Implementing ModernStats Standards

Linked Open Metadata Design Guidelines

Version 1.0 – December 2016

Outline

1. Introduction

HLG has been jointly developing common models and vocabularies to prevent each organization from developing their own models and vocabularies for the same concepts, i.e. GSBPM [2], GAMSO [4], GSIM [1], CSPA [3] service descriptions, classifications of Statistical Activities and of Types of Big Data. Linked open metadata provides the subsequent step to prevent each organization from having to maintain and update their individual vocabularies, as this would be made available and managed in a centralized way. This not only reduces costs but also prevents discrepancies in structural and reference metadata and semantic heterogeneity.

Linked open metadata are also the key enablers of linked open data as they allow for accessing and sharing data across organizations. In open data settings, it allows for searching and integrating data from a large number of data sources and combined with the semantic web, to endless possibilities of associating it with (contextual) information available from outside structured databases.

The statistical community develops metadata standard of good quality, but these standards are rarely available internally or for other users in open and machine-actionable formats. The linked data format is especially relevant for metadata dissemination, because:

- global uniform naming and addressing is crucial for structural metadata like classifications, code lists, cube dimensions, etc. so that there is a unique and well-known reference always accessible by standard mechanisms;
- there exist several linked data standard models or vocabularies which are dedicated to metadata, i.e. the Dublin Core [9], SKOS/XKOS [6][8], PROV [5] or DCAT [7], which allows for good discoverability and referencing by external consumers or publishers;
- glossaries and vocabularies expressed as linked data can be connected to or from other concept schemes available on the Internet in this format, like Eurovoc or the Library of Congress Subject Headings (LCSH); the linked data format enlarges the base of users of statistical data by relying on standards also coming from outside the statistical community.

Moreover, there is a clear demand from the academic world for reference linked statistical metadata (concepts, code lists, etc.): this was strongly expressed at the recent Semantic Statistics (SemStats) workshop, for example.

Beyond dissemination, formalizing metadata with linked data standards guarantees a level of coherence, interoperability and adaptability that other models do not offer. For example, semantic descriptions of CSPA services can be formally linked to the GSBPM sub-process in which they operate and to GSIM objects that form their inputs and outputs. It is also in line with the active metadata paradigm, since linked data are easy to consume automatically and to integrate into metadata-driven processes. Thus, linked metadata are also a promising tool for achieving a better consistency and integration, within each organization and at the global level, of the statistical production.

A growing number of statistical institutes have understood that problematic and have started to invest in linked metadata. The benefits that can be achieved are several and various, including:

- Easiness of data access, in terms of fetching the data and combining data in whatever format/concept scheme and machine readability;
- Quality tracking by comparing, reproducing, finding inconsistencies, correct interpretation;
- Integration by linking data, avoiding unnecessary duplication of data, semantic homogeneity (same concept/variable, same name), searching multiple database at once for combining data.

2. Implementing ModernStats Standard Project

The IMS (Implementing ModernStats Standards) [10] project has been launched at the HLG Workshop in November 2015.

The main objective of the project is to demonstrate the usefulness of linked metadata for the statistical community and to acquire hands-on experience in that field.

It is proposed to fulfill this objective by constructing two concrete examples of linked metadata-based information systems: one aimed at improving the way that we disseminate core structural metadata, the other at supporting the advancement of the HLG vision by creating an harmonized and semantically enhanced information system grouping the main CSPA models and standards in a coherent and machine-actionable form.

Each of those systems constitutes a deliverable, and thus a secondary objective in itself, of the project. Other important deliverables describe the lessons learned, best practices, and a sustainability plan for the projects outcomes beyond the end of 2016.

The project is divided in three work packages, namely: (i) WP1 concerning classifications and concepts; (ii) WP2 concerning models; and (iii) WP3 relating to maturity model. Both WP1 and WP2 are about linked metadata.

The first objective of WP1 objective is the creation of Linked Metadata sets for: (i) a set of selected classifications namely: ATECO (Italian classification of Economical Activities) [12], NAF (French classification of Economical activities) [13], CPA (European classification of Products by Activity) [17], CPC (United Nations Central Product Classification) [15] and ISIC (International Standard Industrial Classification of All Economic Activities) [14] and for a selected SDMX [16] code list, that is the Measure Unit code list. The second objective of WP1 is the implementation or configuration of software artifacts for navigating classifications and for visualizing the related ontologies.

The first objective of WP2 is the creation of Linked Metadata sets for the following general models: GSIM, GSBPM, CSPA and EARF (Enterprise Architecture Reference Framework) [11] Building Blocks. The second objective of WP2 is the implementation or configuration of software artifacts for navigating models and for visualizing the related ontologies.

This document will describe the results of the work concerning WP1 and WP2. Such results have been achieved by a multi-national project group[1] with INSEE (Franck Cotton) and Istat (Monica Scannapieco) having a coordination role.

3. Use Case Definition

In this section, some use cases [18] are presented to highlight the opportunities provided by an RDF representation of classifications and metadata.

Use Case 1: Comparing National Classification Refinements. This use case considers the Italian and French refinements of NACE classification, respectively ATECO version 2007 and NAF and compares these classifications pointing out the different approaches for refinements, the similarities and the differences between the two classifications and finally, the possibility of realizing federated navigation.

Use Case 2: Seamless Metadata Access. This use case shows the possibility of performing the cross-classification browsing, as for example the navigation between NACE and CPA; the possibility of querying on meta-ontologies, e.g. PROV ontology; and finally the possibility of navigating the project catalogue (DCAT) across semantic assets (ADMS [**19**]).

Use Case 3: Combing SDMX and RDF. This use case shows how to generate RDF data starting from a SDMX code-list and the new possibility to query SDMX code-list expressed in RDF.

Use Case 4: Linking GSIM and GSBPM. This use case shows how to query GSIM-RDF with the same results as in clickable GSIM; how to cross-navigate between GSIM and GSBPM; and how to check the mutual-coherence between GSIM and GSBPM.

Use Case 5: Linking CSPA Services and EARF Building Blocks. This use case shows how to navigate between CSPA services and EARF building blocks and the possibility of understanding which building blocks implement a specific CSPA service just making a query.

Use Case 6: Defining a CSPA service. This use case shows how to define or specify a CSPA service by using GSIM-RDF and GSBPM-RDF ontologies pointing out the enhanced functionalities that comes for free with these RDF representations.

