relevant to the transformation of the conceptual models.

Concerning the scope of the transformation, it was agreed that common RDF vocabularies and guidelines are developed with the purpose of building bridges between INSPIRE's developments to share geospatial information in support of environment policy to potential areas of reuse of its data and reference materials in e-government. It is not intended that these vocabularies will be used to infer new knowledge through reasoning or to validate against RDF data. An integral part of the modelling of common RDF vocabularies is also to find consensus among stakeholders on the use of relevant technical standards and existing vocabulary specifications.

For the analysis of the outcome of the experiment the conversion rules proposed by ISO/DIS19150-2 have been used as a starting point. These rules are described and alternative approaches are

proposed in an attempt to ameliorate the observed issues when applying these rules in the experiment.

The transformation of INSPIRE data into RDF

Testing and developing guidance on how to transform INSPIRE feature data into RDF was beyond the scope of this study. An important aspect of instance transformation, however, is the different modelling discourse that is used in the GI and Linked Data community that has to be aligned with each other. There is, for example, a common understanding that Linked Data requires to speak clearly and distinctly about the subject, which is either the spatial object (the abstraction) or the real-world phenomenon it abstracts. Good guidance and examples are required to demonstrate how feature instances should be represented in RDF as this information is not immediately accessible from the RDF vocabularies and the RDF vocabularies cannot be used for validation of the instance documents.

Conclusions and further steps

The exploration of the transformation of INSPIRE UML models to RDF vocabularies, which has been done on the basis of an experiment, clearly demonstrated that a number of aspects need to be taken into consideration when defining a set of conversion rules. The results of the experiment also gave insight into the challenges of transforming INSPIRE data in RDF. One of the major outcomes of the pilot experiment is that, due to the amount of remaining open issues, potential obstacles and implementation options, a common methodology cannot be elaborated yet. It is important to recognize that the development of RDF vocabularies for INSPIRE is still work-in-progress, and that it requires broader review and discussion as well as testing in applications. Moreover, any generated RDF vocabulary will require reviewing and additional edits. On the other hand, there is an emerging need to expose INSPIRE data as RDF in a short term so that other communities can refer their data to INSPIRE. Therefore, in order to provide short term guidance, the requirement for an in-depth approach – which is a time-consuming process- needs to be balanced with the need for pragmatic solutions that can be offered to the interested community.

Table of contents

List of figures

List of tables

List of abbreviations

ADMS	Asset Description Metadata Schema
API	Application programming interface
CWA	Closed-World Assumption
DC	Dublin Core
DCMI	Dublin Core Metadata Initiative
DCT	Dublin Core Terms
EC	European Commission
FOAF	Friend of a Friend
GCM	INSPIRE Generic Conceptual Model
GFM	General Feature Model
GI	Geographic Information
GIS	Geographic Information System
GML	Geography Markup Language
HTTP	HyperText Transfer Protocol
ISA	Interoperability Solutions for European Public Administrations
JRC	Joint Research Centre
LD	Linked Data
LOD	Linked Open Data
MIG	INSPIRE Maintenance and Implementation Group
ORG	Organization ontology
OSS	Open source software
OWA	Open-World Assumption
OWL	Web Ontology Language
PID	Persistent identifier
PROV	W3C's Provenance family of documents
RDF	Resource Description Framework
SDI	Spatial Data Infrastructures
SKOS	Simple Knowledge Organization System
SPARQL	SPARQL Protocol and RDF Query Language

UML	Unified Modeling Language
URI	Uniform resource identifier
URL	Uniform resource locator
URN	Uniform resource name
VoiD	Vocabulary of Interlinked Datasets
XSLT	Extensible Stylesheet Language Transformations

List of terms and definitions

Feature	abstraction of real world phenomena [ISO 19101] NOTE The term "(geographic) feature" as used in the ISO 19100 series of International Standards, in other specifications like IHO S-57, and in this document is synonymously with spatial object as used in this document. Unfortunately "spatial object" is also used in the ISO 19100 series of International Standards, however with a different meaning: a spatial object in the ISO 19100 series is a spatial geometry or topology.
HTTP URI	HTTP URIs, in the web architecture, have been used to denote documents -- "web pages" informally, or "information resources" more formally. However, with the growth of the Semantic Web, which uses URIs to denote anything at all, the urge to use and practice of using HTTP URIs for arbitrary things grew steadily. The W3C Technical Architecture group eventually decided to resolve the architectural problem that if an HTTP response code of 200 (a successful retrieval) was given, that indicated that the URI indeed was for an information resource, but with no such response, or with a different code, no such assumption could be made. This compromise resolved the issue, leaving a consistent architecture. http://www.w3.org/DesignIssues/HTTP-URI.html
Linked Data	Linked Data, an integral and essential part of the Semantic Web, is a method for exposing, publishing and sharing structured data using URIs and RDF, with the idea of interlinking different datasets for making them more useful. http://linkeddata.org
Ontology	An ontology is a formal specification of a shared conceptualization. In the context of computer and information sciences, an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. Ontologies are considered one of the pillars of the Semantic Web, although there exist many definitions. http://semanticweb.org/wiki/Ontology http://www-ksl.stanford.edu/kst/what-is-an-ontology.html
OWL	Web Ontology Language The W3C Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be exploited by computer programs, e.g., to verify the consistency of that knowledge or to make implicit knowledge explicit. OWL documents, known as ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies. OWL is part of the W3C's Semantic Web technology stack, which includes RDF, RDFS, SPARQL, etc. http://www.w3.org/2001/sw/wiki/OWL
PID	Persistence Identifier An identifier is a unique identification code that is applied to "something", so that the "something" can be unambiguously referenced. For example, a catalogue number is an

	identifier for a particular specimen, and an ISBN number is an identifier for a particular book. It is an overarching term and could take various forms such as persistent identifier systems include: Archival Resource Keys (ARKs), Digital Object Identifiers (DOIs), Persistent Uniform Resource Locators (PURLs), Uniform Resource Names (URNs).
RDF	Resource Description Framework RDF is a standard model for data interchange on the Web. RDF has features that facilitate data merging even if the underlying schemas differ, and it specifically supports the evolution of schemas over time without requiring all the data consumers to be changed. RDF extends the linking structure of the Web to use URIs to name the relationship between things as well as the two ends of the link (this is usually referred to as a "triple"). Using this simple model, it allows structured and semi-structured data to be mixed, exposed, and shared across different applications. http://www.w3.org/2001/sw/wiki/RDF
RDFS	RDF Vocabulary Description Language 1.0: RDF Schema RDFS is a general-purpose language for representing simple RDF vocabularies on the Web. Other vocabulary definition technologies, like OWL or SKOS, build on RDFS and provide language for defining structured, Web-based ontologies which enable richer integration and interoperability of data among descriptive communities. http://www.w3.org/2001/sw/wiki/RDFS
Spatial object	See 'Feature'
Vocabulary	On the Semantic Web, vocabularies define the concepts and relationships (also referred to as "terms") used to describe and represent an area of concern. Vocabularies are used to classify the terms that can be used in a particular application, characterize possible relationships, and define possible constraints on using those terms. In practice, vocabularies can be very complex (with several thousands of terms) or very simple (describing one or two concepts only). There is no clear division between what is referred to as "vocabularies" and "ontologies". The trend is to use the word "ontology" for more complex, and possibly quite formal collection of terms, whereas "vocabulary" is used when such strict formalism is not necessarily used or only in a very loose sense. Vocabularies are the basic building blocks for inference techniques on the Semantic Web. Core vocabularies refer to the vocabularies developed in the context of the ISA programme: person, business, location and public service http://www.w3.org/standards/semanticweb/ontology
URI	In computing, a uniform resource identifier (URI) is a string of characters used to identify a name of a web resource. Such identification enables interaction with representations of the web resource over a network, typically the World Wide Web, using specific protocols. Schemes specifying a concrete syntax and associated protocols define each URI.
URL	Uniform Resource Locator Is a specific character string that constitutes a reference to a resource on the web. It is the global address of documents and other resources on the World Wide Web. http://www.webopedia.com/TERM/U/URL.html
URN	Uniform Resource Name Uniform Resource Names (URNs) are intended to serve as persistent, location-independent, resource identifiers. http://datatracker.ietf.org/wg/urn/charter/

1. Introduction

1.1. Context

This deliverable has been prepared in the context of the INSPIRE Directive [8] and the ISA Action 1.17 [9] which aims to create a platform to support the reuse of location/geospatial data, metadata and services to support cross-border and cross-sector interoperability tasks in public administrations across the European Union (EU).

The INSPIRE Directive – 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 – entered into force in May 2007. The European Commission initiated the INSPIRE initiative to enhance the sharing of spatial data and services between public authorities in Europe and between the Member States and the European Institutions in particular. The EU Member States transposed the Directive into national legislation between May 2007 and May 2009. Key components are specified through technical Implementing Rules described in Commission Regulations. In addition to the Implementing Rules, non-binding Technical Guidance documents, which are based on international standards (ISO/TC 211, CEN/TC 287 and OGC), describe how data providers might implement the Implementing Rules. However, the different approach of implementing the standards, the regular evolution of standards and challenges in coordinating changes between standards, alongside varying choices in the technologies being adopted are creating interoperability challenges.

In order to address these interoperability challenges, the European Commission's (EC) Joint Research Centre (JRC), as part of the Interoperability Solutions for European Public Administrations Programme [10], has established a *Reusable INSPIRE Reference Platform* [9].

1.2. Motivation

INSPIRE is focusing on addressing the interoperability of geospatial data sets and services through the creation of application schemas (using UML) and geospatial encodings mechanisms (using GML, GeoTIFF and other formats), for the exchange of data related to one of the 34 spatial data themes defined in the INSPIRE Directive. At the same time, e-Government applications and tools start to use the Linked Data paradigm, based on Semantic Web languages and technologies, as the Resource Description Framework (RDF) of W3C. Several European projects and initiatives in Member States have created RDF vocabularies based on the conceptual INSPIRE data models or the corresponding XML implementation. Several approaches have been applied and several issues remain unanswered:

- **Lack of agreed rules or guidelines on how to create RDF vocabularies from the UML models**. In the context of INSPIRE, data models are developed on a conceptual level using the Unified Modeling Language (UML), and the default encoding for most INSPIRE themes is based on the Geography Markup Language (GML). However, RDF is currently widely used in EU Member States and EU projects to make available e-government applications and tools as Linked Data. EU Member States and EU projects are also creating RDF vocabularies for INSPIRE, but no agreed rules or guidelines exist on how this creation should be performed.
- **Lack of best practices and guidelines in the area of global and persistent identifiers (PIDs).** Though EU Member States have created governance structures, processes, rules/guidelines and tools to create, manage, maintain and use PIDs in their Spatial Data Infrastructures (SDIs) that INSPIRE is built upon, different approaches have been identified.

1.3. Scope

The current default encoding recommended in the INSPIRE Technical Guidelines for most INSPIRE themes is based on the GML. The GML schemas are automatically generated from the UML data models based on explicitly defined encoding rules [11]. While GML is widely known within the geospatial information domain, many e-government applications and tools are made available as Linked Data using RDF. This document provides a first set of guidelines and recommendations on methodologies for the creation of RDF vocabularies representing the INSPIRE data models and the transformation of INSPIRE data into RDF. Before this study, in the context of this project, the following activities were performed (Figure 1):

- Stakeholders and relevant literature identification
- State-of-play – the current landscape with regard to the transformation of INSPIRE data and data models to RDF

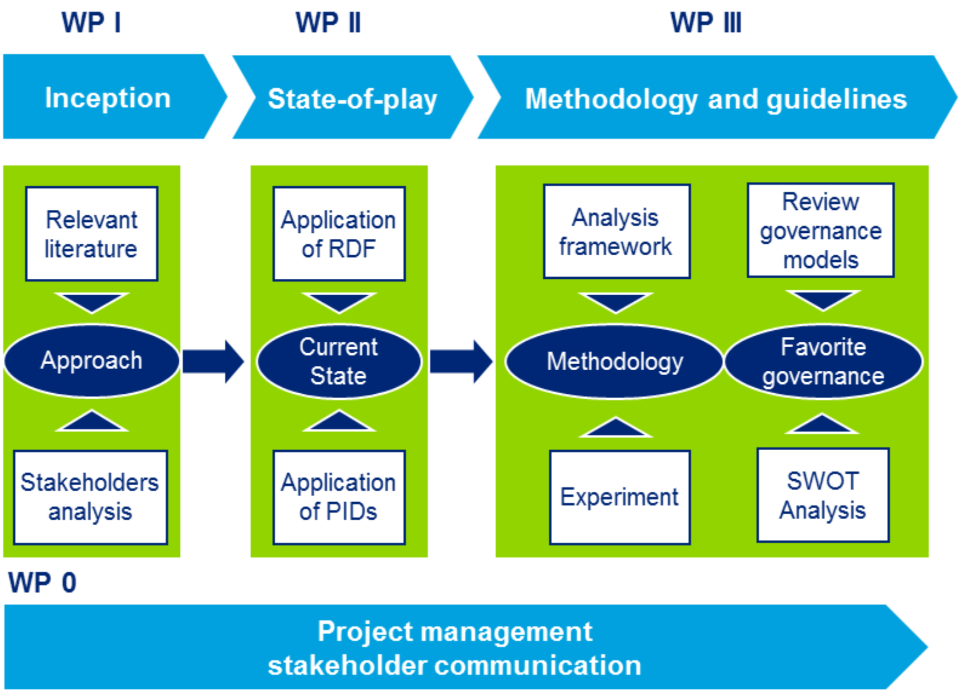


Figure 1: Scope of the study

The current deliverable is based on the observations collected during the previous phases and the outcome of an experiment which was designed to explore the challenges of transforming INSPIRE models to RDF vocabularies. The end-result consists of a set of conclusions and recommendations for the generation of RDF vocabularies from UML data models (schema level) as well as the transformation of data to RDF using these vocabularies (instance level).

These recommendations contribute towards the ultimate goal to publish agreed RDF vocabularies for INSPIRE together with guidelines for how to use them for creating instance RDF data.

Note that (once an agreement has been reached inside the INSPIRE community on the methodology to be followed) the generation of RDF vocabularies will have to be done only once (e.g. by JRC), whereas the transformation of instance data (according to these vocabularies) will need to be done for each data set. The recommendations on the schema-level mapping should thus be considered mainly as a basis for discussion on the most appropriate approach rather than as a methodology to be followed by each data provider.

2. An empirical and comparative approach

This chapter describes the methodology used for elaborating guidelines on the transformation of INSPIRE UML models into RDF vocabularies. First, it explains how a pilot experiment has been set up (objectives, participants, data models), and then details the approach for assessing the results of the experiment.

2.1. Pilot experiment

This study sought to have an in-depth exploration of the application of RDF for INSPIRE by setting up a pilot experiment with three domain experts, ideally using different methodologies and tools for the requested tasks. The aim of the pilot experiment was to compare the different methodologies taken by the experts and to try to establish a common methodology for the transformation of INSPIRE UML models into RDF. In cooperation with the JRC and the contractor, the experts were requested to:

- Propose a methodology to transform the conceptual INSPIRE UML data models into RDF vocabularies.
- Provide examples of the PIDs that the methodology needs, including PIDs for spatial objects, real world things, information items, services, etc.
- Apply the methodology for transforming UML data models into RDF vocabularies to three INSPIRE Annex Themes.
- Participate in a meeting with other experts to compare methodologies and the generated RDF vocabularies with the goal to derive a common INSPIRE RDF methodology.
- Participate in a webinar for the wider review and update of the preliminary results.
- Outline open issues or potential obstacles to the application of the proposed common methodology to other INSPIRE Annex Themes.
- Describe potential tools to be used for the transformation of INSPIRE-related source data (in their original format and schema) as well as INSPIRE-compliant data (in GML) to the generated RDF vocabularies.
- Outline any implications that using RDF as an encoding would have for other INSPIRE components.

The progress and results of the pilot experiment have been documented ([2], [3], [6], [7], [4], [5]), analysed and served as a basis for the creation of this document.

2.2. Design and organisation of the pilot experiment

The idea behind the experiment was to involve 3 domain experts that are familiar with the concepts of both the INSPIRE framework and Linked Data. Because of their involvement in on-going research or projects related to the topic of this study, following experts were selected to participate in the pilot experiment:

- Expert 1: Linda van den Brink (Geonovum – NL);
- Expert 2: Stuart Williams (Epimorphics Ltd – UK);
- Expert 3: Clemens Portele (Interactive Instruments GmbH – DE).

These experts were requested to propose a methodology based on their experience and knowledge of the topic and to apply their methodology to three INSPIRE Annex Themes [12]. The following criteria were used to select three INSPIRE themes for each expert:

1. Each expert should test a theme whose UML model is considered as simple, medium and complex.
2. There is at least one common INSPIRE theme to all experts.
3. The selected INSPIRE Annex Themes should cover most of the UML modelling patterns that are applied within the entire INSPIRE framework.
4. The selected INSPIRE Annex Themes should cover most of the base models and base types of the INSPIRE Generic Conceptual Model.
5. The selection should include INSPIRE Annex Themes that have cross-sector relevance, i.e. they are not only relevant to environmental policies.

It was decided to disregard INSPIRE Annex themes that either have an very complex UML model (e.g. Geology, Mineral Resources, Soil) or that are mainly focused on coverage data (e.g. Atmospheric Conditions, Orthoimagery). For themes that contain multiple application schemas, priority was given to the most important or relevant schemas.

Based on the abovementioned criteria each expert was assigned three INSPIRE Annex Themes. If the theme contains multiple application schemas, it has been indicated in the table below which ones were selected (Table 1).

Table 1: Selection of INSPIRE Annex Themes (and application schemas)

Complexity	Expert 1	Expert 2	Expert 3
Low	Buildings *(BuildingsBase, Buildings2D, Buildings3D)*	Area Management Zones *(Area Management Restriction and Regulation zones, Water Framework Directive)*	Land Cover *(LandCoverNomenclature, LandCoverRaster, LandCoverVector)*
Medium	Statistical units *(Statistical Units Base, Statistical Units Vector)*	Hydrography *(Hydro-base, Hydro-Network, Hydro-Physical Waters)*	Transport Networks *(Common Transport Elements, Road Transport Network)*
High	Environmental Monitoring Facilities		

2.3. Analysis framework

The approach for assessing the different methods proposed by the experts was based on a modified use of the Delphi research method [13]. Delphi is a structured group communication method for soliciting expert opinion about complex problems or novel ideas, through the use of a series of questionnaires and controlled feedback. The technique was designed to gather independent input from participants without working face-to-face. Often, the process is used to reach consensus among experts who may have different views and perspectives on a specific topic. One of the main reasons for choosing the Delphi method, is that it ensures an equal opportunity for the experts to contribute to the elaboration of a common methodology and that it avoids one of the experts imposing his/her view right from the start of the experiment.