Use Case 7: Documenting changes to the version of an object/artifact. This use case shows how to record the changes between the versions of an artifact (i.e. SDMX artifacts) and how this can provide an undo/redo facility and an easy comparison functionality between versions.

4. Naming Policy

In this section, the naming policy [20] adopted in the project is presented.

[1] Raffaella Aracri, Mauro Bruno, Franck Cotton, Eric Debonnel, Taeke Gjaltema, Dennis Grofils, Hans van Hoof, Olivier Levitt, Enrico Orsini, Andrea Pagano, Jean-Baptiste Rudant, Monica Scannapieco, Romain Tailhurat, Laura Tosco and Luca Valentino

The repository contains different types of data specifically: models (GSIM, GSBPM, GAMSO, etc.), information about CSPA services, glossaries, codes, classifications and correspondence tables, as well as metadata about the main content, namely: provenance, cataloging and publication information. The RDF [21] vocabularies used depend on the type of data; that is: models are mostly expressed in OWL [22], glossaries and classifications are represented in SKOS/XKOS and metadata use the standard vocabularies ADMS, DCAT, VoID and PROV.

Given the content of the repository, we have to identify different types of resources:

- For the model part: vocabulary or ontology elements, namely ontologies, classes and properties, datatypes, individuals;
- For the classification part: concept schemes, classification levels, concepts, notes, correspondence tables, concept associations, etc.;
- For the metadata: ADMS catalogs, assets and asset distributions, VoID datasets, PROV entities, activities and agents, named graphs, etc.

The data sources for the project come from different producers: Eurostat, the UNECE, the UNSC, national offices, the SDMX sponsors group, etc. Ideally, each resource should be identified by a URI based on a domain controlled by its publisher: for example the GSIM should use URIs in the http://www.unece.org domain, the CPC should use URIs in the http://www.unsd.org and the NACE should use http://ec.europa.eu/eurostat. However, it is clearly impossible within the timeframe of the project to design a naming policy for each of these actors, specific to their data, and have it validated by them.

This is why it is suggested to define a naming policy based on a "neutral" domain name, for example stamina-project.org. We suggest to divide the root domain name according to the main type of data: */models*, */concepts* (for glossaries and classifications) and */metadata*.

Under */concepts*, we subdivide by an identifier of the major version of a classification (isicr31, nacer2, etc.) or, for the correspondences, by the combination of the major versions of the classifications that the table compares: isicr4-cpcv21 for example. The source classification scheme should be first.

Similarly, the */metadata* path root will be further refined according to the vocabulary, for example */adms* or*/prov*.

Under */models*, it is useful to distinguish between the identification of the individuals (a given GSBP phase or GSIM object) and the identification of the vocabulary terms defined to represent the models themselves (the OWL class corresponding to a GSBPM phase or to a GSIM object). For individuals, we subdivide further according to the name of model: */gsim*, */gsbmp*, */gamso*, etc. OWL vocabularies usually use hash-namespaces, so our OWL objects will be in the *http://stamina-project.org/models/def#* namespace.

Inside the given context, the default pattern for identifying a given resource will be: */{resource-type}/{resource-identifier}*, except for OWL artifacts whose URI will be *http://stamina-*

project.org/models/def#{name-of-artifacts}. The {resource-identifier} can be any local identifier for the resource, for example the item code for a classification item, a version number or a publication date for a concept description, etc. When there is only one resource of a given type within the context, the /{resource-identifier} path element can be omitted.

When a resource strongly depends on another, this can be represented in the path hierarchy, for example:

- An explanatory note for a classification item exists only in the context of this item, so (if there is a necessity to represent it as a resource and not simply as a RDF literal) it will have an URI like *http://stamina-project/concepts/nacer2/class/51.22/inclusion-note*.
- Likewise, an ADMS asset distribution will be identified by extending the asset URI. On the contrary, the asset URI does not extend the catalog URI since an asset may be included in several catalogs.

To better explain the adopted naming policy, some examples follow.

For identifying section B of the ISIC Rev.3.1 we have the following URI components:

Element	Value
Authority	http://stamina-project.org
Path element for classifications	/concepts
Path for ISIC Rev.3.1	/isicr31
Resource type	/section
Resource identifier	/B

Table 1. Section B of the ISIC Rev. 3.1 URI components

The URI is thus *http://stamina-project/concepts/isicr31/section/B*.

For sub-process 3.1 of the GSBPM, we have:

Element	Value
Authority	http://stamina-project.org
Path element for models	/models
Path for GSBPM	/gsbpm
Resource type	/sub-process
Resource identifier	/3.1

Table 2. Sub-process 3.1 of the GSBPM URI components

The URI is thus http://stamina-project/models/gsbpm/sub-process/3.1.

Table 3 shows additional examples on the naming policy adopted.

Resource	URI
ADMS asset for CPC Ver.2.1	http://stamina-project.org/meta/adms/asset/cpcv21
VoID dataset of the GSBPM	http://stamina-project.org/meta/void/dataset/gsbpm

individuals	
VoID dataset of the GSBPM ontology	http://stamina-project.org/meta/void/dataset/gsbpm-def

Table 3. Additional examples on naming policy

5. Guidelines for ADMS and DCAT-AP meta-ontologies

In this section, some best practice on the usage of ADMS and DCAT-AP meta-ontologies for Classification catalogue are described [23].

DCAT-AP (DCAT- Application Profile)[24] is a specification that re-uses terms from one or more base standards, adding more specificity by identifying mandatory, recommended and optional elements to be used for a particular application of DCAT (Data Catalogue Vocabulary). For our statistical artifact, our application profile DCAT-AP is defined by the following concepts:

- **Agent**: entity that is associated with Catalogues and/or Datasets. If the Agent is an organization, the use of the Organization Ontology is recommended;
- **Catalogue**: a catalogue or repository that hosts the Datasets being described;
- **Dataset**: a conceptual entity that represents the information published;
- **Category**: a subject of a Dataset;
- **Category Schema**: a concept collection (e.g. controlled vocabulary) in which the Category is defined;
- **Distribution**: a physical embodiment of the Dataset in a particular format.
- **Location**: a spatial region or named place. It can be represented using a controlled vocabulary or with geographic coordinates. In the latter case, the use of the Core Location Vocabulary47 is recommended.
- **Linguistic System**: a system of signs, symbols, sounds, gestures, or rules used in communication, e.g. a language;
- **Period Of Time**: an interval of time that is named or defined by its start and end dates;
- **Publisher Type**: A type of organization that acts as a publisher;
- **Identifier**: an identifier in a particular context, consisting of the string that is the identifier; an optional identifier for the identifier scheme; an optional identifier for the version of the identifier scheme; an optional identifier for the agency that manages the identifier scheme.
- **License Document**: a legal document giving official permission to do something with a resource;
- **vCard**: a description following the vCard specification, e.g. to provide telephone number and e-mail address for a contact point.
- **Media Type**: a media type or extent, e.g. the format of a computer file.
- **Literal**: a literal value such as a string or integer; literals may be typed, e.g. as a date according to xsd:date. Literals that contain human-readable text have an optional language tag as defined by BCP 4746.
- **Resource**: anything described by RDF.

ADMS is a profile of DCAT related to semantic assets. The specific concepts useful to describe the statistical semantic assets are:

- **Interoperability Level**: the interoperability level (e.g. legal, organizational, political etc.) of the Asset, linked using adms:interoperabilityLevel. The interoperability level may be taken from a list of levels such as that of the European Interoperability Framework;
- **Asset Type**: the classification of an Asset according to a controlled vocabulary. The controlled vocabulary has to be defined so that it represents statistical semantic assets (classification, correspondence tables among classifications, etc.);
- **Publisher**: the organization making a Repository, Asset or Distribution available;
- **Representation Technique:** the machine-readable language in which a Distribution is expressed, this is more fine-grained than file format, i.e. "Word 2003".

Table 4 shows the correspondence between ADMS Concepts and DCAT-AP Concepts.

ADMS Concepts	DCAT-AP Concepts
adms:AssetRepository	dcat:Catalogue
adms:Asset	dcat:Asset
adms:AssetDistribution	dcat:Distribution
adms:ContactInformation	dcat:VCard
adms:GeographicalCoverage	dcat:Location
adms:Identifier	dcat:Identifier
adms:Language	dact:LinguisticSystem
adms:License	dcat:LicenseDocument
adms:PeriodOfTime	dact:PeriodOfTime
adms:Theme	dcat:Category
adms:ThemeTaxonomy	dcat:CategorySchema

Table 4. Correspondence between ADMS and DCAT-AP Concepts

6. Guidelines for XKOS classification

This section describes some best practices for representing statistical classifications as XKOS. These are only examples: a more complete list is maintained in [25]. For each proposed rule, a SPARQL query is provided when appropriate: for a conformant RDF store, the query should return no result.

Classifications and classifications schemes

Rule 1. All classification schemes MUST have a ***skos:notation*** property which value is the short name of the classification scheme with no language tag.

Associated query:

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
```

```
SELECT ?s {
  ?s rdf:type skos:ConceptScheme .
  MINUS {
    SELECT ?s {
      ?s rdf:type skos:ConceptScheme .
        ?s skos:notation ?code .
    }
  }
}
```

Rule 2. All classification schemes MUST have a ***skos:prefLabel*** property which value is the complete name of the classification scheme in English. Names in other languages may be provided with the same property. All names must have a language tag.

Associated queries:

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>

SELECT ?s {
  ?s rdf:type skos:ConceptScheme .
  MINUS {
    SELECT ?s {
      ?s rdf:type skos:ConceptScheme .
      ?s skos:prefLabel ?label .
    }
  }
}
```

Rule 3. All classification schemes MAY have additional labels represented by values of the **skos:altLabel** property. The SKOS integrity rules MUST be applied. The *XKOS specification*[http://rdf-vocabulary.ddialliance.org/xkos.html#add-labels] gives rules regarding the representation of fixed-length labels.

Rule 4. All classification schemes SHOULD have a ***dc:description*** property which value is the short descriptive text about the classification scheme in English. Description in other languages may be provided with the same property. All descriptive texts should have a language tag.

Associated query:

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
PREFIX dc: <http://purl.org/dc/elements/1.1/>

SELECT ?s {
  ?s rdf:type skos:ConceptScheme .
  MINUS {
    SELECT ?s {
      ?s rdf:type skos:ConceptScheme .
      ?s dc:description ?description.
      FILTER (LANG(?description)="en")
    }
```

```
  }
}
```

Rule 5. All classification schemes MUST have a ***dcterms:issued*** property which value is the publication date of the classification scheme with datatype *xsd:date*.

Associated query:

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
PREFIX dcterms:<http://purl.org/dc/terms/>

SELECT ?s {
  ?s rdf:type skos:ConceptScheme .
  MINUS {
    SELECT ?s {
      ?s rdf:type skos:ConceptScheme .
      ?s dcterms:issued ?issued .
    }
  }
}
```

Rule 6. All classification schemes SHOULD have a ***dcterms:modified*** property which value is the last modification date of the of the classification scheme with datatype *xsd:date*.

Associated query:

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
PREFIX dcterms:<http://purl.org/dc/terms/>