As the method is primarily used for theory generation by a large number of experts (rather than testing and evaluation of methodologies), a modification of the standard technique was proposed and adopted for guiding and structuring the experiment. In this study the modified Delphi technique consisted of an iterative but short process of non-anonymous information collection, both individual and group-based. The different rounds in the pilot experiment are explained in the paragraphs below.

2.3.1. Step 1: Questionnaire

The modified Delphi technique began with the development of a questionnaire (Annex A) containing a series of open-ended questions related to the methodology for the development of common RDF vocabularies for INSPIRE. The basic idea behind the questionnaire is to acquire the first thoughts of every expert on the methodology. The questionnaire was then distributed to the three experts who were asked to list as many responses as possible (ideas, approaches, solutions, …). The responses were analysed, compiled and presented during a joined face-to-face meeting. Based on the input to the questionnaire a list of topics and issues that require further discussion during the meeting was drawn up.

2.3.2. Step 2: Face-to-face meeting

A joint face-to-face meeting (experts, contractor and JRC) was organised on 15 April 2014. The objective of the meeting was twofold:

- Introduce, compare and discuss the different approaches of the experts,
- Define a common baseline for the future findings and conclusions of each expert.

First, the experts were asked to give a brief presentation on research/work done related to this study and to provide a first set of ideas on how the transformation from INSPIRE UML models to RDF vocabularies could be approached. Next, the results of the questionnaire were presented to summarize the different angles taken by the experts. Then, the participants were invited to discuss topics and issues that have been prepared and listed based on previous input, including the availability of conversion rules, scoping of properties, external vocabularies, voidability, versioning, base types and models, code lists, modelling language, metadata, identifiers, instance data, etc…. Finally, agreements were concluded on the planning and next steps concerning the transformation of seven INSPIRE Annex themes to RDF.

2.3.3. Step 3: Revision of the methodology

Based on the discussions and outcome of the face-to-face meeting the experts were requested to revise their methodology and apply the revised methodology to the themes they were assigned to. The experts were asked to prepare a report describing:

- the methodology for transforming UML data models into RDF vocabularies,
- the tools used for the generation of RDF vocabularies, and
- the generated RDF vocabularies for the three themes.

2.3.4. Step 4: Discussion in a wider group

The outputs of the experts were compiled and consolidated to present them to a wider group of stakeholders. The preliminary results were presented during a webinar on 6 May 2014 to collect feedback on the approach and findings of this study and to have the contribution of the broader community for taking informed and well thought decisions on future steps to be taken.

2.3.5. Step 5: consolidation and reporting

Based on the output of the experts and the outcome of the webinar, a draft version of a common methodology outlining the key elements of a common INSPIRE RDF methodology (this document) was prepared and reviewed by the experts.

2.4. Documenting the results

The table below summarizes how each of the experts has approached the schema transformation from UML to RDF for the imposed themes. Besides the development of a methodology for schema conversion the experts were also asked to provide instance examples of that illustrate how feature instances should be represented in RDF as this information is not immediately accessible from the RDF vocabularies.

Table 2: Approach of schema transformation per expert

Expert	Themes	Tool	Remarks
Linda van den Brink	Buildings Statistical units Environmental Monitoring Facilities	ShapeChange (latest public version)	The ShapeChange version used in the experiment does not yet support ISO 19150-2 nor does it support changes proposed by Linda van den Brink, e.g. not using domain and range. The UML models and ShapeChange were extended to support the transformation to existing vocabularies by annotating the UML attributes that have a meaning that is standardized in some well-known vocabulary in RDF, with a link to their RDF counterpart. The annotation is recorded in UML via a tagged value. The transformation of the GCM was not included. Version of INSPIRE data models: r4618
Stuart Williams	Area Management Zones Hydrography Environmental Monitoring Facilities	Manual editing using TopBraid Composer	The transformation of the GCM was included. Version of INSPIRE data models: r4530
Clemens Portele	Land Cover Transport Networks Environmental Monitoring Facilities	ShapeChange (snapshot version)	The code of the snapshot version will be included in a new ShapeChange distribution in May 2014 under the GPL licence[1]. The transformation of the GCM was included. Version of INSPIRE data models: r4618

The analysis of the outcome of the experiment comprises 2 distinct but interrelated aspects:

1. The transformation of INSPIRE conceptual models, and
2. The transformation of INSPIRE data into RDF.

In the next chapters 3 and 4 we compile, consolidate and synthesize the output and major findings of the experts ([2], [3], [4], [5], [6] & [7]). For each aspect, the context of the concerned transformation is first described followed by an overview of possible conversion rules and implementation options respectively. The resulting RDF vocabularies for each of the investigated INSPIRE Annex Themes are

[1] See http://shapechange.net/targets/ontology/uml-rdfowl-19150-2/ for details.

published on the MIG collaboration platform[2]. Instance examples are provided in the individual reports of the experts ([2], [4] & [6]).

[2] https://ies-svn.jrc.ec.europa.eu/attachments/download/483/ARE3NA_D.TD.03_AnnexB_Vocabularies.zip

3. Transformation of INSPIRE conceptual models

3.1. General context for transformation of UML models

From the beginning of the study it became clear that future guidelines first should elucidate the general context before proposing a set of rules and options for the conversion of INSPIRE UML models and data into RDF. Two major factors constitute the general context for making INSPIRE data available as Linked Data.

3.1.1. Scope

Currently, INSPIRE provides a mechanism for standardizing and harmonizing spatial objects in 34 thematic domains. It is characterized by a service-based dissemination of (mostly) GML structured data that are modelled according to data specifications providing clear definitions of semantics in predefined domains and use cases. The structure and semantics are expressed in XML schema, and interoperability can be achieved by sharing and using a set of common GML application schemas. A downside of the INSPIRE approach is that it mainly addresses users that embrace the same service-oriented architecture and XML-schema-based encodings. However, INSPIRE data could be of use in other technological environments as well.

The intended use, therefore, of the RDF vocabularies is to publish INSPIRE data which might be linked to data from other technological environments or to which other data providers might want to refer their own data without a transition to another technological environment. Common RDF vocabularies and guidelines are therefore developed with the purpose of building bridges between INSPIRE's developments to share geospatial information in support of environment policy to potential areas of reuse of its data and reference materials in e-government. Guidelines on the creation of 'Linked Data' vocabularies and instance data shall support users of a specific technology family to establish cross-sector interoperability i.e. Semantic Web technologies to use spatial data from INSPIRE beyond the environmental domain. In this way it allows INSPIRE data to become part of an RDF-based web of data where it is integrated with other data and data models can be interrelated and harmonized – similar to the current support for the XML-based web of data.

It is not intended that these vocabularies are used to infer new knowledge through reasoning or to validate against RDF data. Validation, as it is quite commonly done with XML Schema in the traditional approach, cannot be compared to validation against RDF or OWL vocabularies. Whilst XML Schema is primarily used to define the structure (grammar) of data, RDFS/OWL is used for describing the knowledge model (semantics, meaning) of data. In other words, INSPIRE RDF vocabularies will primarily focus on describing, not prescribing, the entities within the INSPIRE domain of discourse. For these reasons, the intent is to use RDFS as much as possible for modelling the vocabulary and OWL only where necessary and appropriate.

3.1.2. Relevant technical standards and existing vocabulary specifications

An integral part of the modelling of common RDF vocabularies is an agreement on the use of relevant technical standards and existing vocabulary specifications that form the basis for harmonization and interoperability. The INSPIRE models import a lot of standardized information models coming from the ISO 19100 series. With the ISO/DIS 19150-2 [14] the ISO/TC 211 committee also provides a starting point that can be applied for generating RDF vocabularies. ISO 19150-2 is an ISO standard under development by ISO/TC 211. It gives the rules for developing geographic information ontologies in OWL. Clause 6 of the draft standard specifies general schema conversion rules for UML models conforming to ISO/DIS 19103 [15]. Clause 7 of ISO/DIS 19150-2 adds rules for application schemas based on the General Feature Model defined by ISO/DIS 19109 [16]. Schema conversion rules how classes with stereotype <<featureType>> are represented in a RDF vocabulary consistent with ISO/DIS

19150-2 can be inferred. Annex D of the draft standard includes an OWL ontology for the purpose of ISO geographic information ontologies.

The schema conversion rules specified by ISO/DIS 19150-2 make an attempt to represent most of the information in the UML model in the RDF vocabulary in some way. However, many of the rules specified by the draft standard may not be applicable for all use cases and often results in RDF vocabularies that show strongly the UML roots and do not really reflect common practice in the linked open data world. It is also important to understand that ISO/DIS 19150-2 is not finalized and technical comments have been submitted to ISO/TC 211 as part of the DIS vote. Some of the comments overlap with issues raised in the pilot experiment, but many do not. At this stage in the ISO process it is not foreseen to raise any new comments, so it is likely that RDF vocabularies for the INSPIRE application schemas would not conform to the final version of ISO 19150-2, if the conversion rules would be taken as starting point. Nevertheless, INSPIRE should take into account to consult and further discuss the outcome of the DIS comments on ISO 19150-2.

In the Linked Data community it is considered good practice to reuse well-defined and properly maintained vocabularies (e.g. FOAF, PROV, DC, DCT, RDF Data Cube, VoID, ORG, etc…) as this improves interoperability. For several feature attributes and classes in INSPIRE application schemas, commonly used properties and classes from existing RDF vocabularies are deemed appropriate for reuse. Whenever the semantics of such properties matches that of a feature attribute, the existing property should be used instead. The same applies for classes. This requires further review to ensure that the use of items from other vocabulary specifications is appropriate and can be specified in either a general or a theme-specific encoding rule.

3.1.2.1. Conversion rules for transforming INSPIRE conceptual models

For the analysis of the outcome of the experiment the conversion rules proposed by ISO/DIS19150-2 have been used as a starting point. This section describes the schema conversion rules that can be inferred from ISO/DIS19150-2 and discusses alternative approaches in an attempt to ameliorate the observed issues.

3.1.3. RDF namespaces

ISO/DIS 19150-2 converts UML packages to RDF namespaces. This does not recognize that the application schema represents a more natural namespace independent of the package structure within the application schema. However, the UML-to-OWL conversion rules in ISO 19150-2 apply to the models of the ISO/TC 211 standards and not to application schemas in the first place. In Clause 7 also the application schema is used as the basis for the RDF namespaces.

In addition, ISO/DIS 19150-2 uses a normalized version of the package name in the URI. As this might become quite long, a different convention based on the shorter application schema codes may be more appropriate, which is also aligned with the XML namespaces of the INSPIRE application schemas.

Examples of the adapted convention is:

```
http://example.com/ont/inspire/{app-schema-code}#
```

Final versions of the INSPIRE RDF vocabularies might use the following namespaces:

```
http://inspire.ec.europa.eu/ont/{app-schema-code}#
```

Recommendation 1

The RDF namespaces should be aligned with the XML namespaces of the INSPIRE application schemas instead of normalized versions of the package name as in ISO/DIS 19150-2.

3.1.4. Annotations

In INSPIRE application schemas the following information is (or should be) available for each model element:

- The so-called "language independent name" (the classifier, attribute or role name): According ISO/DIS 19150/2 this can be represented by a skos:notation element using xsd:NCName as the type.
- The human-readable name (if not available, the "language independent name" is used): According ISO/DIS 19150/2 this can be represented by a skos:prefLabel element.
- The definition: According ISO/DIS 19150/20, this can be represented by a skos:definition element.
- The description, if applicable: ISO/DIS 19150/2 does not specify a conversion rule for the description of classes and properties. Optionally, this can be represented by a skos:scopeNote element. It should be noted that also existing properties from the RDFS (e.g. rdfs:label) or DC (dc:description) vocabularies are suitable for making annotations.

```
lcv:LandCoverDataset  a  owl:Class ;
        skos:notation    "LandCoverDataset"^^xsd:NCName ;
        skos:prefLabel   "Land Cover Data set"@en ;
        skos:definition  "A vector representation for Land Cover data."@en ;
        skos:scopeNote   "This representation allows Land Cover data being supported by a 
vector geometry."@en .
```

Figure 2: Example of RDF vocabulary annotations

Recommendation 2

It should be further discussed if the SKOS properties proposed by ISO/DIS 19150-2 are the most appropriate properties for making annotations to INSPIRE RDF properties and classes

3.1.5. Property names, domains and ranges

In UML, properties (attributes and association roles) are scoped to the UML Classes in which they are defined and inherited by subclasses thereof. By 'scoped' we mean that the definition of a named property is determined by the UML Class with which it is associated. Two independent UML classes may each bear attributes with the same name, e.g. 'label' and the purpose or significance of the attribute may be identical, however, strictly, because the two classes are independent, they are two different UML attributes. Although in this example common significance or purpose has been attributed to these two properties, in general UML attributes/association-roles on independent UML classes whose names are identical may have completely different definitions and purpose.

In contrast, RDF defines properties as first class entities that can exist independently of the classes that use them. They may be 'bound' to RDF/OWL classes simply through the use of 'rdfs:domain' statements or more complexly (in OWL) through the use of property restrictions that can impose cardinality restrictions on the use of the property. It is also possible to specialise the range of a property used in conjunction with instances of a given RDF/OWL class - although such specialisations may only narrow rather than extend the range of a property.

Another important difference between UML and RDF/OWL is the presumption being made. In UML class models we work under a Closed-World Assumption (CWA): all statements that have not been mentioned explicitly are false. In contrast, OWL uses an Open-World Assumption (OWA) where missing information is treated as unknown. These different semantics make it necessary to add various restrictions to the ontology during the transformation process from a UML model to an OWL ontology to preserve the original semantics of the model.

Because of the central role of properties – more so than classes – in comprehending RDF statements one of the best practices when publishing Linked Data is to reuse and share existing terminology rather than 're-invent' [17]. In RDF the drive toward reuse is encouraged by the use of open property domains, and for common purpose by more narrowly defined property ranges - though as general as is reasonably possible. So for example, "skos:prefLabel", "skos:altLabel" and "rdfs:label" are widely used (annotation) properties for labelling an entity.

The most important differences between UML and RDF/OWL property concepts have been summarized in Table 3.

Table 3: Differences between UML and RDF/OWL in modelling properties

UML	Linked Data
• Closed-world Assumption • UML properties (attribute and associations or more specifically association roles) are scoped to the UML Classes in which they are defined and inherited by subclasses thereof • Properties cannot exist independently	• Open-world Assumption • RDF properties are first class entities that exist independently from classes • Best practice to reuse and share existing terminology ("skos:prefLabel", "skos:altLabel" and "rdfs:label") • The use of domain and range statements provide additional information on the scope (rdfs:domain) and set of acceptable values (rdfs:range) of properties.

ISO/DIS 19150-2 includes the classifier name in property names, if the same property name is used for multiple properties in the RDF namespace. The use of rdfs:range is mandatory, whereas rdfs:domain may be used to specify the domain for properties, but there is no guidance when the domain should be defined and when not.

The preference for less context in the RDF/OWL properties seems appropriate. However, it is recommended to go one step further and try to consolidate properties with conflicting names. This can only be done automatically to a limited extent.

Recommendation 3

The definition of conversion rules for processing identically named and spirited UML attributes/association roles into shared RDF properties across UML classes within the same application schema and in different application schema needs further study and experiment. Given the OWA in RDF/OWL, it should be further discussed to what extent the original semantics (in particular, any restrictions) of the UML model should be preserved.

3.1.6. Multiplicity

In the application schemas in UML, multiplicity specifies the allowable cardinalities for the instantiation of an element. It is expressed by the pair of lower and upper bounds of the number of times the element can be instantiated. ISO/DIS 19150-2 converts UML multiplicity to restrictions on a data or object property using an owl:Restriction declaration in combination with cardinality specifications. Cardinality specifications are restricted to the use of cardinalities only using owl:cardinality, owl:minCardinality, and owl:maxCardinality together with owl:allValuesFrom. In cases where allValuesFrom references the same class or datatype as the range of the property, this additional restriction adds no value as it duplicates the meaning of the range of the property, and therefore can be omitted. In other cases, it is included (e.g. Figure 3).

```
net:LinkSequence  a            owl:Class ;
        rdfs:subClassOf        iso19150-2:FeatureType , net:GeneralisedLink , geo:Feature ,
gfm:AnyFeature ;
        rdfs:subClassOf        [ a                   owl:Restriction ;
                                 owl:allValuesFrom   net:DirectedLink ;
                                 owl:onProperty      net:link
                               ] ;
        iso19150-2:isAbstract  true ;
        skos:definition        "A network element which represents a continuous path in the
network without any branches. The element has a defined beginning and end and every position
on the link sequence is identifiable with one single parameter such as length."@en ;
        skos:notation          "LinkSequence"^^xsd:NCName ;
        skos:prefLabel         "LinkSequence"@en ;
        skos:scopeNote         "EXAMPLE A link sequence may represent a route."@en .
```

Figure 3: allValuesFrom included for property net:link in class net:LinkSequence

However, information on multiplicity is on purpose not included in the INSPIRE regulations. Cardinality restrictions also offer little value for syntactic validation in the context of RDF. In only very few cases, they are valuable at a semantic level e.g. a person has exactly two parents. Therefore, it is proposed to suppress cardinality restrictions in the RDF vocabularies.

Recommendation 4

Preserving cardinality constraints expressed in UML into derived RDFS/OWL has limited use from the point of view of validating spatial objects, because in RDFS/OWL they are not an expression of syntactic constraints and the open-world-assumption applies. However, it requires further consultation and examination if their preservation does usefully express modelling intent both to data publishers and data consumers.

3.1.7. Voidable properties

Void is a concept defined by ISO/IEC 11404 as "an object whose presence is syntactically or semantically required, but carries no information in a given instance." The concept is used in INSPIRE and allows to state explicitly that:

1. a property of a spatial object, for example the name of a road, is not known and distinguishes this from
2. stating explicitly that a road is known to have no name.

Therefore, thanks to the voidable concept, the INSPIRE application schemas, although generally based on the closed-world assumption, support unknown facts.

ISO/DIS 19150 does not consider the concept of voidable properties. Such properties present a certain amount of difficulty to RDF under the open-world assumption. Just because the value of a property may not be given does not mean that there is no value for that property that could be given elsewhere. In RDF not stating a property is equivalent to setting the property to nil in the GML encoding. As a result there is no need to add a schema conversion rule for <<voidable>>. Note that this would no longer be the case, if cardinality restrictions would be represented as the minimum cardinality specified in the application schema. Some voidable properties have a minimum cardinality of 1, i.e. that always have to be present.

RDF has no proper mechanism to state that a road is known to have no name. Without a natural way to express such facts in an INSPIRE RDF representation, the RDF representation will state that no road name is known. While this is a loss of information, it is probably not essential for many applications.

INSPIRE also supports stating the reason why something is not known (unknown, unpopulated or withheld). However, this information is optional and in most cases of limited practical value. If the need arises to explicitly state the void reason, an approach could rely on the creation of a code list for void reasons (a skos:ConceptScheme with instance gcm:voidReasonConceptScheme and a skos:Concept subclass with gcm:VoidReason). There are two possible approaches, both of which rely on the creation of a codelist for void reasons (a skos:ConceptScheme instance gcm:voidReasonConceptScheme and a skos:Concept subclass gcm:VoidReason).

1. On voidable attributes, specify an rdfs:range that is an owl:unionOf the properties 'natural' range and gcm:VoidReason.
2. Specify (1) voidable properties with their 'natural' rdfs:range and (2) add an additional property to associate a list of void property values with instance data, which state voided properties for that particular instance and the associated gcm:VoidReason.

The second approach includes the definition of an additional open-domained property for referencing the class gcm:VoidPropertyValueList, i.e. the property gcm:voidPropertyValueList.

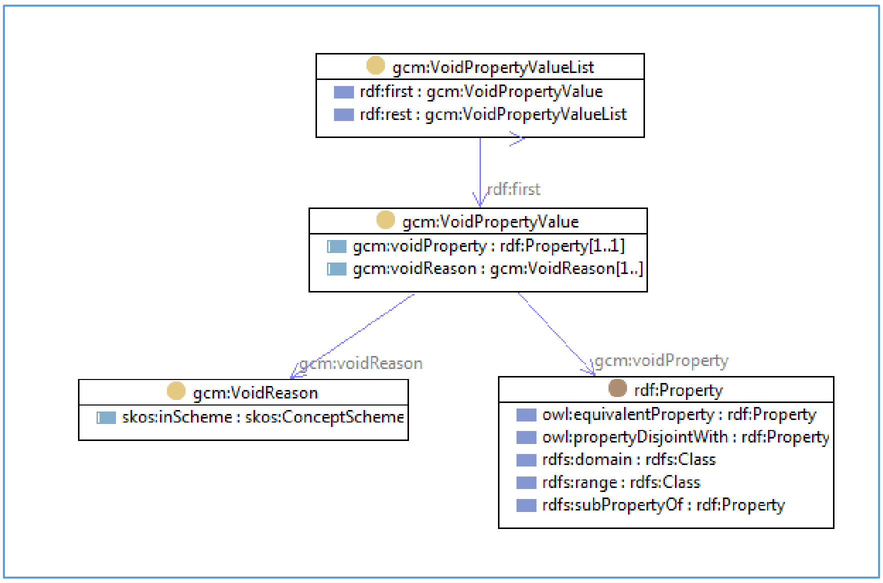


Figure 4: possible approach to model void reasons.

Recommendation 5
Because information on void reasons is optional and in most cases of limited practical value, it is not recommended to extend in RDF the 'natural' range of a voidable property with the set of available void reasons. An alternative approach would be to annotate instances with explicit information about properties that have been given void values and the reason for the void.

3.1.8. Lifecycle and other metadata properties

As it will be discussed in section 4, most feature attributes and roles represent properties of the real-world phenomenon. According to the General Feature Model, it is common practice to model both properties that describe the real-world phenomenon and properties that describe the feature ("spatial object" in INSPIRE terminology) document, i.e. which are feature metadata, as feature properties.

In INSPIRE, most properties are properties that describe the real-world phenomenon. However, there are exceptions:

- Properties that represent life-cycle information (in particular, the beginLifespanVersion and endLifespanVersion attributes) are marked with the stereotype <<lifeCycleInfo>>.
- Properties that have a value type from ISO 19115 are often feature metadata. However, this is not always the case, in particular for CI types. An example is ProtectedSite.legalFoundationDocument with value type CI_Citation.
- There are also some properties that require a closer review to identify them as feature metadata. Examples are CadastralZoning.estimatedAccuracy with value type Length or CadastralZoning.originalMapScaleDenominator with value type Integer. These properties are not properties of the real-world phenomenon, but of the feature.

From the perspective of the RDF vocabularies there is no distinction between the two types of properties, because the rdfs:domain is not required by ISO/DIS 19150-2 (see 3.1.5). Nevertheless, it impacts how instances are represented in RDF as it is important in Linked Data and the Semantic Web to be clear about the subjects. In this case we have two subjects – the real-world phenomenon and the feature, and should distinguish the two. This is discussed in more detail in section 4.1.

It would therefore be useful, if there is an unambiguous specification, to define which properties are feature metadata and should be used in conjunction with the feature document as the subject. Following the <<lifeCycleInfo>>-stereotype approach, one option would be to use additional stereotypes for other metadata properties in the model, e.g. <<provInfo>>, <<qualityInfo>>, etc.

Recommendation 6

It requires further examination to review which UML attributes/association roles can be considered as feature metadata and should be used in conjunction with another subject i.e. the feature document instead of the real-world subject. Moreover, further analysis with a wide range of stakeholders is needed to decide on candidate classes and properties of existing RDF vocabularies.

3.1.9. Source and association properties

ISO/DIS 19150-2 mandates that dc:source is included on classes and properties. As the source is the same for the whole application schema, it reduces readability and increases the size of the RDF vocabularies to include a (redundant) dc:source property with all resources.

ISO/DIS 19150-2 also includes association names in object properties that were derived from association roles. In the INSPIRE application schemas association names are defined for documentation purposes.

Recommendation 7

As the application of dc:source, which is mandated by ISO/DIS 19150-2, adds no extra value, it is recommended to suppress this conversion rule.

3.1.10. Stereotypes and tagged values

ISO/DIS 19150-2 includes tagged values in the RDF vocabularies. In general, stereotypes and tagged values are UML-specific extension mechanisms. They should only be supported in schema conversion rules that map the values to native RDFS/OWL constructs and carry relevant information. For most tags there is little or no value. For example, there is no value in representing tagged values supporting the GML schema conversion rules in the RDF vocabulary.

Recommendation 8

Tagged values supporting the GML schema conversions rules should not be represented in the RDF vocabulary. The only exception is the <<FeatureType>> stereotype.

3.1.11. Code lists

The transformation of code lists into RDF vocabularies depends on the management of code lists and whether they are embedded in or separated from an application schema.

3.1.11.1. Separated from an application schema

In INSPIRE they are managed separately in the INSPIRE code list register [18], which supports representations in different formats (including RDF/XML [19]) through content negotiation. Therefore, it is inappropriate to include classes and SKOS concept schemes for code lists in the RDF vocabularies as it is described by the conversion rules of ISO/DIS 19150-2.

The RDF/XML representation provided by the INSPIRE registry models code lists using skos:ConceptScheme and code list values using skos:Concept[3]. Therefore, code list-valued attributes in the UML model should be defined as properties with the range skos:Concept.

Separation from an application schema allows for easier management because the code lists can be externally managed and changes (addition, supersession and deprecation of code lists and code list values) can be performed without altering the expression of the derived ontology. However, it also tends to lead to a proliferation of namespaces for code lists that are only used for instances of a given class or that have very few code values.

3.1.11.2. Embedded in an application schema

A second possibility is that the code list is embedded in an application schema, which could be the case if INSPIRE application schemas are extended by a Member State or a thematic community. In this scenario it is proposed to transform code lists and controlled vocabularies into SKOS concept schemes. All code list values are made members of the scheme using skos:inScheme. They are also made instances of a distinguished subclass of skos:Concept, which should be used to restrict the range of a property (see above). Typically, an open domained property is defined that can be used to make use of the code with an arbitrary entity.

Note that currently, work is underway in the MIG sub-group on registers and registries[4] on register federations, which could be used to manage additional code lists and code list values in national or thematic extensions outside the application schemas.

[3] In addition, since code lists can also be considered individual registers with code list values as their register items, they are also represented using dcat:Dataset (from the W3C Data Catalog vocabulary, http://www.w3.org/TR/2014/REC-vocab-dcat-20140116/) and prv:DataItem (from the Provenance Vocabulary, http://trdf.sourceforge.net/provenance/ns.html), respectively.

Recommendation 9

The SKOS representation for code lists (skos:ConceptScheme) and code list values (skos:Concept) provided by the INSPIRE registry should be used to represent INSPIRE code list in RDF. Instance data should refer to code list values through their http URIs.

3.1.12. Features and geometries

ISO/DIS 19150-2 specifies in subclause 7.5 that feature type classes should be sub-classes of gfm:AnyFeature (General Feature Model vocabulary) and iso19150-2:FeatureType (ISO 19150-2 vocabulary). Note that there is a comment on the DIS that questions the definition of iso19150-2:FeatureType.

Besides the ISO/DIS 19150-2 specifications there are also other vocabularies that specify an encoding for geometries. A recent, thorough overview of ways to encode geometry in RDF can be found in S. Anastasiou et al. [20]. Below, a short summary of relevant well-known vocabularies is presented. These differ in what they offer, ranging from only lat/long point geometries, point line and surface geometries, topology, to the possibility to use any coordinate reference system.

3.1.12.1. W3C Basic Geo

An early vocabulary for representing mapping/location data in RDF is W3C Basic Geo [21]. Basic Geo is a basic RDF vocabulary that provides the Semantic Web community with a namespace for representing lat(itude), long(itude) and other information about spatially-located things, using WGS84 as a reference datum. Although a W3C activity, this vocabulary is not a W3C standard nor is it in the process of becoming one. As is evident from the name, the vocabulary is very basic and has only classes for SpatialThing (similar to GML's Feature) and Point, and properties latitude, location, longitude, and altitude. It has no classes or properties for topology. The acceptance in the Linked Data community is high: it is used in both GeoNames and DBPedia, both in turn highly used data sets, and in web applications and services including Yahoo! Maps. W3C Basic Geo is too limited to use with INSPIRE data, because it only supports point geometries and no coordinate reference systems other than WGS84.

3.1.12.2. GeoRSS

GeoRSS can be used in RDF to make simple geographical assertions about objects. However, this is only applicable for GeoRSS Simple, not for its other available variant: the GML geometry encoding. GeoRSS Simple is a very basic format with point, line, box and polygon properties, and allows only WGS84. GeoRSS Simple is used in e.g. DBPedia and implemented in e.g. OpenLayers, GeoServer, Drupal, and the Google Maps API. Although it has support for more geometry types, it is still probably too limited for use with INSPIRE data because it only supports WGS84; also it is not a formalized standard.

3.1.12.3. NeoGeo

NeoGeo (http://geovocab.org/doc/neogeo/) has different namespaces for features (spatial: http://geovocab.org/spatial) and for geometry (http://geovocab.org/geometry). Geometries can have any format based on HTTP content negotiation. The vocabulary also has properties for modeling topological relations such as a city being part of a province, a country being externally connected to another, or disconnected, overlapping, containing another,or being a (non)tangential proper part. It does not mention how to reference coordinate reference systems. The content of geometries can be

[4] https://ies-svn.jrc.ec.europa.eu/projects/inspire-registry/wiki

represented in NeoGeo in different formats other than RDF, such as GML, KML or WKT. It then depends on these formats if it is possible to refer to the coordinate reference system that is used.

NeoGeo is the result of a community effort, a VoCamp, and not maintained by a standards organization. The latest version is from 2012.

3.1.12.4. GeoSPARQL

The OGC standard GeoSPARQL specifies with geo:Feature another class that is similar to gfm:AnyFeature. A sub-class property to geo:Feature has been added for all feature type classes.

```
net:Node  a                  owl:Class ;
        rdfs:subClassOf      net:NetworkElement , iso19150-2:FeatureType , geo:Feature ,
gfm:AnyFeature ;
        iso19150-2:isAbstract  true ;
```

Figure 5: Example of a sub-class property to GeoSPARQL geo:Feature

The OGC GeoSPARQL specification [22] allows two serializations for geometry, Well Known Text (WKT) and GML. WKT is a text based format for encoding geometries, defined in the Simple Features specification. The WKT option in GeoSPARQL allows only simple feature geometry types, but this is still a wide range of geometry types such as points, curves, surfaces and geometry collections. The GeoSPARQL vocabulary defines a property asWKT in which a geometry can be recorded as a text value. It is possible to use any coordinate reference system (CRS); a reference to the used CRS is recorded with the coordinates.

The GML option in GeoSPARQL allows all ISO 19107 spatial schema geometry types, which is a much wider range than the simple features allowed in WKT, including a lot of less commonly used types. The vocabulary defines a property asGML in which a geometry can be recorded as a GML literal, i.e. a geometry element from the GML schema can be embedded in the RDF. The GeoSPARQL asGML is offered as an option to record geometry in several other vocabularies, such as the Location Core Vocabulary .

3.1.12.5. Location Core

The Location Core Vocabulary was recently published as a W3C document [23]. The Location Core Vocabulary defines three classes, Location, Address, and Geometry, and several properties for describing places in terms of their name, address or geometry. The vocabulary allows for both a WKT encoding and the GeoSPARQL asGML option.

3.1.12.6. Feature representation for INSPIRE RDF

Besides the ISO/DIS 19150-2 specifications there are also other vocabularies currently used in the Linked Data community to represent location information. The GeoSPARQL and Location Core vocabulary seem at this moment valuable alternatives since these standards are actively managed (contrary to NeoGeo). Optionally, depending on the approach taken for creating instances (see also section 4.3), the feature class can also be made a subclass of gcm:Model whose role would be to link a feature to the real-world phenomenon that it abstracts. It may serve as a 'join' point between multiple features that abstract (aspects of) the same thing, whether published by the same publisher or from different publishers.

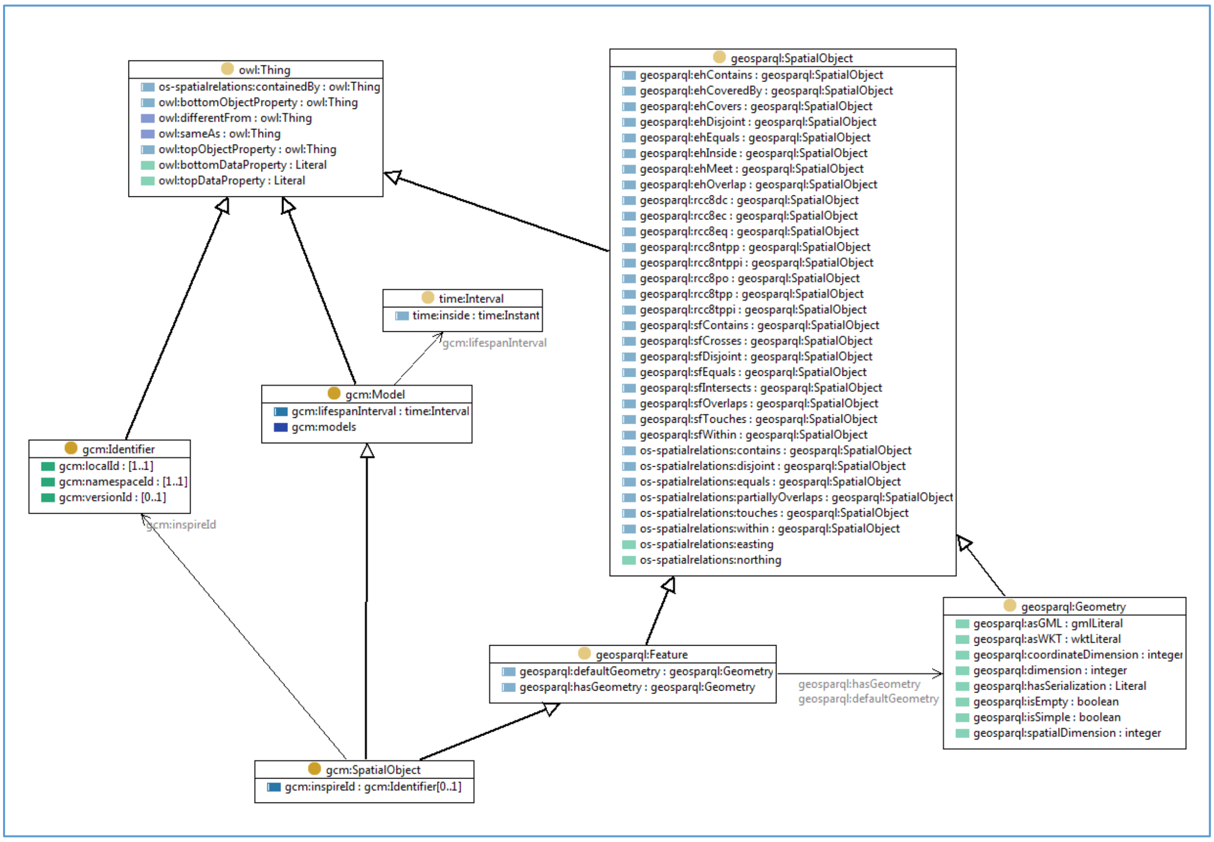


Figure 6: Modelling feature as subclass of a geoSPARQL feature and a Model class

However, at this stage no explicit recommendation for any of the vocabularies shall be made as it will to a large extend depend on future modifications and uptake by the community. It is worth mentioning that W3C and OGC are currently collaborating in a joint working group on Spatial Data on the Web[5], which aims to produce, among other things, an agreed spatial ontology conformant to the ISO 19107 abstract model and based on existing available ontologies such as GeoSPARQL, NeoGeo and the ISA Core Location vocabulary.

Recommendation 10

It is recommended to wait for the outcome of the joint OGC/W3C Spatial Data on the Web working group to make any decisions on the use of existing vocabularies for features and their geometries.

3.1.13. Foundation schemas

The INSPIRE application schemas that were part of the experiment make use of types from ISO 19103 [15], ISO 19107 [24], ISO 19108 [25], ISO 19111 [26], ISO 19115 [27], ISO 19123 [28] and ISO 19156 [29]. No sufficiently mature and tested RDF vocabularies exist for these ISO/TC 211 types, which is a problem for any attempt to represent INSPIRE data in RDF at this time[6].

5 See charter at http://www.w3.org/2015/spatial/charter

6 Simon Cox has developed some drafts RDF vocabularies that are available at http://def.seegrid.csiro.au/static/isotc211/ . These also differ significantly from RDF vocabularies that would be created using the ISO/DIS 19150-2 schema conversion rules.

For the purpose of the schema conversion of the INSPIRE application schemas, rdfs:Resource or owl:Class is proposed by default for all types without a known, more specific class.