SELECT ?s {
  ?s rdf:type skos:ConceptScheme .
  MINUS {
    SELECT ?s {
      ?s rdf:type skos:ConceptScheme .
      ?s dcterms:modified ?modified .
    }
  }
}
```

Rule 7. All classification schemes should have a ***skos:scopeNote*** property which value is a resource of type ***xkos:ExplanatoryNote***. The explanatory note must have a ***xkos:plainText*** property which value is a long descriptive text about the classification scheme in English, with a language tag set at '@en'. Long descriptives in other languages may be provided: for each language a dedicated ***xkos:ExplanatoryNote*** resource will be created, with a ***xkos:plainText*** string bearing the corresponding language tag.

7. Considerations on the role of ontologies with respect to statistical models GSIM, GSBPM and CSPA

Within the IMS project, OWL ontologies for important and widespread statistical models as GSIM, GSBPM and CSPA have been proposed.
Briefly, OWL [**22**] is a knowledge representation language with the following basic notions:

- **Axioms**: the basic assumptions that an OWL ontology expresses;
- **Entities**: elements used to refer to real-world objects and are all atomic constituents of statements: objects, categories, or relations;
- **Expressions**: combinations of entities that form complex descriptions from basic ones.

In OWL objects are denoted as **individuals**, categories as **classes**, and relations as **properties**. Properties are further subdivided into **Object Properties** that relate objects to objects, and **Datatype Properties** that assign data values to objects.

7.1. GSIM Ontology

In this section, the GISM Ontology [29] is presented. GSIM (Generic Statistical Information Model) is a framework for statistical metadata promoted by the UNECE and internationally endorsed by statistical organizations, which enables generic descriptions of the definition, management, and use of data and metadata throughout the statistical production process.
Starting from the UML representation of GSIM, the UML classes have been mapped into OWL Classes, relations into Object Properties, and attributes into Datatype Properties. The subclasses of UML schemas have been modeled with the OWL object property called *subClassOf* that relates the main class with its subclasses.
UML and OWL are two languages with substantial differences thus there is not always a direct correspondence between their respective elements. As an example, an abstract class in UML does not have a direct correspondent with any OWL class because the notion of what belongs to a class in OWL is more fluid. Additional statements on Classes, ObjectProperties, and DataProperties (e.g. objectIntersectionOf, disjointClasses) allow representing the domain of interest more fully.
Two different approaches have been followed to realize the GSIM ontology.
In the **first approach**, the design of the ontology started from the specification document of the GSIM model. From this, we deduced which concepts are to be represented in the ontology, which properties link the concepts to each other, and what the data properties of the concepts are.
GSIM concepts are grouped in five main topics areas:

I. Base: contains the basic concepts, as Agent, Agent Role and Identifiable Artifact;
II. Business: contains concepts defined to capture the designs and plans of statistical programs, and the processes undertaken to deliver those programs. This includes the identification of a Statistical Need, the Business Processes that compose the Statistical Program, and the evaluations of them;
III. Concepts: contains concepts to define the meaning of data, providing an understanding of what the data are measuring;
IV. Exchange: contain concepts defined to catalogue the information that comes in and out of a statistical organization via Exchange Channels. It includes classes that describe the collection and dissemination of information.
V. Structures: contain concepts to describe and define the terms used in relation to structures for organizing data.

In the ontology, we maintained this structure by defining five Classes with those names; in each Class we defined sub-classes for each concept contained in the topic. I

In the GSIM specification document, the relationships between concepts often have the same name as the attributes of the concepts, to distinguish the properties we adopted the following notation:

Property Name || first letters of Domain || first letters of Range

The notation is valid for Object Properties, while in the case of Data Properties the "first letters of Range" are missing.
We obtained an ontology with the characteristics reported in Table 5 with a DL expressivity *ALCIRQ(D)* according to Protégé.

Concept	Number
Classes	134
Object Properties	203
Data Properties	383
subClassOf Axioms	117
equivalentClass Axioms	14
ObjectPropertyDomain Axioms	204
ObjectPropertyRange Axiom	240
FunctionalDataProperty Axioms	378
DataPropertyDomain Axioms	378
DataPropertyRange Axioms	382

Table 5.Number of classes, properties, and axioms defined in the GSIM ontology

In the **second approach**, we translate directly and automatically GSIM from Enterprise Architect, in which GSIM-UML is already fully specified in machine-actionable format, to RDF.
The method used to transform UML to RDF is an ad hoc XSL transformation, which is direct, efficient, and simple, but depends on the way UML is used. We could do it because the GSIM specification is coherent and well-written. Other approaches could have been used, for example Model Driven Architecture based solutions or the Ontology Definition Metamodel, a standard defined by the Object Management Group (OMG) and implemented in EA, which extends UML with additional modeling notations to allow representation in OWL. Since GSIM uses only simple UML constructs, we felt that those approaches were too complex in this specific case.
The procedure is composed of the following steps:

1. **Export the UML description from EA** (version 10) **to a file in the XMI 2.1 format**, which is the standard created by the OMG for expressing UML in XML. We then used an existing XSL transformation (XSLT) provided by the UNECE to extract the relevant information in a simpler and more convenient XML format. From there, we wrote the XSL transformation to produce RDF/XML.
2. **From XML to RDF for classes and packages.** The UML and RDF concepts for classes are very close, so no adaptation was needed. Basically, we did a one-to-one mapping. However, since the UML model was divided into packages corresponding to the different GSIM groups described above, we adopted the same method as in the previous approach and made all classes of a package sub-classes of a class representing the package (Structure, Business, etc.).
3. **From XML to RDF for properties**. UML attributes and relationships are both represented by, and transformed into, RDF properties. RDF distinguishes between annotation, data, and object properties, depending on the type of their range (we did not use annotation properties). Most parts of UML attributes and relationships are easy to transform into RDF.

Only the nature of the relationships between classes (associations, compositions, etc.) was not used. The approach described in 4.1 proved that this was not necessary, and we felt the complexity added in taking them into account was not worth it. The domain of a property (the class described by the attribute or relationship) is always known, by construction of the UML. Cardinality restrictions are specified in UML the same way as in OWL, even if zero-minimum and n-maximum cardinality restrictions need not be specified in OWL. The range of a relationship (the class it points to) is found by a one-to-one mapping, but the range of an attribute cannot always be kept as is. If the original attribute range cannot be mapped to a known class or type (binary to xs:boolean for instance), we transformed it to a xs:string. Among the UML attributes are also three types of comments (Definition, Explanatory text, and Synonyms), which we transformed into corresponding RDF properties.

4. **Difficulty to find a name.** The name part of UML attributes and relationships is much harder to transform into RDF, since in UML the name is a tag, whereas in RDF it must uniquely identify the property. The conversion between UML and RDF is not straightforward, because UML attributes are parts of one class and UML relationships exist only to connect classes, whereas RDF properties are first-level objects by themselves. That is why we had to build a clear algorithm to construct unique names for properties.
   A simple choice would have been to create one property for each attribute and each relationship, as in the previous approach, but this raises a problem by creating redundant properties. A good example is the "name" attribute that many UML classes have: all of them link the property to a character string, and most of them have the same cardinality restriction (at most one name is possible). Creating only one property for those cases is more desirable, because someone wanting to know the label of a class would only have to query its "name", but there is a risk of merging relationships or attributes where it is not appropriate. Those accidents produce weird property domains or ranges that are easily spotted in the resulting ontology. They can thus be reported back to the GSIM designers in order to be fixed directly in the UML model.

5. **Detailed process to obtain RDF properties.** We adopted the following algorithm to decide whether two or more attributes or relationships of the same name can be grouped into one RDF property. Apart from the name, the decision criteria are range, domain, range cardinality restrictions, and domain cardinality restrictions. When several attributes have to be merged into one property, different non-empty comments are merged by concatenating the domain of the attribute and the comment. To merge properties together, the following decision tree is used:
   Is original attribute/link name unique?
   - [Yes] one property
   - [No] is the original name with source name and destination name unique?
     - [Yes] one property
     - [No] is the combination original name / range name unique?
       - [Yes] one property
       - [No] is the cardinality restriction on range unique?
         - [Yes] one property with a union of domains
         - [No] since in the original file, the triple property range domain is unique, we build for each triple a property name including the names of property range domain.

7.2. GSBPM Ontology

Expressing the GSBPM as OWL is quite straightforward: the model is rather a taxonomy of statistical activities than a business model. This observation leads to the idea of reusing two important models available for the semantic web: SKOS and PROV-O, both W3C recommendations. SKOS is a model for expressing the structure and content of controlled vocabularies, thesauri, taxonomies and other concept schemes. PROV-O is an OWL expression of a model about provenance metadata ("Provenance is information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness") [30].

The connection between PROV and the GSBPM is the notion of statistical activity, which can be seen as a special case of the general concept of activity defined in PROV. Consequently, we define (the prefix declarations are omitted for brevity):