It should be noted that some INSPIRE application schemas use types from foundation schemas that are not covered by the rules for application schemas. An example is the use of GM_Boundary in the Environmental Monitoring Facilities application schema.

3.1.13.1. ISO 19103

ISO/DIS 19150-2 specifies two conversions for types specified by ISO/DIS 19103 [15]. As there is no obvious value in using the iso19150-2:GCOLiteral option, the mapping to the XML Schema types is proposed. The schema conversion rules in ISO/DIS 19150-2 are incomplete and do not provide datatypes for several of the types in ISO/DIS 19103. Notably there is no support for the Measure types. To implement Measure there are at least 3 options:

1. Represent a Measure as a class and with an rdf:value and a 'unit of measure' property.
2. Represent the measure as a string value where both parts are encoded
3. Represent Measure by making use of the QUDT ontology, an ontology for quantities, units, dimensions and data types [30].

It is not yet clear which option is preferable in general.

```
sc:Measure
        a       owl:Class ;
      skos:notation  "Measure"^^xsd:NCName ;
      skos:prefLabel "measure"@en .

sc:uom
       a        owl:ObjectProperty ;
        rdfs:domain     sc:Measure ;
        rdfs:range      rdfs:Resource ;
        skos:notation   "uom"^^xsd:NCName ;
        skos:prefLabel  "unit of measure"@en .
```

Figure 7: Measure – Option 1: as Class

```
sc:Measure
        a       rdfs:Datatype ;
       skos:notation  "Measure"^^xsd:NCName ;
        skos:prefLabel "measure"@en ;
        skos:defnition "a text representation of a measure value. The decimal value is
followed by a space and the unit of the measure." .
```

Figure 8: Measure – Option 2: as datatype

ISO/DIS 19150-2 also provides no schema conversion rule for the types "LocalisedCharacterString", "PT_FreeText" or "URL" (and potentially other types, but those did not occur in the application schemas used in the pilot experiment). These types have been mapped to "xsd:string" and "xsd:anyURI" respectively.

3.1.13.2. ISO 19107

ISO/DIS 19150 references a yet-to-be-specified ontology for ISO 19107:2003 that should be used for the spatial properties. It is also doubtful, if that ontology, created using the ISO/DIS 19150-2 schema conversion rules, is valuable. For now, it seems advisable to use one of the RDF vocabularies for spatial geometries that are in use (see also section 3.1.12). For example, geo:Geometry from GeoSPARQL may be an interesting optinon in the generated RDF vocabularies as this supports both the Simple Feature representations that would be sufficient for many datasets (using asWKT or asGML) as well as as more complex GML geometries. In addition, GeoSPARQL geometries may use any coordinate reference system. A subproperty geo:hasDefaultGeometry allows for making a distinction between default and specific geometries.

3.1.13.3. Observations

The Observations UML package contains several application schemas: Observation References, Processes, Specialised Observations and Observable Properties. The usefulness of transforming these application schemas without a stable RDF vocabulary of ISO 19156 is difficult to judge.

The Semantic Sensors Network (SSN) ontology [31] may also be relevant, but as SSN is not yet standardized mappings to SSN were not defined. EnvironmentalMonitoringFacility could, for example, perhaps be mapped to ssn:Sensor or ssn:System. A mapping of the Observations Package to SSN would be useful.

3.1.13.4. External foundation schemas

Some of the INSPIRE application schemas are based on or inspired by external foundation schemas such as GeoScienceML, EarthResourceML, CityGML. In that case it needs to be checked with the corresponding community if an 'offical' vocabulary exists in RDF or OWL. For example, for CityGML there is no official vocabulary in RDF or OWL. An experimental one is available: Prof. Gilles Falquet from the University of Genève has done the translation mostly automatically. The OWL representation is available here: http://cui.unige.ch/isi/icle-wiki/ontologies. However, there is no information on the ontology available, so whether the ontology is complete, stable, has known issues, etc. is unknown. As the existing CityGML ontology needs to be further investigated, it cannot be considered to re-use this vocabulary. However, due to its importance, it is proposed to support the development of this ontology and promote its endorsement by a standards organization.

Recommendation 11

It was concluded that 'official' RDF vocabularies of the foundation schemas are needed in order to model INSPIRE's relationship with them. Therefore, it is recommended to take into account the disposition of the DIS comments of ISO/DIS 19150-2 and further discuss the use of foundation schemas that are managed by thematic communities.

3.1.14. Existing RDF vocabularies

It should be checked if an existing RDF vocabulary could be re-used for classes like RelatedParty, OfficialJournalInformation, LegislationCitation or Contact as well as address properties. These are all general and not primarily spatial concepts, so INSPIRE RDF vocabularies should avoid establishing their own resources and properties for such classes. For several feature attributes and classes in INSPIRE application schemas, commonly used properties and classes from existing RDF vocabularies can be reused. Whenever the semantics of such properties matches that of a feature attribute, the existing property should be used instead. The same applies for classes. This requires review for each of the INSPIRE application schemas to ensure that the use of items from other vocabularies is appropriate. A thorough review of the INSPIRE application schemas will likely identify additional candidates from vocabularies like FOAF, PROV, DC, DCT, RDF Data Cube, VoID, ORG, etc… The ISA core vocabularies may also provide some coverage as essentially these types are relevant for e-government.

Properties that are common to several application schemas (such as responsibleParty, representativePoint, etc…) can be removed from the theme-specific vocabulary and added to the base types vocabulary. It should be emphasized that the conversion to existing RDF vocabularies should not necessarily be a one-to-one conversion from one class or property to another. As a start, the following recommendations were proposed by the experts:

- Map general and non-spatial concepts of INSPIRE application schemas as much as possible to well-known and actively maintained existing vocabularies like FOAF, PROV, DC, DCT, RDF Data Cube, VoID, ORG, ISA Core…
- Map all properties with name "label" or "name" to "rdfs:label"
- Map all properties with a value type of "GeographicalName" to "rdfs:label" (a string with possible language tag). The INSPIRE type GeographicalName can represent more information about a name than rdfs:label (which is basically restricted to the spellings and the associated language of each spelling). Consequently, all the metadata about the name from the GeographicalName datatype is lost in the conversion. However, as this is very likely sufficient for most geographical names in datasets this seemed sufficient for the INSPIRE application schemas used in the experiment. Such simplifications need to be clearly explained in order for an RDF encoding rule to be compliant with the INSPIRE Implementing Rule. Alternatively, we can retain the structure of GeographicalName and not map the name property. The disadvantage of that is that in the RDF without an explicit mapping to rdfs:label (or something else like dct:title), it will not be clear what the meaning of the label property is.

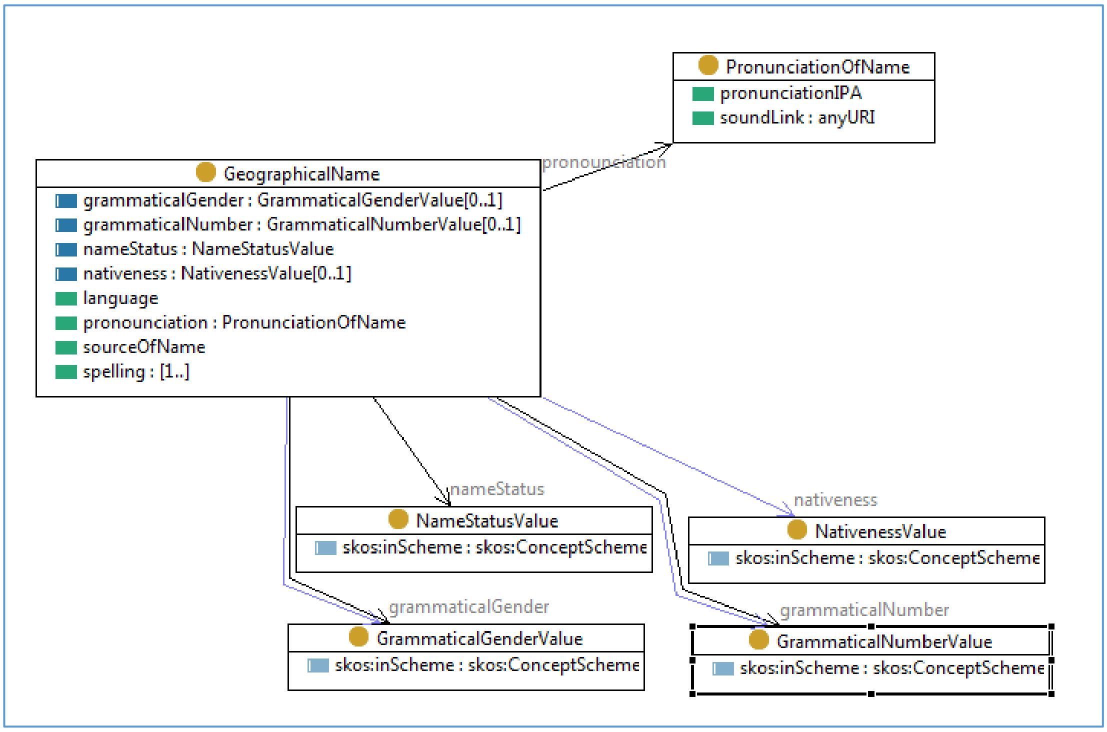


Figure 9: ‘full’ schema conversion of Geographical Names

- Map all properties with name "geometry" to "geo:hasGeometry" (see also section 3.1.12).
- Map all different date and datetime properties (including lifecycleinfo and validity information) to a common date/dateTime representation (dct:date or owl:DatatypeProperty with range xsd:dateTime). The properties prov:generatedAtTime / prov:invalidatedAtTime from PROV-O [32] seem to be good candidates for beginLifespanVersion / endLifespanVersion in the INSPIRE application schemas. The OWL Time ontology also gives support for both time intervals and instants.

Recommendation 12

The use/re-use of existing linked-data vocabularies such as FOAF, SKOS, DCAT, VoID, ORG, CUBE, SSN and others needs further study and experiment. Concrete rules can only be established on the basis of an assessment of the suitability of a given vocabulary to represent data required of a given INSPIRE schema. A softer approach may be to provide more general guidance about the adoption of external vocabularies, allowing practice to evolve, and to consolidate 'best' practice at a later stage.

3.1.15. Object identifiers

INSPIRE distinguishes two types of object identifiers in spatial data sets:

- The external object identifier or inspireId as a unique object identifier, published by the responsible body, which may be used by external applications to reference the spatial object.
- The thematic identifier as a descriptive unique object identifier applied to spatial objects in a defined information theme, e.g. a parcel code for parcel spatial objects in a cadastral theme.

As it will be pointed out in section 4, at least the inspireId property does not offer any value in an INSPIRE RDF representation as the resource URI represents the identifier (it is assumed that all resource URIs will be persistent identifiers). The inspireId attribute should therefore be ignored during the schema conversion process.

It may be discussed whether ThematicIdentifier needs to be represented as this is basically implemented by persistent URIs of the real-world phenomenon. Thematic identifiers are (typically local) identifiers of the real-world phenomenon. Therefore, it is the closest guess to the real-world subject and may be used in a URI to refer a real-world phenomenon. However, as pointed out in the next chapter, this should be verified for each of the application schemas.

Recommendation 13

The inspireId property should not be converted in a RDF property, as the resource URI already represents the identifier. It should be discussed whether ThematicIdentifier needs to be represented as this is basically implemented by persistent URIs of the real-world phenomenon.

3.1.16. UML modelling artifacts

Some of the UML modelling conventions are not or insufficiently considered by the ISO/DIS 19150-2 conversion rules. Others may result in awkward OWL constructions when strictly applying the schema conversion rules. The experiment has revealed that the conversion rules are inappropriate or incomplete for the modelling artefacts listed in Table 4.

Table 4: schema conversion problems related specific UML modelling artifacts

UML modelling artifact	Schema conversion problem
Compositions and aggregations	Two UML constructs – aggregation and composition - are encountered in several themes, but cannot be translated to OWL, as it does not feature predefined mereological relationship constructs in the knowledge representation ontology [33]. The constructs are supported in ISO/DIS 19150-2 but only as annotation information while the mereological semantics are lost.

UML modelling artifact	Schema conversion problem
Union data types	Conversion rules do not handle cases where • values are a mix of object or datatypes, or • the same value type is used by more than one option.
Association classes	Association classes are currently not supported by the schema conversion rules in ISO/DIS 19150-2
OCL constraints	The ISO/DIS 19150-2 schema conversion rule also maps constraints from the UML model. However, including OCL in the RDF vocabulary is questionable. Probably the most reasonable way would be to include only the documentation of a constraint.
Hierarchical code lists	The concept of hierarchical code lists is to be discussed in the relevant ISO TC 211 committees (notably on the revision of ISO 19103 and ISO 19109)

Recommendation 14

It requires further discussion with the corresponding ISO TC 211 committees to establish conversion rules for the modelling artefacts listed above.

3.1.17. Versioning

3.1.17.1. Known issues

Versioning is a known open issue in general as the base standards from ISO/TC 211 and OGC do not natively support this – at least not to the necessary level (WFS and GML have some support, but it quickly turns out to be insufficient as soon as features are related to each other). Also RDF/OWL does not natively support versioning. As a result, one typically has to "build" one's own framework on top of the existing standards and technology, which in practice is a problem (complex model and data that is often hard to understand, low performance due to non-native support in software products, no reuse of COTS tools, etc...).

INSPIRE has established rules on this topic (see Generic Conceptual Model [34], 9.7.2 and 14.5):

- The value of property that is an association role is the feature – not a specific version of a feature.
- The version information that is included in the inspireId property is not part of the feature identifier.

The Generic Conceptual Model concludes that "*current INSPIRE data specifications are only fully specified for spatial data sets that only publish the last version of a spatial object (valid or retired). If historic versions are maintained and provided, additional specification work is needed with regard to the consistency of the spatial objects at any time. This is particularly the case whenever associations are modelled as part of the application schemas where both roles are navigable as changes to one spatial object will result also in a new version of the associated spatial objects. This can result in a domino effect of new versions of the spatial objects in the data set.*"

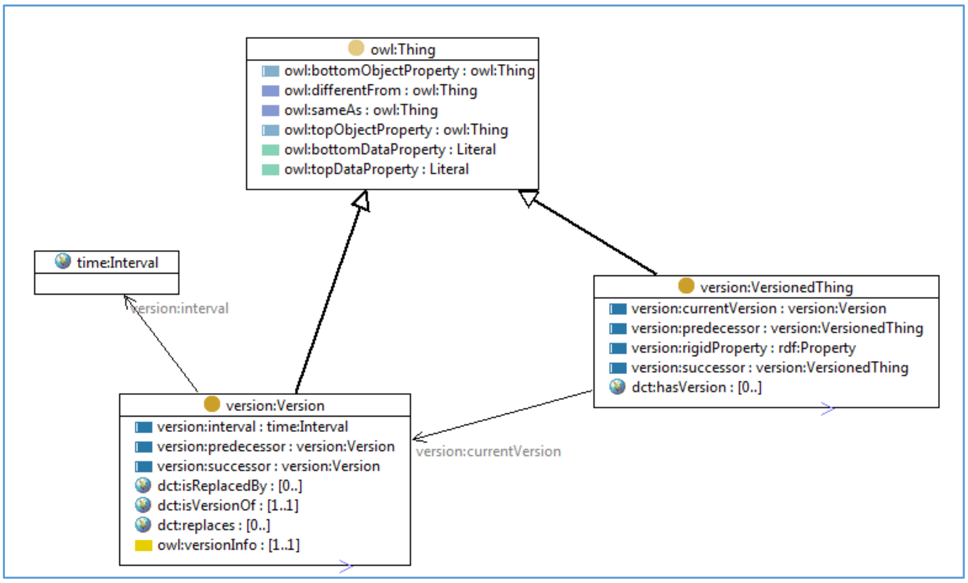
From this, it can be concluded that:

- A URI is needed for every real-world phenomenon and feature document, just as before and use these (and only these) for references.
- At least the properties of the real-world phenomenon that may change need a new subject.

If versioning would be explicitly required, with access to historic versions, the versioning model described by Chris Welty et al. [35] could be a possible approach.

3.1.17.2. Enduring spatial-objects with versioned temporal parts

In this approach a spatial-object is formulated as an enduring spatial-object which serves to collect versioned snapshots (a temporal-part or 'Version') that represent the state of the object at different points during its lifetime. The enduring object (a 'VersionedThing') represents the object over its entire lifetime and even beyond. In the proposed approach we also maintain a non-monotonic "currentVersion" link as shortcut link to the most recent 'snapshot' (a Version). 'VersionedThing' and 'Version' act as 'mixin' types added to spatial object nodes. 'VersionedThings' are somewhat vestigial and should only carry 'rigid' properties, properties that are invariant across the different spatial-object versions. The diagrams below illustrate a small versioning vocabulary that was created and used for a project to publish water quality information as Linked Data.



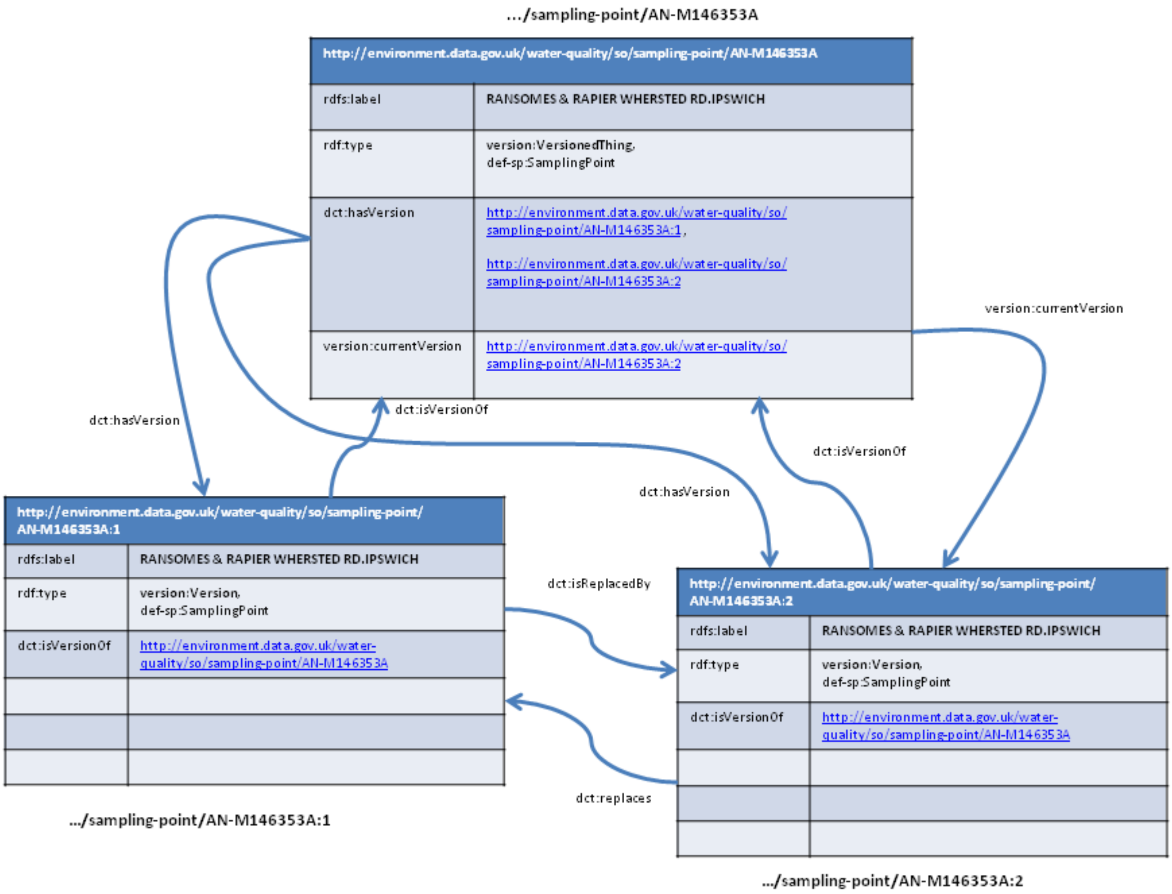


Figure 10: Versioning vocabulary for water quality information

In the Linked Data community one of the principles is that 'cool' URIs don't change or disappear. Creating a URI of the form {namespaceId}/{localId}[:{versionId}] implies that versionId becomes part of a potentially bookmark-able, reference-able, cite-able web identifier and may be required to persist for a long time.

Typically, a series of related access URI will be created as illustrated in the table below [12, 13]. A specific practice is still evolving, but is along the lines shown below.

Table 5: URI patterns for versioning

URI Pattern	Behaviour
{namespaceId}/{localId}	References and accesses the enduring entity
{namespaceId}/{localId}:{versionId}	References and accesses a specific version/snapshot
{namespaceId}/{localId}[:{versionId}]/current	An API to retrieve the current version snapshot relative to a specific version or the enduring entity.
{namespaceId}/{localId}[:{versionId}]/versionAt/{yyyymmdd[:hhmm]}	An API to retrieve the version snapshot current at a particular

URI Pattern	Behaviour
	date/time. (in the past)
{namespaceId}/{localId}:{versionId}/previous {namespaceId}/{localId}:{versionId}/next	An API to retrieve previous or next versions relative to some specific version.

The key point is that where {versionId} become part of URI that identify particular spatial-object versions there is as need to consider persistence of both the identifier and the data behind it. When making reference to a spatial-object from published data, we invariably refer to the enduring entity (a version:VersionedThing) rather than a specific version (version:Version). In most cases it is almost impossible to anticipate the temporal context a given application would be interested in. Instead, by making reference to the enduring spatial object - which remains reference-able over all future time and ideally accessible - a particular application can determine the temporal scope of interest and access the relevant temporal snapshots.

The approach taken for versioning also impacts the lifecycle information of versioned spatial objects. Different lifespans can be covered:

- the lifespan of the real-world phenomenon, which can only really be closed when it ceases to exist;
- the lifespan of the (enduring) spatial-object which is potentially quite different from the lifespan of the corresponding real world phenomenon;
- the lifespan of a spatial-object version (a snapshot). However that is the role of the version:interval property in the proposed versioning vocabulary.

Numerous variation and other approaches exist. Therefore, a detailed analysis would be required to come to a sound proposal. The INSPIRE Drafting Team had already recommended some time ago to initiate "a study on the topic taking requirements and software capabilities into account". In the INSPIRE implementation so far, the practical limitation to current data has rarely been raised as an issue.

It is obvious that support for the history of objects adds a new level of complexity and the added complexity needs to be balanced with the requirements and priorities.

Recommendation 15

It is proposed to disregard versioning in RDF vocabularies because versioning is a known open issue in general as the base standards from ISO/TC 211 and OGC do not natively support this.

If versioning would be explicitly required, with access to historic versions, it is recommended to search for more best practises that illustrate a potential versioning model.

3.1.18. Summary - Applicability of ISO/DIS 1950-2

The applicability of the conversion rules from ISO/DIS 19150-2 have been summarized in Table 6.

Table 6: summary of applicability of ISO/DIS 19150-2 conversion rules

Sub-clause of ISO/DIS 19150-2	Recommended usage for generating INSPIRE RDF vocabularies	Remarks
6.2.2 Ontology Name	modified	sub-packages of an application schema package should not become separate ontologies, but should be part of the application schema ontology instead of umlPackageName a code for the application schema should be used (the same that is used for the prefix)
6.2.3 RDF namespace	as-is	
6.2.4 Class name	as-is	
6.2.5 Datatype name	as-is	
6.2.6 Property name	modified	umlClassName should not be used inside the property name
6.2.7 Names for code lists	modified	In INSPIRE, code lists are not managed within the application schemas and therefore code list classes should not be converted, however guidance is needed in case code lists are embedded in the application schema (e.g. extensions). see 3.1.11
6.3 Package	modified	a dc:source declaration for packages that provides the title of the Ontology reference document or standard, is based on the assumption that the package is from the harmonized model, i.e. defined in a standard of ISO/TC 211. owl:versionInfo uses the version information in tagged value "version" instead of a date
6.4 Class	modified	dc:source should be omitted as it was considered to be of limited value It needs to be decided which properties (SKOS or DC properties) to apply for language-independent name, human readable name, definitions and descriptions.
6.5 Abstract class	as-is	
6.6 Class stereotype	ignored	see 3.1.10 The only exception is that based on stereotype <<FeatureType>> a class can be defined as a subclass of gfm:AnyFeature (General Feature Model vocabulary), iso19150-2:FeatureType and geo:Feature from the GeoSPARQL vocabulary.
6.7 Attribute	modified	dc:source can be omitted as it was considered to be of limited value, see 3.1.9 gcoDatatypes are not used, see 3.1.13 It needs to be decided which properties (SKOS or DC properties) to apply for language independent name, human readable name, definitions and descriptions. rdfs:domain should be left open, see 3.1.5 rdfs:range should support owl:unionOf for cases where multiple UML attributes are "merged" to an RDF property some attributes may be suppressed, see 3.1.14, or implemented using other RDF vocabularies, see 3.1.15

Sub-clause of ISO/DIS 19150-2	Recommended usage for generating INSPIRE RDF vocabularies	Remarks
6.8.1 Enumeration	modified	dc:source should be omitted as it was considered to be of limited value. It needs to be decided which properties (SKOS or DC properties) to apply for language independent name, human readable name, definitions and descriptions.
6.8.2 Code list	modified	see 3.1.11
6.9 Union	ignored	see 3.1.16
6.10 Multiplicity	ignored	see 3.1.6 In ISO 19150-2 multiplicity would be mapped to property restrictions owl:minQualifiedCardinality, owl:maxQualifiedCardinality, owl:qualifiedCardinality. allValuesFrom can be included in certain cases
6.11.1 Generalization	as-is	
6.11.2 Association	modified	iso19150-2:associationName not included, see 3.1.9 dc:source is omitted as it was considered to be of limited value, see 3.1.9 It needs to be decided which properties (SKOS or DC properties) to apply for language independent name, human readable name, definitions and descriptions. rdfs:domain should not be provided, see 3.1.5 rdfs:range supports owl:unionOf for cases where multiple UML attributes are "merged" to an RDF property, see 3.1.16. Alternatively, the range could also be left open (see 3.1.5) Note that ISO/DIS 19150-2 does not provide rules for association classes, see 3.1.16
6.11.3 Aggregation	as-is	
6.12 Constraint	as-is	but see 3.1.16
6.13 Tagged value	ignored	see 3.1.10
7.2 Rules for identification	as-is	
7.3 Rules for documentation	modified	It needs to be decided which properties (SKOS or DC properties) to apply for language independent name, human readable name, definitions and descriptions. iso19150-2:associationName not included, see 3.1.9
7.4 Rules for integration	as-is	
7.5 GF_FeatureType	modified	It needs to be decided which properties (SKOS or DC properties) to apply for language independent name, human readable name, definitions and descriptions. sub-class to GeoSPARQL Feature can be added, too; see 3.1.12
7.6 GF_PropertyType	modified	see comments on Clause 6 for ranges specified by types from ISO 19107 and other foundation schemas, see 3.1.13
7.7 GF_AssociationType	modified	see comments on Clause 6

Sub-clause of ISO/DIS 19150-2	Recommended usage for generating INSPIRE RDF vocabularies	Remarks
7.8 GF_AggregationType	as-is	
7.9 GF_InheritanceRelation	as-is	
7.10 GF_Constraint	as-is	but see 3.1.16

3.2. Towards a common methodology

One of the major outcomes of the pilot experiment is that, due to the amount of open issues, potential obstacles and implementation options, a common methodology cannot be elaborated yet. It is important to recognize that the development of RDF vocabularies for INSPIRE is still work-in-progress, and that it requires broader review and discussion as well as testing in applications. Moreover, any generated RDF vocabulary will require reviewing and additional edits. The table below describes identified issues, bottlenecks and implementation options. For some of them, suggestions and proposals from the consulted experts have been included.

Table 7: summary of issues related to the transformation of INSPIRE UML models.

Topic	Problem description	Suggestions made by consulted experts
Foundation schemas	ISO/DIS 19150-2 is not finalized and technical comments have been submitted to ISO/TC 211 as part of the DIS vote. These will be discussed during an ISO/TC 211 meeting in June 2014. Some of the comments overlap with issues raised in the reports, but others do not. At this stage in the ISO process it is not foreseen to raise any new comments, so it is likely that RDF vocabularies for the INSPIRE application schemas would not conform to the final version of ISO 19150-2, if the general direction of the methodology used in this document is used.	1. It was concluded that 'official' RDF vocabularies of the foundation schemas are needed in order to model INSPIRE's relationship with them. Therefore, it is recommended to take into account the disposition of the DIS comments. 2. In general, proposals to amend the rules from ISO/DIS 19150-2 should be reviewed.
Schema conversion rules	The schema conversion rule for union data types in ISO/DIS 19150-2 is insufficient as it does not handle cases where values are a mix of object or datatypes, or the same value type is used by more than one option. Association classes are currently not supported by the schema conversion rules in ISO/DIS 19150-2, but are required for INSPIRE application schemas. Some of the base types from ISO19103 are not covered by ISO19150-2	1. For association classes, the approach taken in GML3.3 (equivalent transformation of the UML) can be taken here, too. 2. Identify existing RDF types or alternatively define an INSPIRE RDF vocabulary for base types from ISO 19103 that are used by INSPIRE e.g. Measure.
Annotations	For the conversion of annotations of packages, classes and properties, several existing RDF vocabularies (SKOS, RDF, DC…) could be applied.	1. It needs to be decided which properties (SKOS or DC properties) to apply for language independent name, human readable name, definitions and descriptions of classes and properties

Topic	Problem description	Suggestions made by consulted experts
Properties	The preference for less context in the RDF/OWL properties seems appropriate. However, conflation of properties with identical names need to be addressed.	1. When properties have identical names, but inconsistent ranges or inconsistent semantics, the definition of the conflated property needs to be updated or separate properties with appropriate names have to be created. This requires review and editing of the RDF vocabulary. 2. Do not include rdfs:domain in the property declarations 3. If the ranges of the properties differ and both are classes, a union should be created as the range. 4. Similarly, properties should be reviewed, if distinct properties should be conflated into a single property. 5. Indicate whether properties are global, specific to a package or specific to a class.
Source and association names	ISO/DIS 19150-2 mandates that dc:source is included on classes and properties. ISO/DIS 19150-2 also includes association names in object properties that were derived from roles.	1. The RDF vocabularies should only include a dc:source property on the owl:Ontology resource. 2. In the INSPIRE application schemas they add no extra value and can be suppressed.
Stereotypes and tagged values	ISO/DIS 19150-2 includes tagged values in the RDF vocabularies. In general, stereotypes and tagged values are UML-specific extension mechanisms.	1. Stereotypes and tagged values should only be supported in schema conversion rules that map the values to native RDFS/OWL constructs and carry relevant information.
Code lists	The transformation of code lists into RDF vocabularies depends on the management of code lists and whether they are embedded in or separated from an application schema.	1. Decide on a holistic approach for code lists. 2. Decide, if properties that reference an open code list should use rdfs:Resource or skos:Concept as range. 3. The INSPIRE registry should support a SKOS representation for code lists (skos:ConceptScheme) and values in code lists (skos:Concept).
Features	Several upper ontologies are candidate for representing INSPIRE features (GeoSPARQL, Core Location, ISO 19150-2…)	1. Clarify to which upper ontologies INSPIRE spatial object types should be linked. 2. Decide about the geometry vocabulary. All 3 experts suggested GeoSPARQL for the time being. However, at this stage no preference for any of the vocabularies can be made and will to a large extend depend on future modifications and uptake by the community. 3. WKT is preferred over the GML serialization option, as long as there is no need to go beyond simple feature geometry types. WKT is more compact than the GML serialization.

Topic	Problem description	Suggestions made by consulted experts
Constraints	The ISO/DIS 19150-2 schema conversion rule maps constraints from the UML model. However, including OCL in the RDF vocabulary is questionable.	1. Probably the most reasonable way would be to include only the documentation of a constraint.
Use of existing RDF vocabularies	For several feature attributes and classes in INSPIRE application schemas (including GCM), commonly used properties and classes from existing RDF vocabularies can be reused.	1. Whenever the semantics of such properties matches that of a feature attribute, the existing property should be used instead. The same applies for classes. This requires review to ensure that the use of items from other vocabularies is appropriate. Example: prov:generatedAtTime / prov:invalidatedAtTime from PROV-O are good candidates for beginLifespanVersion / endLifespanVersion from the INSPIRE application schemas. 2. Some of the INSPIRE Annex themes are based on or related with existing models e.g. CityGML, GeoScienceML, … For these themes it is proposed to support the development of a theme-specific ontology and promote its endorsement by a standards organization.
INSPIRE Base types	Representation of at least the following base types in the INSPIRE RDF vocabularies requires more discussion: • SpatialDataSet • ThematicIdentifier • all types that represent general concepts that are not really INSPIRE-specific or spatial in nature; e.g. types related to documents, citations, contacts and parties	1. See also suggestions on use of existing RDF vocabularies
Versioning	The implementation of versioning has a high impact and effect on the RDF vocabularies.	1. It is proposed to disregard versioning in RDF vocabularies because versioning is a known open issue in general as the base standards from ISO/TC 211 and OGC do not natively support this. 2. If versioning would be explicitly required, with access to historic versions, the versioning model described by Chris Welty et al. could be a possible approach.
Voidability	There is no common design pattern for this in Linked Data. Voidable properties present a certain amount of difficulty to RDF under the open-world assumption. Just because the value of a property may not be given does not mean that there is no value for that property that could be given elsewhere.	1. It is proposed to not implement a conversion rule for the concept of voidability.

4. Transformation of INSPIRE data into RDF

Testing and developing guidance on how to transform INSPIRE feature data into RDF was beyond the scope of this study. Nevertheless, there is a common understanding that Linked Data requires to speak clearly and distinctly about the subject, which is either the spatial object (the abstraction) or the real-world phenomenon it abstracts. Good guidance and examples are needed that illustrate how feature instances should be represented in RDF as this information is not immediately accessible from the RDF vocabularies and the RDF vocabularies cannot be used for validation of the instance documents. In this paragraph an high-level approach is proposed on how INSPIRE features can be positioned in the Linked data context.

4.1. General context for transformation of INSPIRE data

An important aspect of instance transformation is the different modelling discourse that is used in the GI and Linked Data community. Whilst both communities regard 'data' as an information resource, the nature of Linked Data/RDF is that it demands explicit subjects. This is the consequence of the RDF which is made up of a collection of triples composed of a subject, a predicate or property and an object. Subjects are designated by URI (or blank) nodes; predicates are designated by URI only; and objects are designated by URI, blank nodes or literal values (Figure 11). So the question raises what the RDF subject actually is: the data (spatial object) itself or the abstracted phenomena.

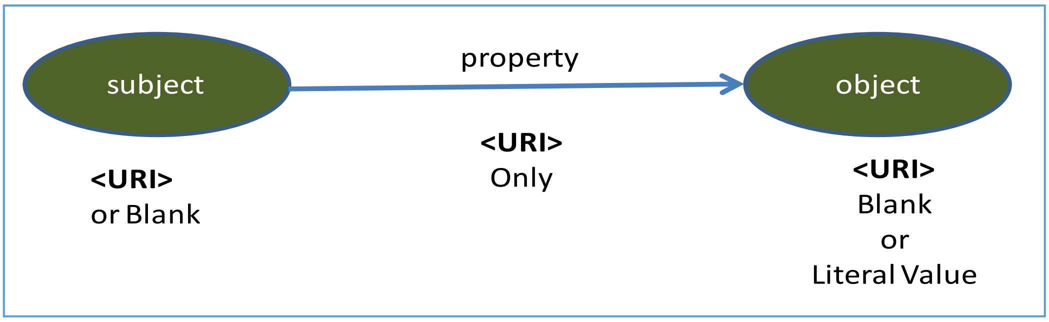


Figure 11: RDF triple structure with subject identifier for real-world phenomenon

RDF graphs are made up of a collection of statements and each statement requires an explicit subject node. Within RDF, in general, the subject URI can designate anything: real, imagined, abstract... anything. The key thing is being clear about intended subjects and avoiding conflation. In the case of spatial real-world phenomena it can be expected that at least one subject identifier within the RDF graph is explicitly intended to designate the real-world phenomenon about which statements are being made. However, in the GI community the subject of any implied statement is less explicit than in RDF. The General Feature Model (GFM) defined in ISO 19109 does not make a separation between properties where the subject is the real-world phenomenon and where the subject is the spatial object. According to the General Feature Model, it is usual practice to model both

- properties that describe the real-world phenomenon and
- properties that describe the feature ("spatial object" in INSPIRE terminology) document, i.e. which are feature metadata,

as feature properties.