```
gsbpm:StatisticalProductionActivity
    a rdfs:Class, owl:Class ;
    rdfs:label "Statistical production activity"@en ;
    rdfs:subClassOf prov:Activity .
```

This class is further specialized into *gsbpm:Phase* and *gsbpm:SubProcess*, which are also defined as sub-classes of *skos:Concept*, so that the actual instances (all GSBPM classes and sub-processes) can be organized as a *skos:ConceptScheme* (the GSBPM itself) using the usual SKOS properties. In particular, the hierarchies between classes and sub-processes is represented with standard *skos:broader*/*skos:narrower* properties.

With this simple modeling, we can benefit from all the possibilities offered by SKOS and PROV-O: add labels and notes to our GSBPM components, attach provenance information to our statistical activities, etc. This is an illustration of how easy reusing is in LOD and how rewarding it is.

7.3. CSPA Ontology

The conception of the CSPA ontology is detailed in [31]. The work took into account the CSPA specification document, but also the existing examples of CSPA service documentations provided by various organizations.

In the CSPA specification, two areas were more specifically selected: the levels of service documentation (service definition, service specification and service implementation description), with the different properties defined at each level, and the CSPA roles which are defined along the service lifecycle (Assembler, Builder, Configurer, Designer, etc.).

The CSPA specification defines different properties for the services, many unique to one level of documentation. For a better structuring of the ontology, we grouped those properties into 8 topics:

Additionally, we realized from the existing examples that current implementers of CSPA often want to specify two slightly different things, namely packages and functions. One service may actually be a bundle of functions, maybe accessible via different protocols. Thus we needed to describe not only the bundle of functions as one entity but also each and every function in the bundle. As a result, we introduced a distinction between functions and packages, even if the current CSPA specification does not make this distinction clearly. This distinction was referred to as "Service granularity", which is probably not the most appropriate term.

On the whole, we have distinguished in CSPA three main semantic axes related to the statistical services: levels of documentation, service granularity and property topics. These axes organize the global structure of the ontology for the part dealing with service documentation.

A (minimalistic) RDF instance for a given service could look like:

```
example:ErrorCorrection {
  service:ErrorCorrection a cspa:package;
  cspa:label "Error Correction";
  cspa:hasPackageDefinition [ a cspa:ServiceDefinition;
  cspa:aimsAt [ a cspa:BusinessFunction ;
      cspa:description "This Statistical Service corrects erroneous values in a record";
      cspa:outcomes "A consistent repair of records";
      cspa:gsbpmSubProcess igsbpm:5.4;
      cspa:restrictions "None"
      ] ;

  cspa:definitionHasInput [ a cspa:DefinitionInput ;
      cspa:gsimInput gsim:UnitType;
      cspa:gsimInput gsim:UnitDataSet;
      cspa:gsimInput gsim:UnitDataStructure
      ] ;

  cspa:definitionHasOutput [ a cspa:DefinitionOutput ;
  cspa:gsimOutput gsim:UnitDataStructure
  ] .
}
```

For the part of CSPA dealing with organization roles in the service lifecycle, we simply rely on the ORG ontology [32], which again allows us to leverage all the work done around this ontology.

7.4. Open Issues on Statistical Models Ontologies

In the previous sections we highlighted that:

- GSBPM is a framework for classifying *business processes* needed to produce Official statistics;
- according to GSIM a *business process* performs *business functions*;
- CSPA defines a framework for describing *services* at different levels of abstraction, namely: definition, specification and implementation. In a CSPA service definition, it is included the concept of *business function*.

The question we want to address [**26**] is:

> *How do we position in a coherent view the concepts of business process, business function and service* (as defined in the standards above)?

According to TOGAF, at business level the relationship between the above concepts is as described in Figure 1.

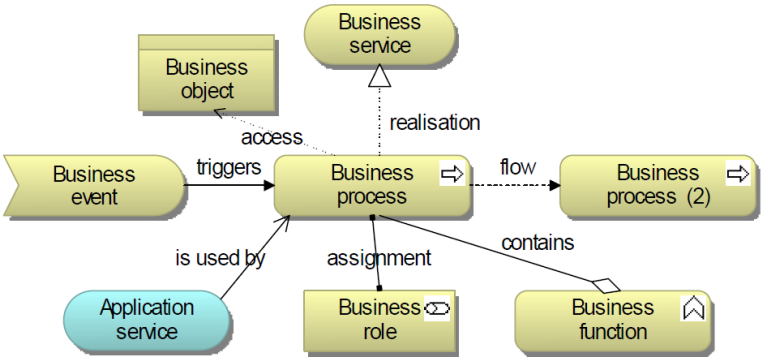

Figure 1. Relationship between Business concepts as in TOGAF

In particular:

- Business Services are “realized” by Business Processes;
- Business functions “contain” Business Processes.

The GSIM view of these concepts, as shown in Figure 2, is:

- Business Processes “uses” Business Services;
- Business functions are “performed” by Business processes;
- Business services “deliver” Business functions.

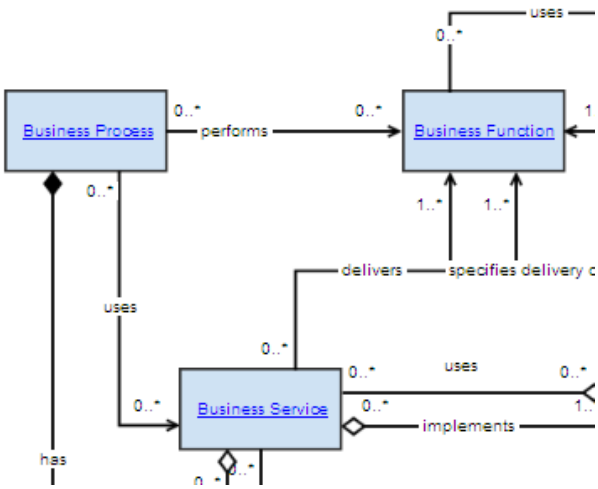

Figure 2. Excerpt of GSIM UML diagram

In CSPA specification, a statistical service definition includes “the business function that it performs”, i.e.:

- Business function is “performed” by a Business service.

Some open issues still need to be addressed as for example:

- A Business function is “performed” by: (i) only a business process (GSIM is right!), (ii) only a business service (CSPA is right), (iii) by both of them (GSIM and CSPA are incomplete)?
- A Business Process “uses” a Business Service (realized elsewhere) (GSIM is right!) or a Business process is “realized” by Business Services (TOGAF is right!)?

8. Overall Results

In this section we briefly present the results of the IMS projects. In more details, in Section 8.1 we present the artifacts we uploaded in the Stardog Environment; in Section 8.2 we describe the Classification Explorer we developed to navigate the classifications published; and finally, in Section 8.3 we describe the Model Explorer we developed to navigate the statistical models produced in OWL.

8.1. Stardog Environment

We created a Stardog (Sandbox) environment [27], to upload the artifacts produced in the project. Specifically, we uploaded the following artifacts:

- **SDMX measure_unit code list**: the SDMX codelist related to the unit of measures;
- **UN Classification**: ISIC and CPC;
- **Eurostat classifications**: NACE and CPA;
- **National classifications**: NAICS, Ateco, NAF and CPF, UK SIC and Dutch SBI;
- **Vocabularies**: GSIM, CSPA, GSBPM.

The main RDF vocabulary used for representing classifications is XKOS, published by the DDI Alliance. XKOS (eXtended Knowledge Organization System) extends the SKOS (Simple Knowledge Organization System) vocabulary, extending it for representing statistical classifications.

Follows a brief description of each artifact and of the corresponding creation process.

SDMX measure_unit code list.

Starting from the SDMX code list “Measure unit” downloadable from the Istat Single Exit Point with the following settings: “Codelist=IT1:CL_UNIT_MEASURE(1.2)"; agencyID="IT1"; version="1.2"; isFinal="true"”.

To model the content of the classification we used the SKOS ontology and its extension XKOS. Furthermore, to certify the provenance of the data, we used the PROV ontology framework; the activity of publishing the codelist “CL_Unit_MeasureV1.2” was carried out by two different actors, namely: (i) ISTAT_UfficioGestioneOntologie, that is the creator of the classification, and (ii) UNECE that is the publisher. We also enriched the metadata of the classification using the DCAT and

ADMS ontologies. As already described, DCAT ontology allows to insert the classification as dataset in our data catalogue and to decouple the abstract entity notion of dataset from its actual implementation. The ADMS ontology allows to specify and remark that the classification is a semantic asset, since it can be effectively used as integration element between different data, thus enabling semantic interoperability.
Figure 3 shows a graphical representation of the Unit_measure codelist expressed in OWL. In this representation it is possible to observe that: (i) SKOS allows to express that CL_UNIT_MEASUREV1.2 is a ConceptSchema; (ii) ADMS allows to express that CL_UNIT_MEASUREV1.2 is also a SemanticAsset; (iii) DCAT allows to express that CL_UNIT_MEASUREV1.2 is also a Dataset, and finally (iv) SDMX ontology allows to express that CL_UNIT_MEASUREV1.2 is also a CodeList.

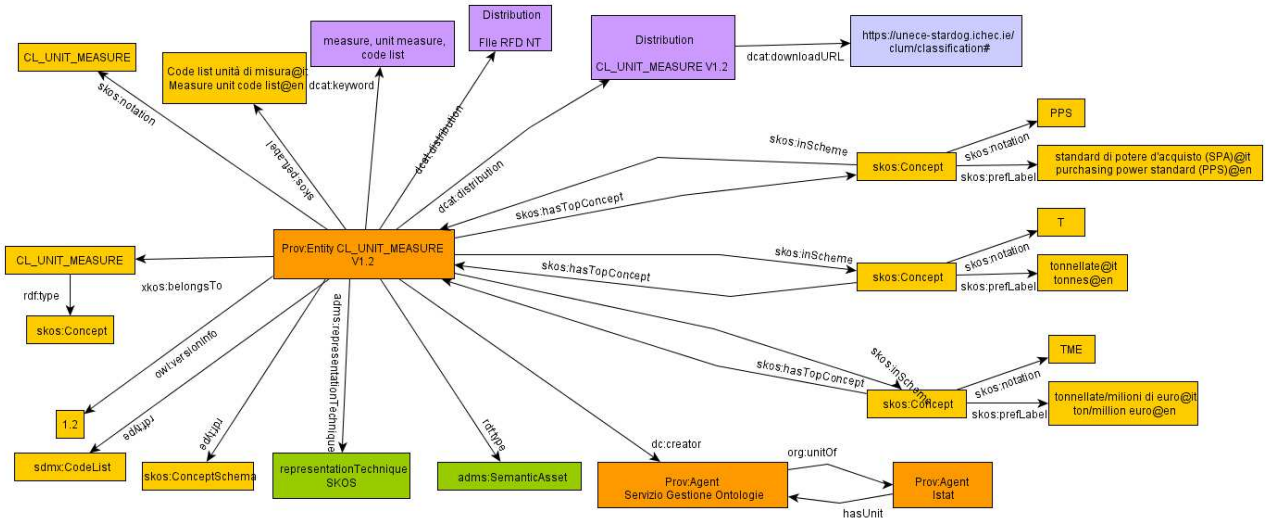

Figure 3. Graphical representation of Unit_measure codelist in OWL

UN-level classifications: ISIC and CPC.

ISIC (International Standard Industrial Classification of All Economic Activities) is the classification of the economic activities at the UN (United Nations) level. We produced in SKOS the last two versions of ISIC, namely revisions 3.1 and 4, as well as the historical correspondence table between them.

CPC (United Nations Central Product Classification) is also a classification at UN level; its SKOS representation of version 2.1 has been uploaded in Stardog.

In Stardog, we also uploaded the correspondences between the latest versions of ISIC and CPC (ISIC Rev.4 and CPC Ver.2.1 at the time of writing). Also, in order to provide use cases related to the evolution in time of classifications, it was decided to include the previous versions of both classifications, as well as the associated historical correspondences. Because of the correspondence structure between ISIC and CPC, this implies in fact to include in the system the last three versions of the CPC, which align with the last two versions of the ISIC.

The authoritative source for the UNSD classifications is the UNSD classification registry, and in particular the download page at http://unstats.un.org/unsd/cr/registry/regdnld.asp. The information is available in several formats and languages. Only PDF, MS Access and online HTML

have the explanatory notes, and only Access and HTML have the latest corrections. Additional sources provided by the UNSD include, amongst others, French and Spanish labels and correspondence tables (which are only available in English).

Eurostat classifications: NACE and CPA.

The NACE (Statistical Classification of Economic Activities in the European Community) and CPA (Statistical Classification of Products by Activity in the European Community) are the central classifications of economic activities and products in the European Statistical System. They are consequently included in the project perimeter. More precisely, the Stardog triple-store contain the last two versions of both classifications (NACE Rev. 1.1 and Rev. 2, and CPA Ver. 2008 and Ver. 2.1), as well as the historical correspondences between the two revisions of NACE and between the two versions of CPA, and the correspondences between NACE Rev. 2 and the two versions of CPA. Additionally, the correspondence between ISIC Rev.4 and NACE Rev.2 is also included.

The authoritative source for the Eurostat classifications is RAMON, Eurostat's Classification Server. The information is generally available in HTML, CSV and XML. The latter seems to be preferable for the main files giving the structure, labels and notes, whereas CSV can be used for simpler files like correspondence tables.

National classifications: NAICS, Ateco, NAF and CPF, SIC, SBI.

The NACE International Classifications of Economic Activities and Products is in general refined or adapted in each country in order to fit the local needs. In other cases, local classifications may have specific structures, but are linked to UNSD classifications by correspondence tables. We include here different examples of national classifications:

- NAICS 2012: North-american classification (US version), linked to ISIC;
- Ateco 2007: Italian version of the NACE;
- NAF rév. 2: French version of the NACE;
- CPF rév. 2.1: French version of the CPA;
- SIC 2007: UK version of the NACE;
- SBI 2008: Dutch version of the NACE.

The authoritative sources for national classifications is generally the country's NSI (National Statistical Institute). For NAICS, we used the publication made by the US Census Bureau; Ateco has been downloaded from Istat's web site, SIC and SBI from their respective NSI web sites, and, finally, NAF and CPF from Insee's web site. Ateco, NAF and CPF expressed in SKOS, are already published by Istat and Insee, so we took those "as-is" even if the modeling can differ from the one used for the rest of the classifications.

8.2. Description of Classification Explorer

The Classification Explorer [28] is a single-page browser application that communicates with the RDF database. The client side is a JavaScript application written using NodeJS, a modern JavaScript runtime that allows modern JavaScript development; ReactJS, a library for building user interfaces; and finally Webpack, a tool to package, deploy and redeploy the application. The client side is based on the React-Redux pattern (http://redux.js.org/) which manages the state of the application. As the application is single-page, the navigation is entirely done on the browser; React-router (https://github.com/reactjs/react-router) takes care of updating the URL so that the user can refresh, resume later or share the page he is on. The application fetches the data from a remote server, detailed in the next section, using SPARQL 1.1 queries over HTTP. Data is then stored locally and so is accessed only once.

The server-side of the application is a RDF database that hosts all the classification data in a unified database; it exposes the data as a SPARQL web server. Specifically, a Stardog instance was used as database.

The Web client offers different main features:

- **Browse various classifications**. At the moment, about 15 classifications are supported and can be browsed.
- **Compare two classifications**. When available, correspondence tables allow user to compare classes and see which items they were merged or split to.
- **Search among classifications**. A search box allows the user to search for any text within the title or the description of the items. The search can be done either on a specific classification or on all classifications.
- **Export data to CSV**. Each displayed list can be exported and downloaded as CSV file.

8.3. Description of Model Explorer

The Model Explorer is a deliverable of the work package 2 of the IMS project. It implements some of the use case described in section 3, for example use cases 4 and 6, and partially use case 2.

The application uses an RDF database on the Stardog sandbox, containing:

- The GSBPM, GSIM and CSPA ontologies described previously
- The description of 10 CSPA services, in conformance with the CSPA ontology
- A list of NSIs provided by UNECE and transformed in RDF in conformance with the ORG and vCard ontologies

The client application is based on exactly the same technical stack than the classification explorer, so the description will not be repeated here. The main functionalities of the client are:

- **Browse the CSPA services**. For each service, the service definition is provided with the relevant information, in particular the GSBPM sub-process and GSIM inputs and outputs.
- **Browse the GSBPM**. The client presents the usual graphical layout of the GSBPM, where each phase and sub-process can be clicked in order to access detailed information, in particular the list of CSPA services operating in the phase or sub-process.
- **Browse the GSBPM**. The client presents a graphical layout of the GSIM, organized by package (Base, Business, etc.). For each package, the list of GSIM object is provided, and

each object can be clicked in order to access detailed information, in particular the list of CSPA services consuming the object as input or producing it as output.

- **Edit CSPA services**. Every characteristic of the CSPA services can be modified, and services can be created or deleted.
- **Browse by NSI**. The client displays a list of statistical institutes by country. The country code links to the entry dedicated to the country in DBpedia (an RDF extract of Wikipedia), and the NSI name can be clicked in order to access detailed information, in particular the list of CSPA services for which the NSI plays a CSPA role. The NSI address is also provided with a link to OpenStreetMap.

Like the classification explorer, the model explorer is open-sourced (30) under a MIT licence. A developer's guide is provided for potential contributors.

9. Recommendations for Sustainability of IMS Project's Results

There are three main issues concerning the sustainability of the Linked Open Metadata part of the IMS project, namely:

1. Maintenance of the project's software artefacts
2. Extending the work on design guidelines
3. Promoting the adoption of linked metadata as a modelling framework for MOS projects.

Each issue will be separately detailed in the following and some suggestions on how to address them will be described.

Maintenance of the project's software artefacts

As detailed in the present document, the project involves several software artefacts that need to be maintained in terms of:

1. Having an IT platform that can make them accessible.
2. Being updated if changes are required due to underlying vocabularies or models' updates.

Suggestion 1: So far the projects artefacts are deployed on the "Sandbox" environment created with support from the Central Statistics office (CSO) of Ireland and the Irish Centre for High-End Computing (ICHEC). The "Sandbox" environment could remain accessible until new facilities for sharing the project's RDF artefacts are available.

Suggestion 2: In order to manage the updates of the projects artefacts, including both RDF artefacts and client software solutions (Classification Explorer and Model Explorer), there should be the involvement of some groups of the HLG-MOS . The groups could be "Supporting Standards", mainly involved on the issue of updating standards and related software artefacts and "Sharing tools", mainly involved in the maintenance and support of client software solutions.

Extending the work on design guidelines

The design guidelines described in this document are the results of a learn by doing approach. Hence they do have some limitations related to the limited number of examples that we were able to take into account, as well as the "pioneering" approach that we followed, due to the lack of previous available examples.

We think that both within the HLG-MOS community but also outside it, it is appropriate to plan activities that could revise, refine and extend such guidelines.

Suggestion 1: Within the HLG-MOS community, the "Supporting Standards" group is the best candidate to undertake activities aimed at extending the work on design guidelines.

Suggestion 2: Outside the HLG-MOS community, there is the DIGICOM project whose scope is very much aligned with the work done within the IMS project. In particular, the design guidelines on classifications could be evaluated by the DIGICOM project participants and possible modifications or enhancements could be proposed. In addition, the European ISA^2 program, in particular the related SEMIC (Semantics Interoperability Community), is also very much aligned with the objectives of the IMS project. It would be very much fruitful to have a direct involvment of SEMIC on the design guidelines work.

Promoting the adoption of linked metadata as a modelling framework for MOS projects.

One of the major outcomes of the IMS project is the proof that Linked Data is an excellent paradigm for representing statistical classifications and models so that they are (i) formally correct and complete and (ii) usable by software systems.

In order to promote the adoption of such paradigm, some concrete suggestions are listed in the following.

Suggestion 1: HLG-MOS could liaise with UNSD and Eurostat to reach international consensus on making international classifications available as linked metadata. If this kind of agreement is actually reached, all the efforts on developing systems to manage classifications could be shared with significant cost and quality benefits.

Suggestion 2: HLG-MOS could promote the Linked Data paradigm both internally and externally and develop associated capabilities. The "Capabilities and Outreach" group could have an activity dedicated to this promotion.

Suggestion 3: HLG-MOS could continue the work done so far on implementing RDF artefacts for both classifications and models. For instance, new ontologies for GAMSO and Quality Indicators could be proposed and adopted. The "Supporting Standards" could have a dedicated activity on that.

10. References

[1] GSIM – General Statistical Information Model:
http://www1.unece.org/stat/platform/display/metis/Generic+Statistical+Information+Model

[2] GSBPM – General Statistical Business Process Model:
http://www1.unece.org/stat/platform/display/metis/The+Generic+Statistical+Business+Process+Model

[3] CSPA – Common Service Process Architecture:
http://www1.unece.org/stat/platform/display/CSPA/CSPA+v1.5

[4] GAMSO – Generic Activity Model for Statistical organizations:
http://www1.unece.org/stat/platform/display/GAMSO/GAMSO+v1.0

[5] PROV Ontology: https://www.w3.org/TR/prov-o/.

[6] SKOS Simple Knowledge Organization System (SKOS): https://www.w3.org/TR/skos-reference/.

[7] DCAT – Data Catalogue Vocabulary: https://www.w3.org/TR/vocab-dcat/

[8] XKOS – eXtended KOS: http://rdf-vocabulary.ddialliance.org/xkos.html

[9] Dublin Core, dcmi-terms: http://dublincore.org/documents/dcmi-terms/

[10] IMS – Implementing ModernStats Standard Project
http://www1.unece.org/stat/platform/pages/viewpage.action?pageId=122323917

[11] EARF – Enterprise Architecture Reference Framework:
https://ec.europa.eu/eurostat/cros/content/ess-enterprise-architecture-reference-framework_en

[12] ATECO 2007 - Italian Classification of the Economical Activities:
http://www.istat.it/it/strumenti/definizioni-e-classificazioni/ateco-2007

[13] NAF – French Classification of the Economical Activities: http://recherche-naf.insee.fr/SIRENET_Template/Accueil/template_page_accueil.html

[14] ISIC – International Standard Industrial Classification of All Economic Activities:
http://unstats.un.org/unsd/cr/registry/regcst.asp?Cl=27

[15] CPC – United Nations Central Product Classification:
http://unstats.un.org/unsd/cr/registry/cpc-2.asp

[16] SDMX: http://sdmx.org/

[17] CPA – European Classification of Products by Activity: http://ec.europa.eu/eurostat/web/cpa-2008

[18] IMS project Use Cases:
http://www1.unece.org/stat/platform/display/IMS/Use+Case+Examples

[19] ADMS Asset Description Metadata Schema: https://joinup.ec.europa.eu/asset/adms/home

[20] Naming Policy: https://github.com/FranckCo/Stamina/blob/master/doc/content.md

[21] RDF: https://www.w3.org/RDF/

[22] OWL - Ontology Web Language: http://www.w3.org/TR/owl-ref/ , 10 February 2004

[23] Guidelines ADMS – DCAP-AP:
http://www1.unece.org/stat/platform/display/IMS/Report%3A+Databases

[24] DCAT-AP Application Profile:
https://joinup.ec.europa.eu/asset/dcat_application_profile/description

[25] Guidelines for XKOS classification:
https://github.com/FranckCo/Stamina/blob/master/doc/xkos-best-practice.md

[26] IMS Project Report: Challenges, Best Practices, Lessons Learned:
http://www1.unece.org/stat/platform/display/IMS/Report%3A+Challenges%2C+Best+Practices%2C+Lessons+Learned

[27] Stardog Environment: https://unece-stardog.ichec.ie/

[28] Classification Explorer:
http://www1.unece.org/stat/platform/display/IMS/Report%3A+Classification+Explorer

[29] M. Scannapieco, L. Tosco, D. Gillman, A. Dreyer, G. Duffes: "An OWL Ontology for the Generic Statistical Information Model (GSIM): Design and Implementation". In the Proceedings of the 4th International Workshop on Semantic Statistics, http://ceur-ws.org/Vol-1654/article-03.pdf..

[30] The PROV Data Model, https://www.w3.org/TR/prov-dm/.

[31] A. Dreyer, G. Duffes, F. Cotton: "An OWL Ontology for the Common Statistical Production Architecture". In the Proceedings of the 4th International Workshop on Semantic Statistics, http://ceur-ws.org/Vol-1654/article-06.pdf.

[32] The Organization Ontology, https://www.w3.org/TR/vocab-org/.

[33] http://github.com/UNECE