Also the INSPIRE Directive and Implementing Rules do not contain an explicit requirement for resources (URIs) for the real-world phenomena. Only the feature resource is needed and both types of properties are linked to it. The feature can be identified with an object identifier to identify the abstraction rather than the abstracted thing or real-world phenomenon. In some application schemas a thematic identifier is modelled to uniquely identify the phenomenon described by the feature, which in INSPIRE is the closest analogy to the formation of URI to designate real-world phenomena. For example, a railwayStationCode from a managed code list can be transformed into a URI to be used as real-world subject designator.

Linked Data requires to speak clearly and distinctly about the subject, which is either the spatial object (the abstraction) or the real-world phenomenon it abstracts. This means that two resources and as a result two persistent URIs for each feature are needed, one for the INSPIRE feature document and one for the real-world phenomenon. There has been considerable debate in the web community about these issues and the conclusions and good practices have been documented in the W3C document "Cool URIs for the Semantic Web" [36].

The different modelling discourse that is used in the GI and Linked Data community will not impact the modelling of the RDF vocabularies, because there is a tendency in Linked Data to keep the domain of object properties 'open', possibly as open as 'rdfs:Resource'. Thus, the object properties are not explicitly linked to either real world things or spatial objects in the model. Nevertheless, it does have an impact on how instances are represented in RDF as it is important in Linked Data and the Semantic Web to be clear about the subjects.

For modelling the features as RDF instances, 3 different representations were analysed during the study:

- features represented as Nodes
- features represented as Graphs
- features represented as Graphs and Nodes

4.1.1. Features represented as Nodes

In this approach features are formulated as nodes in an RDF graph. This allows much more use of RDFS and OWL in describing RDF models/vocabularies derived from INSPIRE application schema. However, it does expose the potential to mistakenly conflate spatial-objects with the things which they model/abstract. Figure 12 shows what a node representation may look like for a small vocabulary.

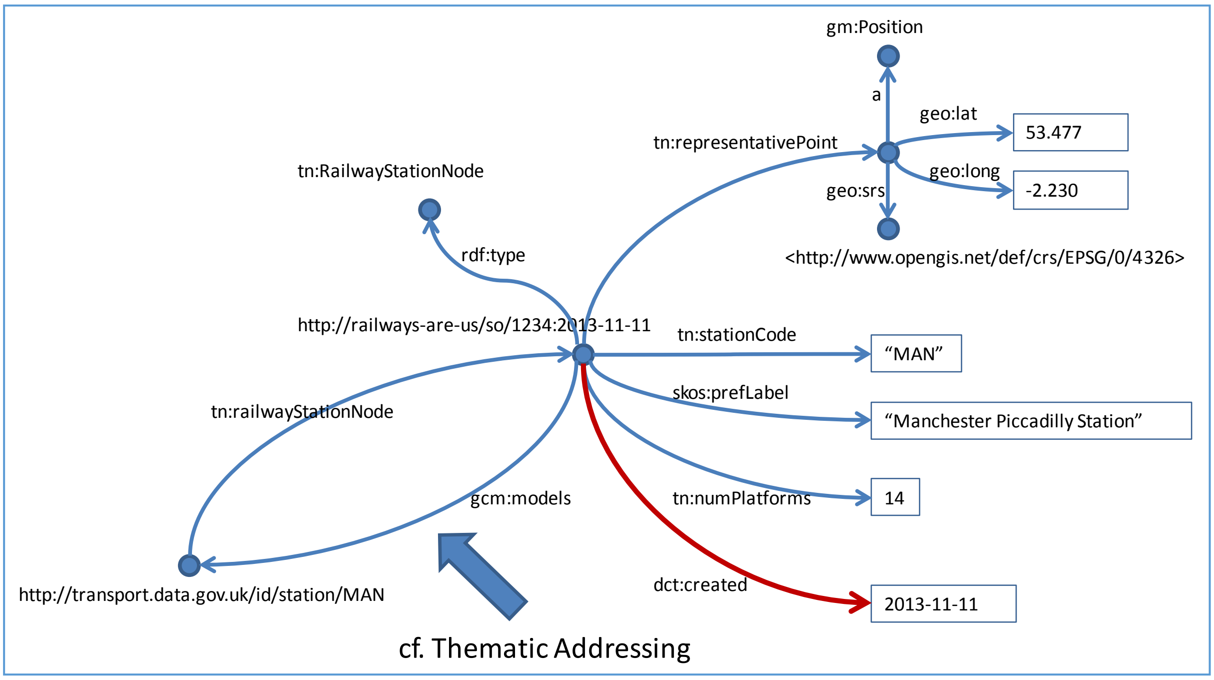


Figure 12: Features represented as nodes

The vocabulary depicted above provides a property, gcm:models, whose role is to link a spatial-object to the real-world phenomenon that it abstracts. It provides a way to explicitly make the link between a feature and the thing that it abstracts. It can serve as a 'join' point between multiple features that abstract (aspects of) the same thing, whether multiple features published by the same publisher or features from different publishers. Note that in this approach the subject URI used in making statements about a real-world phenomenon is the feature URI rather than the real-world subject URI. This risks conflation of the feature (or abstraction) with the real-world phenomenon. This raises an issue: two entities with distinct URIs, but in writing RDF we are in fact proposing the feature URI as a subject designator to make statements about the real-world phenomenon. Using features as subject nodes make speaking clearly and distinctly about spatial-objects and the thing it abstracts somewhat difficult.

4.1.2. Features represented as Graphs

Another approach is to regard a small collection of statements about some subject as serving the role of a feature in UML. Figure 13 shows how the earlier example is extended to express RDF graphs with names.

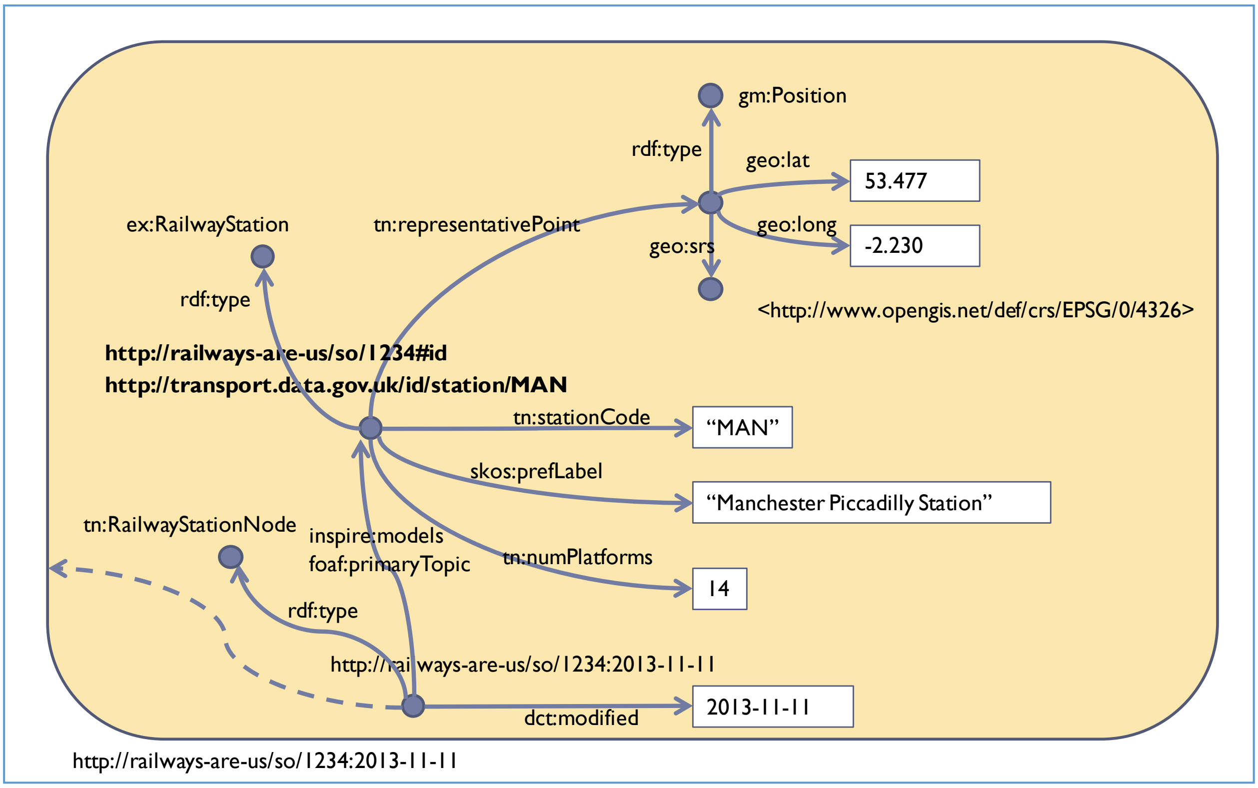


Figure 13: Features represented as graphs

Notice that within the object-graph there are two statement subjects - the real-world station and the feature itself. This allows statements to be made separately about the station and the feature. One can clearly express provenance of the feature (‘feature metadata’) distinct from data expressed about the station.

However, in this approach one particular INSPIRE application schema would constrain the properties that could be used in speaking about a railway station within the context of a RailwayStationNode to just those that are allowed for such a feature. For example, the feature above would be malformed, as an INSPIRE RailwayStationNode if it made statements about Manchester Piccadilly that need to be made in the context of a RailwayStationArea feature e.g. the geometric extent of the railway station.

Moreover, constraints on the form of what can be said are not commonly used in the Linked Data community and their modelling languages RDFS and OWL.

4.1.3. Features represented as Graphs and Nodes

When drawing the previous two subsections together, ideas from the previous approaches can be merged, and potentially address the 2 identified problems:

- the conflation issue mentioned in the node approach.
- The constraining/validating issues as a consequence of using 2 subjects in one single graph i.e. one for the real world phenomenon and one for the feature.

In the 'graphs and nodes' approach most of the characteristics of the 'features as nodes' formulation are retained but a document node which is used as a graph name (positioning the graph itself as a document that describes both itself and a feature) has been added. A feature is considered as an information abstraction that models a real-world phenomenon and which is itself described in a feature document. In this way, lifecycle information and feature metadata associated with the feature publication can be associated with the document that describes it, and no longer with a feature as a 'second' subject within the graph.

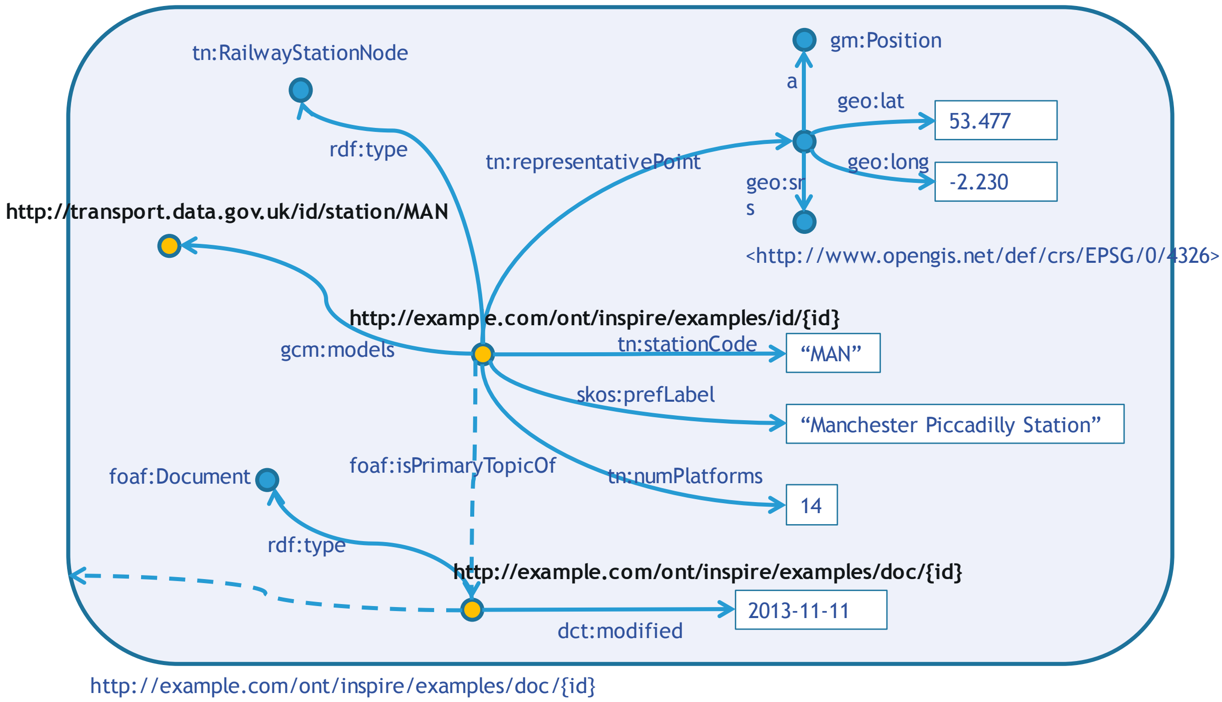


Figure 14: Features represented as 'nodes and graphs'

Hence, as depicted in Figure 14 we have 3 things:

- a real-world station, http://transport.data.gov.uk/id/station/MAN modelled by
- a spatial-object - http://example.com/ont/inspire/examples/id/{id} described in a
- graph or feature document http://example.com/ont/inspire/examples/doc/{id} .

The feature “http://example.com/ont/inspire/examples/id/{id}” is now clearly articulated; we make a pragmatic choice to use the feature URI as a subject for making statements whose real subject is the modelled real-world 'thing' (the ‘models’ relation is comparable with thematic addressing in INSPIRE); and we use a document URI, “http://example.com/ont/inspire/examples/doc/{id}”, as a subject URI to make statements that are really about the feature (feature metadata).

Recommendation 16
The choice amongst the three formulations affects the generation of RDF instance data. It affects much less the translation of INSPIRE application schema into RDF vocabularies. The formulations "..as Nodes" and "..as Nodes and Graphs" are the most pragmatic and once one admits the third subject (the document describing the spatial-object) they are more or less identical. However, making this choice is an important open issue that needs to be settled with the stakeholders.

4.2. Scenarios in INSPIRE

To do this, 3 scenarios that are specific to INSPIRE are listed and depicted (Figure 15) below to illustrate how identifiers could be used in case 2 features are referring to the same real-world phenomenon. This is a case in the INSPIRE SDI that frequently occurs i.e. several digital abstractions may and generally will exist for the same real-world phenomenon. The same organisation may manage multiple datasets on different scales. For example, mapping agencies and road authorities both will usually manage data about the road network in separate datasets. 3 scenarios can be described:

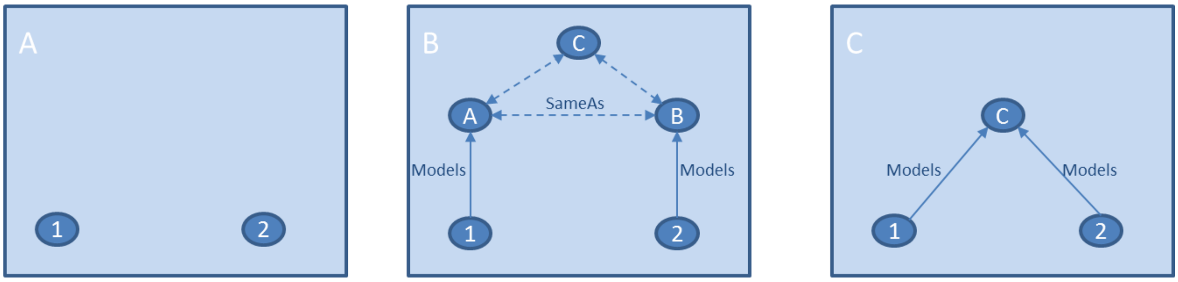


Figure 15: 3 scenarios in INSPIRE

- Scenario A: 2 different spatial objects (1 and 2) without reference to a real-world phenomenon;
- Scenario B: 2 different spatial objects (1 and 2) that model a real-world phenomenon using 2 different real-world identifiers (A and B);
- Scenario C: 2 different spatial objects (1 and 2) that model a real-world phenomenon using 1 unique real-world identifier (C).

Scenario A will be applicable in most cases because the INSPIRE directive has no requirement for identifiers for real-world phenomena. INSPIRE only requires an object identifier to identify the abstraction rather than the abstracted thing. If data providers would publish their data as Linked Data, they have to create themselves a new identifier to refer to the real-world phenomenon or refer to an already existing identifier. This requirement is represented in scenario B. It is important to understand that there is no requirement that only a single URI is used for the real-world phenomenon - it is perfectly fine to use different URIs. In the Linked Data context it would be preferable, if only a single URI is used consistently for the same real-world phenomenon (Scenario C). However, that would imply an organisational challenge to implement the mechanisms and processes for managing the identifiers for the real-world phenomena and it would require significant efforts in the Member States to establish the necessary governance and infrastructure.

When multiple URIs for the same real-world phenomenon exist and this is known, that fact may be declared using owl:sameAs. Note that here the distinction between the real-world phenomenon and the feature document is essential as the sameAs would only make the real-world phenomena the same, but not the feature documents.

INSPIRE has the concept of "thematic identifiers". These are (typically local) identifiers of the real-world phenomenon. It will often be useful to include a triple with the local part of the thematic identifier as the object (subject is the real-world phenomenon), for example a NUTS code or a postal code.

In the INSPIRE application schemas for Annex I, the thematic identifier properties cannot be identified without a review. In the Annex II/III schemas, the value type ThematicIdentifier should have been used for these cases, but it is likely that this is not always the case. In the absence of a clear rule how to identify thematic identifiers it might be worth to consider using a stereotype to identify such properties in the UML model.

4.3. Implementation options

4.3.1. URIs for INSPIRE

All abovementioned scenarios should be supported by a future implementation and have a direct impact on how redirection should be implemented. Scenario B would require a HTTP 303 redirect to the spatial object identifier from each real-world identifier, whereas scenario C would require a HTTP 303 redirect to a page representing a choice list to pick the right spatial object. The latter would also be the case if in scenario B the two (local) real-world-identifiers A and B would link, in turn, to a unique real-world identifier C that is maintained by another authority.

Good practices have been documented in the W3C document "Cool URIs for the Semantic Web" [36]. Basically there are two options ("Hash URIs" vs. "303 URIs"), both with advantages and disadvantages. There is no apparent need for INSPIRE to select one approach over the other in general, as discussed by the W3C document.

In line with the preferred scenario 303 URIs are used in the following example, i.e. the real-world phenomena are identified by URIs

```
http://example.com/ont/inspire/examples/id/{id}
```

which redirect to the URIs of the feature documents that use

```
http://example.com/ont/inspire/examples/doc/{id}
```

Note that the feature documents returned contain also triples with the real-world phenomenon as subject as well as other dependent resources such as geometries and network references.

In the next example Hash URIs are used, i.e. real-world phenomena are identified by URIs

```
http://example.com/ont/inspire/examples/{id}#thing
```

and the feature document URI is

```
http://example.com/ont/inspire/examples/{id}
```

The feature document contains the triples with the real-world phenomenon as subject (with the local id "thing") as well as other dependent resources (geometries and network references), again with local ids.

Note that using content negotiation, the request to

```
http://example.com/ont/inspire/examples/{id}
```

with application/gml+xml as the preferred representation would (be expected to) return the GML document or send a redirect to a WFS GetFeatureById request. There is an issue with this as in the GML representation there would be no resource with id "thing". This may turn out to be an issue with the use of Hash URIs for INSPIRE data.

4.3.2. Linking real-world objects with a featuredocument

As described above two resources would be needed to represent INSPIRE data in the Linked Data context. As a result we would assign two persistent URIs for each resource, one for the INSPIRE feature document and one for the real-world phenomenon. It is also useful to link the two resources. The following approach can be used:

```
:realworldobject    rdfs:isDefinedBy    :featuredocument .
:featuredocument    foaf:primaryTopic   :realworldobject .
```

The use of rdfs:isDefinedBy follows the convention used in [36]. The use of foaf:primaryTopic for the inverse statement seems to be frequently used in the Linked Data world; a side effect is that :featuredocument is a foaf:Document, but this should be appropriate. However, there seems to be discussion whether the property 'rdfs:isDefinedBy' can be used to link real-world objects with a featuredocument. According some Linked Data experts, 'rdfs:isDefinedBy' is mainly used for vocabulary terms (classes and properties) in an ontology that only have a single point of (authoritative) definition as a web vocabulary. rdfs:isDefinedBy is also used to enable versioning of ontology documents whilst maintaining stable term URI. An alternative would be to use foaf:isPrimaryTopicOf or wdrs:describedby from the POWDER-S vocabulary [37], but preferably this should be discussed in a wider group of experts.

4.3.3. Feature type classifications

A second open item for discussion is which of the subjects is going to be typed using the feature type classifications. Or presented in a different way, which of the two following statements is correct:

```
:featuredocument    rdf:type            cp:CadastralParcel
:realworldobject    rdf:type            cp:CadastralParcel
```

Intuitively, the second statement feels more correct as the classification is a classification of the real-world phenomenon. On the other hand, cp:CadastralParcel is a sub-class of the base classes representing features, e.g. gfm:AnyFeature and geo:Feature, so looking at the use of terminology it would seem more correct to select the first option. This confusion is the result of the fact that the General Feature Model – from a Semantic Web point of view – collapses both resources. This is probably a reflection of the fact that the GI community uses the term feature for both – and humans know from the context what is meant. On reflection, it seems appropriate to follow the intuition and use the second option. However, one should be aware that the inconsistency in the use of the term feature may be confusing to newcomers.

An alternative approach is to use the derived feature (e.g. cp:CadastralParcel) as a 'mix-in' class attributing rdf:type to the real-world phenomenon: a single generic feature class that can be used as domain or range of properties that are 'about' the spatial-object (as a graph) itself rather than the real-world phenomenon that the object abstracts. It may even be useful to create two classes per feature type, one as a 'mix-in' applied to the real-world phenomenon and to serve in the expression of property domain/range constraints on properties that are about the real-world phenomenon and a second to serve as tag on the feature that is indicative of the abstraction contained there-in. The information associated with these second classes could go further and enumerate the properties that can be used to described the real-world phenomenon as illustrated below:

```
cp:CadastralParcel
a       owl:Class ;
owl:disjointWith        cp:CadastralParcelObject ;
rdfs:label      "CadastralParcel "@en;
rdfs:comment    """A mix-in class applied to entities that in some cases may be regarded as
items in a land registry - typically, but not exclusively, cadastral parcels."""@en ;
# Restriction to indicate expected use of nationalCadastralReference with CadastralParcel
        rdfs:subClassOf [
                a       owl:Restriction
                owl:onProperty          cp:nationalCadastralReference;
                owl:minCardinality      1
        ]
##... more restrictions to indicate use of other properties.
        .

cp: CadastralParcelObject
        owl:disjontWith         cp:CadastralParcel;
        rdfs:label              "CadastralParcelObject"@en;
        rdfs:comment            """A class for spatial-object graphs expected to describe
'thing' using only the properties associated with describing a CadastralParcel.""" ;
        rdfs:subClassOf                 gcm:SpatialObject ;
        ## Loose restriction to CadastralParcel expressions
        rdfs:subClassOf [
                a       owl:Restriction ;
                owl:onProperty          gcm:models
                owl:allValuesFrom       cp:CadastralParcel
        ] ;
        ## Annotation properties to enumerate allowed and voidable properties.
        gcm:allowedProperties           (...) ;
        gcm:voidableProperties          (...) ;
```

The two series of classes, for real-world phenomena and for spatial-objects are mutually disjoint i.e. cp:CadastralParcel and cp:CadastralParcelObject are disjoint. The derived mix-in classes are also not subclassed from a common gcm:SpatialObject class - indeed they are necessarily disjoint from it. A drawback of this 'mix-in' approach is that a second classification hierarchy is included that would make the resulting vocabulary considerably larger and more complex.

4.4. Towards a common methodology

Another major outcome of the pilot experiment is the observation of a different modelling discourse that is used in the GI and Linked Data community. In the Linked Data community it is common to separate between 'real world things' and the information you can get about these 'real world things' over the internet. The question is how this can be implemented in the context of INSPIRE. In two well-known geo-Linked Data resources, GeoNames and DBPedia, the separation between the two seems to be in place. In the GI world, however, there has traditionally not been a clear separation the different subjects. The spatial object is often an abstraction, i.e. an entity on a map, and not a representation in the real (time and space) world, although it may represent a real-world phenomenon.

From the perspective of the RDF vocabularies there is no distinction between the two subjects, because the rdfs:domain is not included (see 3.1.5). Nevertheless, it has impact how instances are represented in RDF as it is important in Linked Data and the Semantic Web to be clear about the subjects. Good guidance and examples are needed that illustrate how feature instances should be represented in RDF as this information is not immediately accessible from the RDF vocabularies. The table below describes identified issues, bottlenecks and implementation options related to how feature instances can be represented in RDF. For some of them, suggestions and proposals from the consulted experts have been included. Instance examples are provided in the individual reports of the experts ([2], [4] & [6]).

Table 8: summary of issues related to the transformation of INSPIRE data.

Topic	Problem description	Suggestions made by consulted experts
Representing INSPIRE features as RDF instances	In the experiment 3 approaches were compared to represent INSPIRE features as RDF instances.	• Further discussion is needed to check if the 'feature representation as graphs and nodes' is appropriate and feasible.
Properties	Properties in INSPIRE can refer to the real-world phenomenon or represent property metadata.	• Feature properties that represent feature or property metadata should be clearly identified as such in the UML model, for example, by using additional stereotypes.
URIs	Depending on the chosen approach to separate the different subjects, a corresponding URI pattern should be established to represent features and feature documents.	• Different URI patterns are needed to identify feature documents and real-world subjects. • Guidance is needed on the use of 303 URIs vs Hash URIs • The role of HTTP content negotiation needs to be discussed and how it may be implemented.
Linking real-world objects with a feature document	When two persistent URIs are assigned for each resource, one for the INSPIRE feature document and one for the real-world phenomenon, It would be useful to link the two resources.	• Guidance is needed how these two resources (real-world phenomenon and feature document) are related (e.g., whether to use rdfs:isDefinedBy, foaf:isPrimaryTopicOf, etc.).
Feature type classifications	It is unclear which of the subjects is going to be typed using the feature type classifications.	• Guidance is needed whether the real-world phenomenon, the feature document or both are typed with an INSPIRE spatial object type classification,
Multiple resources for the same real-world phenomenon	Several digital abstractions may and generally will exist for the same real-world phenomenon, e.g. at different scales or representing different thematic domains. For example, a river may be represented as a polygon at large scales or a line at smaller scales, and it may be represented as a link in a water transport network or a physical water body.	• Multiple resources identifying the same real-world phenomenon can be related using owl:sameAs.

Topic	Problem description	Suggestions made by consulted experts
Versioning	Versioning requirements impact the conceptualisation of spatial object. For example, does an object identifier identify an object (over all its lifetime), an object version (snapshot), the container of the current state or some combination of all of the above. This affects how object references are made using URI and how instance data is transformed into RDF	• Guidance is needed how versioning of INSPIRE features should be dealt with in URI patterns
Object identifiers	INSPIRE distinguishes two types of object identifiers in spatial data sets: the inspireId and the thematic object identifier.	• Thematic Identifiers are the closest guess to the real-world subject and may be used in an URI to refer a real-world phenomenon. However, this should be verified for each of the application schemas if the modelled thematic identifier is appropriate.

5. Transformation tools

The pilot experiment showed that manual or semi-automatic conversion of INSPIRE application schemas is laborious and unsustainable. This chapter describes potential schema and data conversion tools to be used for the transformation of INSPIRE-related source data (in their original format and schema) as well as INSPIRE-compliant data (in GML) to the generated RDF vocabularies.

5.1. Schema conversion tools

For schema conversion the most relevant candidates are listed below. Note that no conversion method has been found that fully automates the task. There is always a necessity of configuration / defining mappings beforehand, and/or manually checking and adapting the conversion result to get a usable ontology.

5.1.1. ShapeChange

ShapeChange (http://shapechange.net) is an open source Java tool that takes application schemas constructed according to ISO 19109 from a UML model and derives implementation representations. RDF is one of the supported representations. Currently the RDF output option of ShapeChange is experimental, but a new version of ShapeChange supporting RDF output conform ISO 19150-2 (current DIS), and supporting the mapping of class- and property names to well-known vocabulary terms, will become available in May.

5.1.2. Fullmoon

Fullmoon (https://code.google.com/p/fullmoon-framework/) from CSIRO may have broadly similar capabilities to ShapeChange however it is harder to set up and experiment with. It has not been fully tested during the pilot experiment.

5.1.3. XSLT transformations

One other option is to create a custom transformation (e.g. in XSLT or SPARQL) from UML exchange format XMI to RDF/XML. However, this requires a detailed knowledge of the XMI formats (or other XML based source formats, and much of that is already encapsulated in ShapeChange/Fullmoon

5.2. Data conversion tools

The transformation of 'instance' data into RDF expressed using a particular set of vocabularies is quite different from the process of transforming the application schema itself and likely requires different tools. The tasks is illustrated in the diagram below and apart from the data that is going to be presented as or transformed into RDF, there are two other indispensable pieces of input. Firstly, the target Linked Data vocabulary which has been the main focus of the experiment; and secondly an approach for the Linked data URI space where the data is to be published. The latter interacts with the governance of persistent URIs. It is necessary to consider how the publication is going to be organised, maintained and updated - particularly with respect to the persistence of links that others make to the published data.

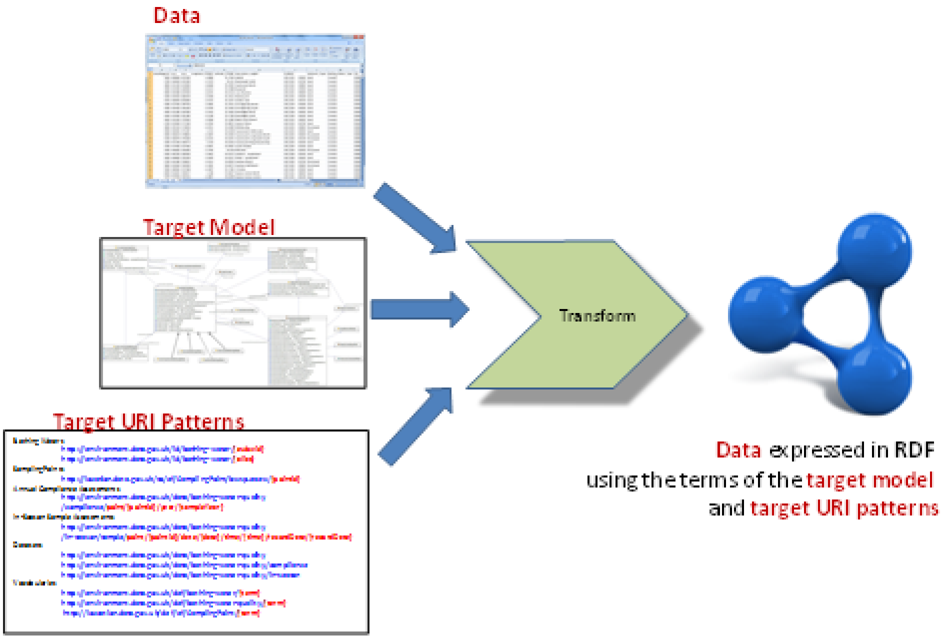


Figure 16: **Data Conversion process**

The transformed data may be published:

- As individual 'documents' (.json, .jsonld, .gml, .rdf, .ttl, .html) published conventionally on a web server. Each 'document' is effectively a 'small' graph about some primary topic that may carry its own 'document' metadata. This style of publication enable 'link-following', but provides no means to query the data based in its content. The publication can be done for both "spatial-objects as nodes" and "spatial-objects as graphs" approaches.

- Into the default-graph of triple store. This loses object boundaries in the sense that it is not possible in general to segregate statements back into groupings representing the individual 'documents' that were contributed. This only works for "Spatial-Objects as Nodes" approach. The advantage is that the data are intrinsically queryable. A Linked Data URI can be 'animated' by:

    - frontend technologies ranging from Apache mod_rewrite to convert request URI into SPARQL describe queries;
    - Pubby which provides RDF and HTML output formats;
    - Linked Data API[7] (Elda[8] and Puelia[9] implementations) which provides URI based querying capabilities and RDF, HTML and developer centric JSON and XML formats.

- As named graphs in a Quad Store (with or without a UNION default graph). This is very similar to the 'individual documents' approach, except that each document is published as a distinct graph in a SPARQL dataset. A UNION default graph gives a merged view with potential for different object versions and/or different objects to provide contradictory views via the

[7] https://code.google.com/p/linked-data-api/wiki/Specification

8 https://github.com/epimorphics/elda

9 https://code.google.com/p/puelia-php/

default graph. It can be implemented for "Spatial-Objects a Graphs" and "Spatial-Objects as Nodes" approaches. The data are intrinsically queryable and Linked Data URs can be 'animated' as above.

These approaches are all based on the Extract, Transform and Load (ETL) model of publishing. Alternatively, data maybe left within an RDBMS and a query translation data base adapter e.g. D2RQ[10] can be implemented to create a SPARQL endpoint that maps inbound SPARQL request to SQL queries and SQL results either back into SPARQL result sets or RDF graphs. Once a SPARQL endpoint is deployed, link-following and/or URI based querying can be obtained in the same way (mod_rewrite, Pubby or LDA, etc...). More recently the W3C has released the R2RML specification (http://www.w3.org/TR/r2rml/) that can support the definition of both query transformation and ETL transformation of the data.

Another approach is to transform data on top of OGC web services e.g. GetLOD[11] takes data from WFS or WCS services and serves them as Linked Data.

In any case, at this moment we are not aware of any easy-to-configure tools for such transformations and expect that either development work or configuration by experts is required for most dataset transformations. Below, we have listed a number of tools that can be used, however they have not been tested in the experiment.

5.2.1. DCLIB

DCLIB[12] (DataConversionLibrary) is an open-source project from Epimorphics. It provides a data conversion library that can be built into a larger data conversion workflow and a simple command line tool for the direct execution of transformation. DCLIB uses an embedded expression language (JEXL[13]) which provides a means to manipulate input values (eg. date and time parsing, creating composite values from multiple fields, syntactic formatting of literals and the like). DCLIB is a work in progress and continues to acquire new features.

5.2.2. OpenRefine

OpenRefine[14] is a powerful tool for working with messy data, cleaning it, transforming it from one format into another, extending it with web services, and linking it to databases. There are several extensions available for OpenRefine among which the DERI RDF extension. OpenRefine is a useful tool for developing a data transform in an iterative fashion, however in operation it requires the entire data set being transformed in memory - which limits the size of the data tables that can be transformed. It can be a useful way to generate test sets of 'correct' data against which transformation via some other technique can be checked. The refine framework introduces a key notion of reconciliation, such that references made in one data set can be reconciled against data in a foreign dataset.

5.2.3. Topbraid Composer

The Topbraid suite of products from TopQuadrant include facilities to create graphical mapping from a source table to a target vocabulary. There is a function library for manipulating and

[10] http://d2rq.org/

[11] http://www.planetek.it/eng/products/all_products/getlod

[12] https://github.com/epimorphics/dclib

[13] http://commons.apache.org/proper/commons-jexl/

[14] http://openrefine.org/

combining/splitting data from different fields. The resulting transformations can be exported for re-use in an automated data conversion workflow.

5.2.4. FME

It is known that others have used FME from Safe Software to create RDF from spatial data.

5.2.5. XSLT stylesheets

XSLT stylesheets have been used to convert GML data to RDF. This has been described, for example, in an article of IJSDIR[15] and in the GeoKnow project[16]. The methodology is based on the fact that GML has its roots in RDF and still has a basic object-property structure that can very easily be translated to triples. Because of this, it is easy to define a transformation for translating any correctly structured (that is, conformant to the object-property triple structure) GML data to RDFS/OWL automatically. As an experiment a transformation was implemented using XSLT 2.0. In a Generic-GML2RDF script, well-known GML content elements such as names and descriptions were mapped to their RDF equivalent. Objects, including nested features, data types and properties were recognized based on their place in the triple structure and are transformed accordingly. However, there are several issues to be resolved before using this for anything other than experimentation:

- There may be problems with the method, such as how to translate hyperlinks in the GML data to resources that reside in web services. This has not yet been considered.
- The stylesheet is highly experimental and should be extensively tested with INSPIRE data.
- Support for converting GML geometries to WKT is incomplete (only gml:Point and gml:Surface geometries at the moment, these are transformed to WKT) and should be extended, either by implementing rules for transforming geometries to WKT in XSLT, or by calling an external library (e.g. ogr2ogr) for this.
- Minor changes may be needed to support GML 3.2 (3.1 supported at the moment).

[15] http://ijsdir.jrc.ec.europa.eu/index.php/ijsdir/article/view/351

[16] https://web.imis.athena-innovation.gr/redmine/projects/geoknow_public/wiki/Inspire2RDF

6. Implications to other INSPIRE components

This chapter lists a number of potential implications to other INSPIRE components, when transforming INSPIRE models and data into RDF.

6.1. Code list register

INSPIRE in RDF has a few consequences for registers and registries:

- Registries including local Member State or thematic extensions to the INSPIRE code lists should support SKOS (skos:ConceptScheme and skos:Concept).
- The existing INSPIRE registers and feature catalogues should be extended to include references to derived RDF terms, in particular where mapping to pre-existing non-INSPIRE vocabularies is used e.g. relating a ResponsibleParty to a foaf:Agent.
- The registers need to be capable of adopting terms used and defined elsewhere so that their use may be validated. This applies both to:
    - Terms defined in local members state or organisational registers (federation); and
    - Terms whose use is imported from widely used RDF vocabulary (e.g. from ORG or FOAF or DC) without themselves ever having been elements of an established application schema (their use is more a consequence of a mapping into RDF).

6.2. RDF vocabularies

The RDF vocabularies must be available under their http URIs.

6.3. Media type register

The media types text/turtle and application/rdf+xml should be added to: http://inspire.ec.europa.eu/media-types/.

JSON-LD with geo-support is a very promising option for publishing linked geo-data in an easily consumable form. JSON-LD (a recent W3C recommendation [38]) is a JSON encoding for Linked Data. It lets you add meaning to the terms and values in a JSON document. This is done inside a @context object that is either referenced from or embedded inside the JSON document. Applications that are not aware of what @context is, can simply ignore it while applications that are aware, can parse the @context and gain knowledge on the semantics of the JSON data. GeoJSON [39] and JSON-LD can be combined and can be offered as the web-encoding for linked geospatial data. GIS server software that already offers JSON as encoding could create GeoJSON-LD as well.

6.4. Metadata

In the framework of ARe3NA, work has been done on the alignment of INSPIRE metadata with the "DCAT application profile for data portals in Europe" (DCAT-AP) [40], a metadata profile developed in the framework of the ISA Programme, based on the W3C Data Catalog vocabulary (DCAT) [41]. This work has contributed to the development of the geospatial extension of DCAT-AP (GeoDCAT-AP)[17]. This work is important to enable that INSPIRE metadata may be provided in an RDF representation and to understand the relationship and roles of ISO 19115 metadata in contrast to DCAT and other metadata vocabularies such as Dublin Core (DC) [42], Vocabulary of Interlinked Datasets (VoiD) [43],

[17] https://joinup.ec.europa.eu/node/139283

and whether and how to project ISO 19115 compliant metadata in a form that aligns with VoiD and DCAT.

6.5. Discovery services

To fit into a Linked Data environment, discovery services that support representations like DCAT / DCAT-AP and RESTful APIs should be supported. CKAN (http://ckan.org) is an example for a software product that may be used for this purpose (which also include some support for CSW).

6.6. Download services

A GeoSPARQL endpoint would likely be a candidate for a Direct Access Download Services that provides RDF triples. However, details would need to be analyzed as the INSPIRE regulation has a number of detailed requirements that would need to be covered in a technical guidance document.

Although INSPIRE distinguishes only between a download of a dataset and the demanding direct access with query support, all http URIs should be required to resolve in order to meet expectations for Linked Data. The latter involves implementing content negotiation. A common pattern of the web is for the truncations of hierarchical URI to list entities, for example:

- http://environment.data.gov.uk/id/bathing-water lists UK bathing waters designated under the EU Bathing Water Directive, while
- http://environment.data.gov.uk/id/bathing-water/ukc2102-03600 is a particular bating water in the North East of England.

Similarly parameters added to list URIs can be used to generate filtered responses, e.g.:

- http://environment.data.gov.uk/id/bathing-water?type=LakeBathingWater limits responses to bathing waters that are lakes, and
- http://environment.data.gov.uk/doc/bathing-water?min-samplingPoint.easting=362452&max-samplingPoint.easting=494951&min-samplingPoint.northing=159624&max-samplingPoint.northing=302123 limits responses to bathing waters within a given bounding box.

At least, in principle, for GML responses these request URIs could be transformed into WFS requests that return feature collections corresponding to the requested items.

In addition, support for a RESTful API would be helpful for application developers. An example is the Linked Data API.

6.7. Access control

Access control mechanisms that are API-aware and/or content-aware (i.e. which parse and/or edit requests and/or responses) need to support INSPIRE RDF. Linked data resources are accessed through the HTTP protocol, consequently access control methods already used on the web can be applied in this case as well. There seem to be no extra technical implications. The W3C Linked Data Platform specification [44]briefly states this in section 8.

However, while this is not a requirement, there is a general expectation that Linked Data is open data and URIs do not resolve to 403 responses. After all, in order to achieve at least a one star rating on the five-star Linked Data scale, the data must be available under an open license.

6.8. INSPIRE Geoportal

The INSPIRE Geoportal would have to be updated accordingly, at least the metadata and discovery support.

Directly supporting Linked Data might require a redesign, although this is difficult to say without knowledge about implementation details. With regard to open data, an integration into the pan-EU Open Data Portal, which harvest metadata from EU Member States, would seem logical.

6.9. Validators and testing tools

These would need to be updated, if there is demand to support testing and validation of RDF data and metadata. It may be worth mentioning that activities are under way at W3C e.g. http://www.w3.org/2012/12/rdf-val/ .

6.10. Monitoring and reporting

In the yearly monitoring of the implementation and use of the Member States' SDIs one of the things evaluated is the accessibility of spatial data sets through view and download services. Accessibility through SPARQL endpoints and as Linked Data on the web could be added to the monitoring procedure.

7. Conclusions and further steps

This section puts forward the main conclusions of this experiment and further steps to be taken for transforming INSPIRE into RDF.

The exploration of the transformation of INSPIRE UML models to RDF vocabularies, which has been done on the basis of an experiment, clearly demonstrated that a number of aspects need to be taken into consideration when defining a set of conversion rules. The results of the experiment also gave insight into the challenges of transforming INSPIRE data in RDF.

Based on this study, the following main conclusions could be derived:

1. The methodology for the UML-to-OWL conversion is to a large extent dependent on the ISO 19150-2, which has been published as an international standard in June 2015. The results of this study were provided to the editor of standard as part of the review process in August 2014, but no feedback was received. The final version of the standard should be analysed to see if any of the issues found with the conversion rules have been taken into account.

2. RDF vocabularies that strongly show UML roots do not really reflect common practice in the Linked Data community. This should be taken into account when defining a final set of conversion rules.

3. Common practice shall be needed with respect to the use of existing vocabularies e.g. geometry representation in RDF. This means that the choice for certain existing vocabularies is also determined by the status and the ongoing development of these vocabularies.

4. The issues mentioned in this report require broader review and discussion as well as testing in applications, before any formal technical guidance can be given to the INSPIRE community. This could include a review of research on generic UML-to-OWL mapping methodologies (e.g. [45], [46]).

5. It is unlikely that an automatic conversion of INSPIRE UML models can be done for every INSPIRE Annex theme based on a set of mapping rules. Any automatically generated RDF vocabulary will require reviewing and additional edits due to the theme-specific context.

6. Good guidance and examples are needed that illustrate how feature instances should be represented in RDF as this information is not immediately accessible from the RDF vocabularies. The generation of instance data is significantly affected by the approach for representing INSPIRE spatial objects and the differentiation between the real-world phenomenon and the abstraction of this.

Given the difficulties and issues encountered during the experiment, it is not surprising that a definitive set of guidelines on how to transform INSPIRE to RDF cannot be drafted at this moment. Further testing and reviewing will be necessary to explore different approaches. The conducted experiment also revealed that the transformation to RDF exposes the need for a fundamental and philosophical discussion on the different modelling discourse between the GI and the Linked Data community. On the other hand, there is an emerging need to expose INSPIRE data as RDF in a short term so that other communities can refer their data to INSPIRE. Therefore, in order to provide short term guidance, the requirement for an in-depth approach – which is a time-consuming process- needs to be balanced with the need for pragmatic solutions that can be offered to the interested community.

8. Bibliography

[1] Deloitte Consulting, "Study on RDF & PIDs for INSPIRE - D2: State of play," European Commission - Joint Research Centre, Ispra, Italy, 2014.

[2] L. van den Brink, "Methodology for automated conversion of INSPIRE UML models to RDF," [Unpublished report]. Geonovum, 2014.

[3] L. van den Brink, "RDF in INSPIRE - Open issues, tools, and implications," [Unpublished report]. Geonovum, 2014.

[4] S. Williams, "INSPIRE in RDF - Deliverable 1," [Unpublished report]. Epimorphics Ltd, 2014.

[5] S. Williams, "INSPIRE in RDF - Deliverable 2," [Unpublished report]. Epimorphics Ltd, 2014.

[6] C. Portele, "INSPIRE RDF vocabularies - methodology, options and examples," [Unpublished report]. Interactive Instruments GmbH, 2014.

[7] C. Portele, "INSPIRE RDF vocabularies - open issues," [Unpublished report]. Interactive Instruments GmbH, 2014.

[8] European Commission, "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)".

[9] European Commission, "ISA Action 1.17, a Reusable INSPIRE Reference Platform," [Online]. Available: http://ec.europa.eu/isa/actions/01-trusted-information-exchange/1-17action_en.htm.

[10] European Commission, "Decision No 922/2009/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 16 September 2009 on interoperability solutions for European public administrations (ISA)".

[11] European Commission, "Guidelines for the encoding of spatial data, v3.3," [Online]. Available: http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/D2.7_v3.3.pdf.

[12] European Commission, "Definition of Annex Themes and Scope, v3.0," [Online]. Available: http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/DataSpecifications/D2.3_Definition_of_Annex_Themes_and_scope_v3.0.pdf.

[13] N. Dalkey and O. Helmer, "An Experimental Application of the Delphi Method to the use of experts.," *Management Science,* pp. 458-467, April 1963.

[14] International Organization for Standardization, "ISO/DIS 19150-2 – Geographic information — Ontology — Part 2: Rules for developing ontologies in the Web Ontology Language (OWL)".

[15] International Organization for Standardization, "ISO/DIS 19103:2013 – Geographic Information — Conceptual Schema Language".

[16] International Organization for Standardization, "ISO/DIS 19109:2013 – Geographic information — Rules for application schema".

[17] W3C, "Best Practices for Publishing Linked Data," [Online]. Available: http://www.w3.org/TR/ld-bp/.

[18] European Commission, "INSPIRE code list register," [Online]. Available:

[18] http://inspire.ec.europa.eu/codelist/.

[19] A. Perego, "Re3gistry RDF format," European Commission - Joint Research Centre, 2014. [Online]. Available: https://ies-svn.jrc.ec.europa.eu/projects/registry-development/wiki/RDF_format. [Accessed 08 07 2015].

[20] S. Athanasiou, L. Bezati, G. Giannopoulos, K. Patoumpas and D. Skoutas, "GeoKnow - Making the Web an Exploratory for Geospatial Knowledge. Market and Research Overview.," 2013. [Online]. Available: http://svn.aksw.org/projects/GeoKnow/Public/D2.1.1_Market_and_Research_Overview.pdf.

[21] D. Brickley, "Basic Geo (WGS84 lat/long) Vocabulary," 2003. [Online]. Available: http://www.w3.org/2003/01/geo/.

[22] M. Perry and J. Herring, "OGC GeoSPARQL - A Geographic Query Language for RDF Data.," 2010. [Online]. Available: http://www.opengeospatial.org/standards/geosparql.

[23] A. Perego, M. Lutz and P. Archer, "ISA Programme Location Core Vocabulary. W3C.," [Online]. Available: http://www.w3.org/ns/locn.

[24] International Organization for Standardization, "ISO 19107:2003. Geographic information - Spatial schema.".

[25] International Organization for Standardization, "ISO 19108:2002 - Geographic information -- Temporal schema".

[26] International Organization for Standardization, "ISO 19111:2007 - Geographic information -Spatial referencing by coordinates".

[27] International Organization for Standardization, "ISO 19115:2003 - Geographic information - Metadata".

[28] International Organzation for Standardization, "ISO 19123:2005 - Geographic information -- Schema for coverage geometry and functions".

[29] International Organization for Standardization, "ISO 19156:2011 - Geographic information - Observations and measurements".

[30] R. Hodgson, P. J. Keller, J. Hodges and J. Spivak, "QUDT - Quantities, Units, Dimensions and Data Types Ontologies," [Online]. Available: http://qudt.org/.

[31] S. Cox, "An explicit OWL representation of ISO/OGC Observations and Measurements.," in *Proceedings of the 6th International Workshop on Semantic Sensor Networks, Vol-1063: 1-18.*, 2013.

[32] W3C, "PROV-O: The PROV Ontology," [Online]. Available: http://www.w3.org/TR/prov-o/.

[33] C. Atkinson and K. Kiko, "A Detailed Comparison of UML and OWL," 2008.

[34] European Commission, "Generic Conceptual Model," [Online]. Available: http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/D2.5_v3.4.pdf.

[35] C. Welty, R. Fikes and S. Makarios, "A Reusable Ontology for Fluents in OWL," *FOIS,* vol. 150, pp. 226-236, 2006.

[36] W3C, "Cool URIs for the Semantic Web," 2008. [Online]. Available: http://www.w3.org/TR/cooluris/.

[37] W3C, "Protocol for Web Description Resources (POWDER)," [Online]. Available: http://www.w3.org/2007/05/powder-s.

[38] W3C, "JSON-LD 1.0 - A JSON-based Serialization for Linked Data," [Online]. Available: http://www.w3.org/TR/json-ld/.

[39] "GeoJSON," [Online]. Available: http://geojson.org/.

[40] European Commission - The ISA programme, "DCAT application profile for data portals in Europe," [Online]. Available: https://joinup.ec.europa.eu/asset/dcat_application_profile/asset_release/dcat-application-profile-data-portals-europe-final.

[41] W3C, "Data Catalog Vocabulary (DCAT)," [Online]. Available: http://www.w3.org/TR/vocab-dcat/.

[42] "DCMI Metadata Terms," [Online]. Available: http://dublincore.org/documents/dcmi-terms/.

[43] W3C, "Describing Linked Datasets with the VoID Vocabulary," [Online]. Available: http://www.w3.org/TR/void/.

[44] W3C, "Linked Data Platform 1.0," [Online]. Available: http://www.w3.org/TR/ldp/.

[45] J. Zedlitz, Jörke, Jan and Luttenberger, Norbert, "From UML to OWL 2," in *Knowledge Technology*, Berlin, Heidelberg, Spinger, 2012, pp. 154-163.

[46] Gasevic, Dragan, Djuric, Dragan, Devedzic, Vladan and Damjanovi, Violeta, "Converting UML to OWL Ontologies," in *Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters*, New York, ACM, 2004, pp. 488-489.

ANNEX A – QUESTIONNAIRE: TRANSFORMING INSPIRE UML MODELS TO RDF

PURPOSE

This questionnaire has been created in the context of the study launched by ARE3NA, a study to see how a common Resource Description Framework (RDF) vocabulary can be developed for INSPIRE and what approaches can be taken for the governance of global Persistent Identifiers (PIDs) for INSPIRE and location-related activities, including URIs and DoIs.

Part of this work has involved a state-of-play of RDF and PIDs, aiming at the collection of all methodologies that are currently applied with regard to these topics.

However not all aspects were documented or details about the methodology were missing. This questionnaire aims at filling the blanks, and collect the view of the three experts that were assigned by the JRC.

AUDIENCE

This questionnaire is addressing Clemens Portele, Linda van den Brink and Stuart Williams who were assigned by the JRC to support with the development and documentation of a methodology for INSPIRE in RDF.

OBJECTIVE

The basic idea behind the questionnaire is to acquire the first thoughts of every expert on the methodology and based on that input establish a common baseline for the experiment. Try to answer each question briefly in max. 10 lines.

QUESTIONNAIRE

METHODOLOGY

M1 Describe in short the study or research you have already conducted related to the transformation of UML models to RDF vocabularies or refer to an existing summary?

Answer:

M2 Identify and describe the overall methodology that was used to conduct the study?

Answer:

M3 Which set of mapping rules have been applied? Were these mapping rules based on existing (draft) standards (ISO19150, OMG QVT, other….)?

Answer:

M4 Does the set of mapping rules reflect an open or closed-world view on the transformation? Did the methodology take into account the re-use of existing vocabularies? If yes, which ones?

Answer:

M5 Were there any other important assumptions made?

Answer:

M6 What were the most important advantages and drawbacks/issues of the methodology?

Answer:

M7 Would you consider the same approach for transforming INSPIRE models to RDF vocabularies? Please explain why (not).

Answer:

M8 Are you already aware of known issues related to the transformation of INSPIRE application schemas? Which one(s)?

Answer:

M9 Is it feasible to solve the abovementioned issues within the experiment or are there external constraints? Are there topics that you think should be excluded from the experiment?

Answer:

TOOLS

TO1 Which tools will you use to transform the UML model? Indicate whether they are open source of proprietary software? Please explain the main reasons why you have chosen for these tools?

Answer:

TO2 Which tools are available to transform an associated dataset? Indicate whether they are open source of proprietary software?

Answer:

TO3 Is it necessary to modify these tools to support transformation to INSPIRE OWL and RDF? What could/should be modified in view of the experiment?

Answer:

EXPERIMENT: PROPOSED THEMES

Below, a selection of INSPIRE themes has been made and assigned to each of the experts.

Clemens Portele	Linda Van den Brink	Stuart Williams
Land Cover	Buildings	Area Management Zones
Transport Networks	Statistical Units	Hydrography
Environmental Monitoring Facilities	Environmental Monitoring Facilities	Environmental Monitoring Facilities

E1 Do you agree with the proposed themes? If not, please mention why?

Answer:

TIMING

The table below proposes a timeline to keep the project on schedule.

	Experts	Contractor
01/04/2014	***Start of the experiment***	
07/04/2014		Send questionnaire
09/04/2014	completion of the questionnaire	
		Define baseline and scope
	-elaborate methodology -first tests on transforming UML model	

	Experts	Contractor
15/04/2014	***Expert meeting: compare methodologies and first findings***	
		Compare methodologies Create reporting template
18/04/2014	Revision own methodology based on outcome of expert meeting	
15-25/04/2014	Testing on INSPIRE themes	
25/04/2014	Reporting intermediate results of tests	
28-30/04/2014		Summarize results (UML models and preliminary transformed datasets Define first set of guidelines (common to all approaches)
30/04/2014	**Discussion on results**	
02/05/2014	Revise guidelines	
5-6/05/2014	***Webinars: Propose methodologies and guidelines to wider audience***	
		Process input of webinar
20/05/2014	Definition of a common methodology	
30/05/2014	Definition of recommendations and potential rules	

TI1 Do you agree with the proposed timing? Are there any bottlenecks?

Answer:

OTHER REMARKS

O1. Please let us know if you have any other questions or concerns?

Answer: