Open Data Management in Agriculture and Nutrition

*This e-learning course is the result of a collaboration between **GODAN Action** partners, including **Wageningen Environmental Research (WUR)**, **AgroKnow**, **AidData**, **the Food and Agriculture Organization of the United Nations** (FAO), **the Global Forum on Agricultural Research** (GFAR), and **the Institute of Development Studies** (IDS), **the Land Portal, the Open Data Institute** (ODI) and **the Technical Centre for Agriculture and Rural Cooperation** (CTA).*

GODAN Action is a three-year project UK's Department for International Development to enable data users, producers and intermediaries to engage effectively with open data and maximise its potential for impact in the agriculture and nutrition sectors. In particular we work to strengthen capacity, to promote common standards and best practice and to improve how we measure impact. [www.godan.info]

UNIT 4: SHARING OPEN DATA

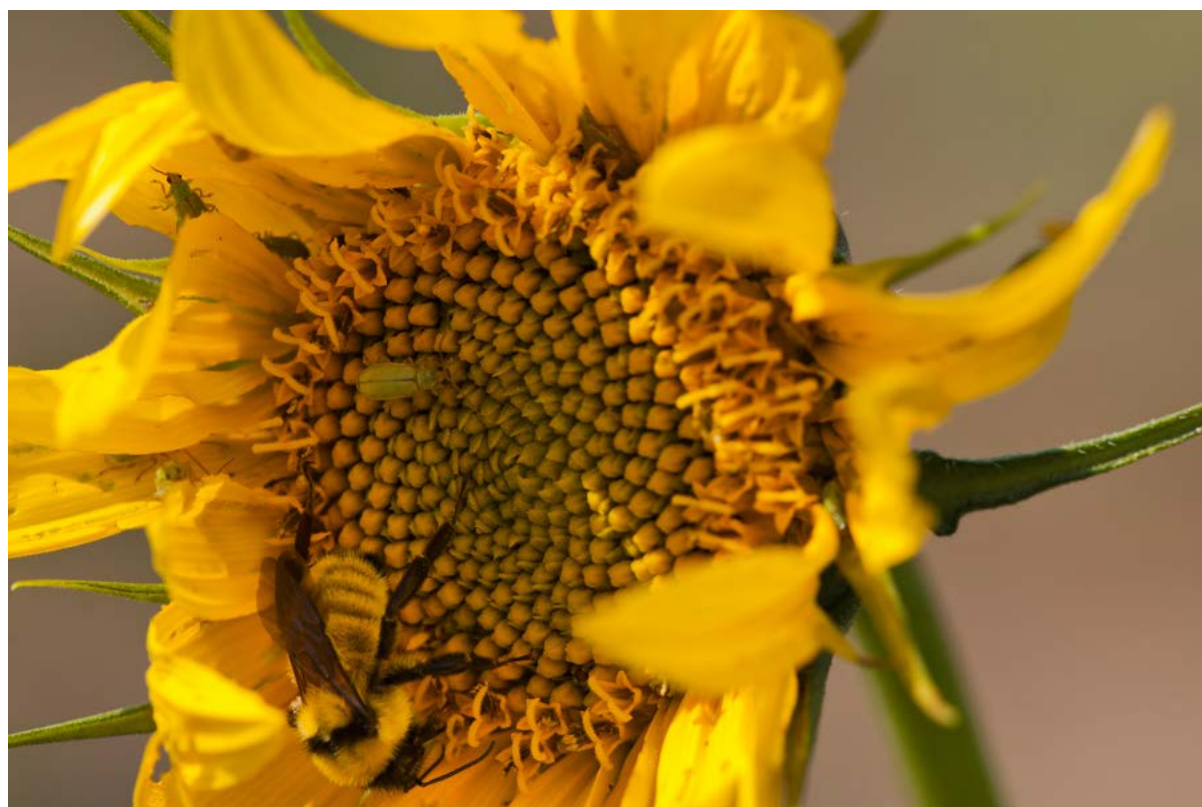# LESSON 4.4.1: USING PUBLISHED SEMANTICS FOR AGRONOMIC DATA

Aims and learning outcomes

This lesson aims to provide guidance and examples on how to encode (meta)data using published semantics, with specific examples for agronomic data

After studying this lesson, you should be able to;

- Understand how to identify suitable data standards to share agronomic data
- understand how semantics are embedded in the (meta)data
- (guide developers to or choose tools that) adopt the most suitable vocabularies for data of a specific type

Contents

List of figures

1. Introduction

In lesson 4.4, we gave an overview of semantic interoperability. In this lesson, we will provide more precise examples of how to implement it, especially identifying and reusing published semantics for agronomic data.

Practical examples that we will use throughout this lesson are related to agronomical observations and experiments. We will also provide examples of how to add semantics to the dataset metadata.

2. Identifying the most suitable vocabularies

We already saw in lesson 4.4 some useful pointers for selecting suitable vocabularies. Now let's see some practical examples, for the case of datasets of agronomic observations/experiments (you can look up all vocabularies mentioned in this lesson in the already mentioned GODAN map at http://vest.agrisemantics.org).

- In general, you will need a vocabulary that helps you **describe the dataset** that contains your data, to facilitate the discovery of the dataset. In this case, you should look for vocabularies that describe as precisely as possible a dataset or your specific type of dataset. For example, in the Agrisemantics Map of standards, you can search for vocabularies that describe 'Information resource metadata' (generic vocabularies, for any type of resource) and you would already find vocabularies like Dublin Core or schema.org that are understood by many tools and can encode **metadata** like title, author, subject, format, spatial and temporal metadata etc.

  Then you can narrow down your search to data standards that describe a 'Dataset'. You would find vocabularies like the W3C Data Catalog and the W3C Data Cube, as well as data formats like NetCDF and HDF5. The choice between these data standards depends on how you want to expose your data: by reading the descriptions, you would find out that NetCDF and HDF5 are data structures with no semantics and not extensible with schemas, so if you use them you cannot use any additional published semantics to enrich the data. On the other hand, you would also read that they are especially suitable for large amounts of observations, so for the case of agronomic observations you may want to consider them.

  You can even decide to expose your data using more than one data standard: see section 3.

- The vocabularies above provide the metadata for describing the dataset, but you may want to also identify value vocabularies to use as **controlled vocabularies for some of your dataset metadata**. For

instance, for the Dublin Core element dc:subject or the DCAT element dcat:theme, you may want to indicate the topic using a controlled term from a thesaurus or a classification. You can then search for vocabularies of type thesaurus or classification or subject heading, and perhaps narrow down the search to a specific domain or type of data. If you want to use a general and well known thesaurus for agriculture you can choose to use terms from AGROVOC to identify your topic, or if you share agronomic data in a specialised community you can narrow down the search to ontologies for plants or for agronomic observations or experiments and you may find that the INRA Agricultural Experiments Ontology has the terms you need. You can see how to use these values in your metadata in section 3.

● Whether you are sharing your dataset metadata and your data in the same file (e.g. an XML or an RDF file) or separately (e.g. an XML file for the metadata and a CSV or NetCDF for the data), you may want to look for additional **semantics for your data**. For instance, if your dataset is a dataset of agronomic observations, you may want to:

○ Use a widely adopted standard for sharing observations (independent of the type of observation): you can search for **schemas or ontologies** describing observations and you would find the ISO/OGC Observations & Measurements (O&M) standard, formalised as XML schema and JSON schema.
○ Use agreed variable names for each phenomenon or observed property: you can search for vocabularies of type **data dictionary** describing field observations and you would find for instance the AgMIP ICASA Master Variable List.
○ Use **standardised values** for the values of some properties (like shape or other phenotypic properties of a plant): you can search for code lists or ontologies of phenotypic properties and you would find the OBO Phenotypic Quality Ontology for general phenotypic properties (like shapes) and several crop-specific ontologies.

Besides choosing a vocabulary suitable for the data you manage, some other criteria are useful for selecting the most appropriate vocabulary:

● The ideal implementation of the 'Semantic Web'[1] is through namespaces, URIs and the Linked Data[2] approach (see lesson 4.3): in order to be able to exploit these technologies, it is preferable to choose vocabularies that have been published as RDF (XML can be a good option too) and follow the Linked Data approach (use URIs, link to other vocabularies).

---

[1] W3C. Semantic Web. https://www.w3.org/standards/semanticweb/
[2] W3C. Linked Data. https://www.w3.org/standards/semanticweb/data

- It is preferable to choose vocabularies that are widely used, so that your data are interoperable with more datasets and more systems.
- If existing vocabularies do not meet your needs, you may decide to create your own vocabulary and publish it (ideally, in collaboration with other partners in your community so as to ensure wide adoption).

Once you have identified vocabularies that are relevant to your type of data, let us see how to use them.

3. Using existing vocabularies for your agronomic data

In lesson 4.4 we explained how to encode data using an XML schema or an RDF vocabulary, and we saw a few examples. In this lesson we will provide more examples potentially relevant to the field of agronomy.

An important caveat: we give examples below assuming the data manager has the freedom to choose the data format of his choice and the vocabularies of his choice. In general, since implementing the examples below or similar ones is not something that you would normally do manually, there are two ways in which a data manager can implement semantics:
- using ad-hoc dataset/metadata generators or converters, written by their developers and configurable at their will;
- using a dataset management tool that supports semantics: it is not easy to find a tool that allows you to expose (meta)data using any vocabulary of your choice; especially for the metadata model, tools normally come with their own model, which can (should) correspond to a published widely used vocabulary, so the data manager can only choose among different tools the one that has the best metadata model; some tools may allow the data manager to use controlled values from value vocabularies for specific metadata.

Therefore, in some cases the examples below (and similar ones) cannot be freely implemented, as there may be technical constraints due to the tools that are used. However, seeing how semantics could be ideally embedded in (meta)data can also help choose the best tools and will raise the expectations of data managers, thus making them more demanding in terms of tools and perhaps leading to the improvement of existing tools.

In the next two sections we will apply the distinction, as mentioned in lesson 4.4, between semantics for the data structure (description vocabularies, schemas) and semantics for the values (value vocabularies).

3.1. Vocabularies for the (meta)data structure

As vocabularies for the (meta)data structure you would normally use a schema or an ontology. Schemas and ontologies are the encoding of a (meta)data model, so while the type of encoding (XML schema, RDF schema, OWL ontology) makes the vocabulary especially suitable for a specific (meta)data format (typically XML or RDF), as metadata models they can be adopted also for (meta)data in different formats, as long as the formats support schemas or custom element names. So, schemas and ontologies can be used in some way in JSON and CSV (see the section on data formats in lesson 2.3), as JSON structure and labels or as CSV column names, as we saw in lesson 4.4.

On the other hand, some data standards are designed as format-agnostic models and are then formalised as schemas of different types (XML schema, ontology) and specifications are provided to use them in other formats (CSV, JSON).

3.1.1. Semantics for dataset metadata

As noted in section 2, when looking for vocabularies for dataset metadata you would find the W3C RDF vocabularies DCAT and Data Cube as well as the ISO 19115 (Geographic Information - Metadata[3]) XML schema (ISO/TS 19139 XML Schema[4]). Reading further and examining the vocabularies, you would notice that:

- DCAT has basic properties for describing the dataset 'resource' and its distributions, but no properties to describe the data structure;
- Data Cube is especially conceived for statistical datasets, but has classes and properties that can describe the data structure of any observation dataset (DataStructureDefinition, dimension, measure) and to encode the observations themselves;
- ISO 19115 has extensive metadata both for describing the dataset as a resource and for describing the data structure.

Let us say that you want to serialise your dataset metadata in RDF for better interoperability and that you want to expose full metadata about the data structure so that software tools can automatically parse the data as well. In this case, the Data Cube vocabulary may be the appropriate choice.

As we briefly mentioned in the previous section, you can use more than one vocabulary in the same dataset. It is sufficient to declare the namespaces of all the vocabularies used in the header and then use their classes and properties in the file. In the example below, we skip the header with all the

---

[3] https://www.iso.org/standard/53798.html

[4] http://www.isotc211.org/schemas/2005/

prefixes and namespaces: suffice it to say that 'eg:' is just a prefix for the local namespace, 'qb:' is the prefix for the namespace of the W3C Data Cube vocabulary, and 'dct:' is a prefix for the Dublin Core vocabulary.

The example below describes a data structure designed to contain one measurement ('minimum daily air temperature, average', indicated with a conventional URI from the ICASA Master Variables List, not yet published) and three dimensions: area, period and identifier of the field where the measurement is taken. This means that these will be the data in that dataset and the names and attributes used in the data will be those indicated here. See the next subsection for an example of the data in this dataset serialised with Data Cube as well.

```
eg:dataset-le3 a qb:DataSet;
    dct:title              "Minimum daily air temperature 2004"@en;
    rdfs:label             "Minimum daily air temperature 2004"@en;
    dct:description        "Minimum daily air temperature 2004 - From fields..."@en;
    dct:publisher          eg:organization ;
    dct:issued             "2010-08-11"^^xsd:date;
    dct:subject            <http://aims.fao.org/aos/agrovoc/c_230> ;
    qb:structure           eg:dsd-le3 ;

eg:organization a foaf:Organization;
    foaf:name              "Test organization";

eg:dsd-le3 a qb:DataStructureDefinition;
    # The dimensions
    qb:component           [ qb:dimension     eg:refArea;        qb:order 1 ];
    qb:component           [ qb:dimension     eg:refPeriod;      qb:order 2 ];
    qb:component           [ qb:dimension     eg:field_id;       qb:order 3 ];
    # The measure(s): "Minimum daily air temperature, average" from ICASA variables
    qb:component           [ qb:measure       <http://purl.org/icasa/variables#tmina>];
    # The attributes
    qb:component           [ qb:attribute sdmx-attribute:unitMeasure;
                                qb:componentRequired "true"^^xsd:boolean; ] .
```

Figure 1 Example of RDF encoding of dataset metadata and data structure using Data Cube

ISO 19115 is also a suitable vocabulary covering dozens of metadata elements for a dataset. While ISO/TS 19139 provides an XML schema for ISO 19115, an RDF (OWL) representation of ISO 19115 has been developed[5] by CSIRO Australia.

You can encode dataset metadata in a separate file from the actual dataset or in the same file as the data. This depends on several factors: your (meta)data model and the vocabulary you choose respectively for the dataset metadata and the data (some vocabularies cover both things in the same data structure, many others do not), your repository/catalogue architecture, the size of the data, the format of your data (if the data format does not allow for dataset metadata, or does not allow for metadata using the vocabulary of your choice, you may need to create a separate file for the

---

[5] http://def.seegrid.csiro.au/static/isotc211/iso19115/2003/metadata.html

metadata). In the next section we shall see how to use semantics in the data section (whether in the same file or in a separate one, depending on the conditions above).

3.1.2. Semantics for data

Let us start from the example of a dataset of observations from agricultural experiments. From your search in section 2, you found that there is a general ISO data standard for Observations and Measurements[6] (O&M), which, starting from a UML model, provides both an XML schema and a JSON schema of the model. The example below shows how to use elements from the O&M XML schema to describe a simple observation: the measurement of fruit mass at the temperature of 22.3 °C.

```
<om:OM_Observation <!-- namespaces hidden -->>
<gml:description>Observation test instance: fruit mass</gml:description>
<gml:name>Observation test 1</gml:name>
<om:type xlink:href="http://www.opengis.net/def/observationType/OGC-OM/2.0/OM_Measurement"/>
<om:phenomenonTime> <!-- hidden --> </om:phenomenonTime>
<om:resultTime xlink:href="#ot1t"/>
<om:procedure xlink:href="http://www.example.org/register/process/scales34.xml"/>
<om:parameter>
  <om:NamedValue>
  <om:name xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Temperature"/>
  <om:value xsi:type="gml:MeasureType" uom="Cel">22.3</om:value>
  </om:NamedValue>
</om:parameter>
<om:observedProperty xlink:href="http://sweet.jpl.nasa.gov/2.0/phys.owl#Mass"/>
<om:featureOfInterest xlink:href="http://wfs.example.org?request=getFeature&amp;featureid=fruit37f "/>
<om:result xsi:type="gml:MeasureType" uom="kg">0.28</om:result>
</om:OM_Observation>
```

Figure 2 Example[7] of observation in XML format using the O&M schema

The next example shows how to use the O&M JSON schema to describe a simple observation: the measurement of air temperature at a specific point. The labels used ('observedProperty', 'featureOfInterest'...) and the nesting structure ('uom' under 'result') clearly show that even if in a different format, the schema is the same. Since in the XML file the prefix 'om:' would be mapped to the namespace of the XML schema, and in the JSON file the @context would be the URL of the JSON schema, any software parsing datasets in either format will interpret the elements/labels in the same way.

```json
{
  "id":"measure-instance-test",
  "type": "Measurement",
  "phenomenonTime": { "instant":"2011-05-11T00:00:00+10:00" },
  "observedProperty": {"href":"http://environment.data.gov.au/def/property/air_temperature"},
  "procedure": {"href":"http://www.opengis.net/def/waterml/2.0/processType/Sensor"},
  "featureOfInterest": {"href":"http://waterml2.csiro.au/rgs-api/v1/monitoring-point/419009/"},
  "resultTime": "2011-05-12T09:00:00+10:00",
  "result": {
          "value": 3.2,
          "uom": "http://qudt.org/vocab/unit#DegreeCelsius" }
}
```

Figure 3 Example of observation in JSON format using the O&M model

If you do more research, you will find that there is also a data standard specific for agricultural experiments: the ICASA data standard[8]. An XML schema for this standard is still under development, but the standard has been serialised into JSON.

The example below shows an experiment encoded in JSON using the ICASA data model and variables. You can see that instead of URIs ICASA uses short coded variable names: all the codes are in the ICASA Master Variables list[9], which defines the meaning of all variables and constitutes the ICASA semantic resource, where you would find that 'fielele' means 'field elevation' and 'icpcr' means 'residue, crop code for previous crop'.

Since variables are identified by codes and not by URIs, and codes are not even associated with definitions in a machine-readable file, software tools cannot look up the meaning and cannot infer the reference semantics behind the code. Therefore, even if the ICASA variables are probably the most complete list for agricultural experiments and are used in other systems, at the moment using them does not ensure full semantic interoperability. There is work to express the ICASA variables in an ontology[10].

---

[8] http://dssat.net/wp-content/uploads/2014/02/White2013ICASA_V2_standards.pdf

[9] http://research.agmip.org/display/dev/ICASA+Master+Variable+List

[10] https://github.com/craig-willis/icasa/blob/master/docs/design.md

```
"exname": "UFGA8201MZ_1",
'local_name': "NIT X IRR, GAINESVILLE 2N*3I",
'people': "BENNET,J.M. ZUR,B. HAMMOND,L.C. JONES,J.W.",
'institution': "UNIVERSITY OF FLORIDA, GAINESVILLE, FL, USA",
'site': "IRR.PARK,UF.CAMPUS 29.63;-82.37;40.;FLA",
'tr_name': "RAINFED LOW NITROGEN",
'id_field': "UFGA0002",
'wst_id': "UFGA",            /* weather station id */
'soil_id': "IBMZ910014",  /* soil id               */
'fl_lat': "29.63",          /* latitude              */
'fl_long': "-82.37",        /* longitude             */
'flele': "40",              /* field elevation       */

'initial_condition': [
    "icpcr": "MAZ",          /* previous crop was maize      */
    "icdat": "19820225",     /* date for initial conditions  */
    "icrt": "100",           /* initial root residue kg/ha   */
    "icrag": "1000",         /* initial surface residue kg/ha */
    "icrn": ".8"             /* initial residue N %          */
    "soilLayer": [           /* init soil layer data:        */
    /* depth (cm), moisture (frac),     NH4 (ppm),    NO3 (ppm) */
    {"icbl":  "5", "ich2o": ".086", "icnh4": ".5", "icno3": ".1"},
    {"icbl": "15", "ich2o": ".086", "icnh4": ".5", "icno3": ".1"},
    {"icbl":"180", "ich2o": ".258", "icnh4": ".5", "icno3": ".1"}
    ]},

'management':[
    "events":[
        {"event":"fertilizer",        /* Fertilizer application  */
                 "date":"19820225",   /* Feb 25, 1982            */
                 "fecd":"FE001",      /* Ammonium nitrate        */
                 "feacd":"AP001",     /* Broadcast, incorporated */
                 "fedep":"10",        /* 10 cm deep              */
                 "feamn":"27"),       /* 27 kg[N]/ha             */
```

Figure 4 Example[11] of an experiment described in JSON format using the ICASA standard

If you look for an RDF vocabulary for full interoperability, then you may want to consider the W3C 'Semantic Sensor Network Ontology' (SSN), an ontology built on the basis of OGC SensorML and O&M standards. The classes and properties that are most relevant for observations are under the namespace http://www.w3.org/ns/sosa/ (Sensor, Observation, Sample, and Actuator – SOSA). You will notice that the vocabulary follows the OGC Observation–FeatureOfInterest–ObservedProperty model. Note that this description also uses other vocabularies, in particular @prefix qudt-1-1: http://qudt.org/1.1/schema/qudt#, a vocabulary for units of measure.

```
<observation/1087> rdf:type  sosa:Observation ;
 rdfs:label "observation #1087"@en ;
  sosa:hasFeatureOfInterest <tree/124> ;
  sosa:observedProperty <tree/124/height> ;
  sosa:hasResult [
    qudt-1-1:unit qudt-unit-1-1:Meter ;
    qudt-1-1:numericalValue "15.3"^^xsd:double ] .

<tree/124>  rdf:type  sosa:FeatureOfInterest ;
 rdfs:label "tree #124"@en .

<tree/124#height>  rdf:type  sosa:ObservableProperty , ssn:Property ;
 rdfs:label "the height of tree #124"@en .
```

Figure 5 Example[12] of observation in RDF Turtle using the W3C SSN ontology

---

[11] Integrated description of agricultural field experiments and production: The ICASA Version 2.0 data standards. http://dssat.net/wp-content/uploads/2014/02/White2013ICASA_V2_standards.pdf
[12] Semantic Sensor Network Ontology. https://www.w3.org/TR/vocab-ssn/#Observations-overview

Still part of the family of data standards for observations stemming from O&M, an OWL version of O&M ('OWL for Observations'[13]) was developed by CSIRO. This ontology is also aligned with the SSN ontology mentioned above.

```
my:obsTest1 a oml:Measurement ;
  rdfs:comment "Observation test instance: fruit mass"^^xsd:string ;
  rdfs:label "Observation test 1"^^xsd:string ;
  oml:featureOfInterest <http://wfs.example.org?request=getFeature&amp;featureid=fruit37f> ;
  oml:observedProperty <http://sweet.jpl.nasa.gov/2.0/phys.owl#Mass> ;
  oml:phenomenonTime [ a w3time:Instant ;
    w3time:inXSDDateTime "2005-01-11T16:22:25.00"^^xsd:dateTime ] ;
  oml:procedure my:Sscales1 ;
  oml:result [ a oml:Measure ;
    rdf:value "0.28"^^oml:Number ;
    oml:uom <http://www.opengis.net/def/uom/UCUM/0/kg> ] ;
 oml:resultTime "2005-01-11T16:22:25.00"^^xsd:dateTime
```

Figure 6 Example[14] of an observation encoded in RDF (Turtle) with the OM Lite ontology

Or, continuing with the example in the previous section using the Data Cube vocabulary for the dataset metadata, observations can also be encoded using Data Cube, although inside the Observation entity other vocabularies are needed. The example below encodes the measurement of daily minimum temperature (following the data structure defined in the Data Cube dataset header example in the previous section).

Besides the namespaces used in the previous example, here we have two additional namespaces for the metadata elements, one for statistical attributes (SDMX) and one for variables (the ICASA variables):

@prefix sdmx-attribute: <http://purl.org/linked-data/sdmx/2009/attribute#>
@prefix icasa-var: <http://purl.org/icasa/variables#>

```
eg:o1 a qb:Observation;
  qb:dataSet  eg:dataset-le1 ;
  eg:refArea        ex-geo:newport_00pr ;
  eg:refPeriod        <http://reference.data.gov.uk/id/gregorian-interval/2004-01-01T00:00:00/P3Y> ;
  eg:field_id        1234 ;
  sdmx-attribute:unitMeasure   qudt-unit:DegreeCelsius ;
  # tmina: "Minimum daily air temperature, average" from ICASA variables
  icasa-var:tmina  1.7 ;
```

Figure 7 Observation encoded in RDF (Turtle) using Data Cube

As we said in lesson 4.4, the higher level of interoperability of using an RDF vocabulary (and therefore an RDF dataset), especially if following the Linked

---

[13] http://www.essepuntato.it/lode/owlapi/http://def.seegrid.csiro.au/ontology/om/om-lite
[14] Cox, Simon J D. Basic Observations and Sampling Feature Ontology. http://www.semantic-web-journal.net/system/files/swj890.pdf

Data pattern (see lesson 4.2) is due to two facts: (a) metadata elements (classes or properties in RDF) are identified by URIs and those URIs are dereferenced to web pages that contain machine-readable information about the class/property, so a computer program can follow the link and read more data, and if there are other links to other external entities it can continue following them, obtaining more and more data; and (b) as we said in lesson 4.3, an RDF file has fewer potential ambiguities than a (non-RDF) XML file and is interpreted more reliably.

The Semantic Web approach has been designed on the assumption that Linked Data technologies are used, so the ideal approach to semantic interoperability is using RDF and the core features of Linked Data (URIs and links to other URIs).

3.2.   Vocabularies for (meta)data values

A different case occurs when you want to use values from an existing vocabulary as **values of some of the metadata**, for instance if you want to use an unambiguous term for 'air temperature' or for identifying a plant species. In this case, you are not looking for a vocabulary (one or more) that has the classes and properties you need to describe/structure/model your data; you are looking for vocabularies that contain the terms, the concepts that you want to use for specific values, because you want to **identify them univocally and unambiguously** (no arbitrary strings), with authoritative terms/codes, independent of languages and local practices. For instance, you may want to use the AGROVOC term for 'air temperature' or the Plant Ontology identifier for a plant species.

As a minimum, once some suitable vocabulary has been identified, if the vocabulary doesn't use URIs and/or the URIs cannot be used in the dataset, at least the literal values of the terms can be used in the dataset: systems that are aware of the vocabulary and can match the literal against the URI can already do something with this.

Ideally, you should use the URI of the term you want to refer to. Let's see how URIs from published semantic resources are used as values in the examples we provided in the previous section.

In the Data Cube dataset metadata example in figure 4.4.1.1, the subject of the dataset is indicated using the URI of a concept in AGROVOC: a linked-data-aware tool would look up the URI http://aims.fao.org/aos/agrovoc/c_230 and would find an RDF description of the concept saying that the label in English is 'air temperature', that it is a narrower concept of 'temperature' and that there are identical concepts in other vocabularies, among which are the USDA NAL thesaurus and the GEMET thesaurus. The tool could exploit this information to combine this dataset with other similar datasets.

Furthermore, the property that is being observed / measured is defined with a URI: in this case the URI is conventional, because it comes from the experimental RDF version of the ICASA Variables Master List, which has not been published (therefore the URI is not resolvable yet). When the ICASA master list is published in RDF, a software tool will be able to see that http://purl.org/icasa/variables#tmina means 'Minimum daily air temperature, average, sowing to harvest'.

The mandatory attribute of the measurement, which is the unit of measure, is also indicated with a URI: the sdmx-attribute prefix is associated with the URI http://purl.org/linked-data/sdmx/2009/attribute#, so sdmx-attribute:unitMeasure is short for http://purl.org/linked-data/sdmx/2009/attribute#unitMeasure: looking up that URI, a tool would find that the concept name in English is 'Unit of Measure', that it means 'The unit in which the data values are measured' and has a full definition in a certain PDF file.

```
eg:dataset-le3 a qb:DataSet;
    dct:title              "Minimum daily air temperature 2004"@en;
    dct:subject            <http://aims.fao.org/aos/agrovoc/c_230> ;
    qb:structure           eg:dsd-le3 ;

eg:dsd-le3 a qb:DataStructureDefinition;
    [...]
    qb:component           [ qb:measure  <http://purl.org/icasa/variables#tmina>];
    qb:component           [ qb:attribute sdmx-attribute:unitMeasure; ] .
```

Figure 8 Use of vocabularies for values in Data Cube example in Figure 1

In the OM Lite example in figure 6, the feature of interest is indicated by the URL of an API that should return the RDF of a 'Feature' from a feature registry (different research communities are creating their own registries of relevant features; the URL used here is only an example).

The observed property (the mass) is indicated by a URI from the (…) https://sweet.jpl.nasa.gov/2.0/phys.owl#Mass: looking up this URI, a machine would find that for this property the default unit of measure is kilogram (indicated in turn by another URI). The unit of measure is specified again in the uom property, this time using a URI from the Open Geospatial Consortium (which is not fully resolvable); the URI from the NASA Japan service used above would have also worked well (https://sweet.jpl.nasa.gov/2.0/sciUnits.owl#kilogram); even better the URI for kilogram from the QUDT ontology (...): http://qudt.org/vocab/unit#Kilogram.

The observed property 'fruit mass' could have also been indicated with the URI for 'Fruit fresh weight' from the ICASA variables (that will be hopefully published soon):http://purl.org/icasa/variables#ffad.

For increased interoperability, all these URIs could have been used. Ideally, all different URIs in different semantic resources referring to the same concept

would be linked to each other through an owl:sameAs property, so that in your data you only need to provide one URI and semantic tools could look up all the linked ones. However, good reliable mappings don't exist yet, so the more external URIs you can link to, the better.

```
my:obsTest1 a oml:Measurement ;
  rdfs:comment "Observation test instance: fruit mass"^^xsd:string ;
  rdfs:label "Observation test 1"^^xsd:string ;
  oml:featureOfInterest <http://wfs.example.org?request=getFeature&amp;featureid=fruit37f> ;
  oml:observedProperty <http://sweet.jpl.nasa.gov/2.0/phys.owl#Mass> ;
  oml:procedure my:Sscales1 ;
  oml:result [ a oml:Measure ;
    rdf:value "0.28"^^oml:Number ;
    oml:uom <http://www.opengis.net/def/uom/UCUM/0/kg> ] ;
```

Figure 9 From example in figure 6

The examples in this lesson, however specific they are for field observations, can be easily replicated for other types of data, using vocabularies that follow a data model suitable for that type of data. The examples are meant to help you understand the mechanisms for adding shared/agreed semantics to your data.

Further Readings

- Antoniou, G., and van Harmelen, F. (2003). *A Semantic Web Primer*. MIT Press, Cambridge, MA, USA.. http://www.csd.uoc.gr/~hy566/SWbook.pdf
- White, J. W. *et al.* (2013). Integrated description of agricultural field experiments and production: The ICASA Version 2.0 data standards. *Computers and Electronics in Agriculture* **96**, 1–12. https://doi.org/10.1016/j.compag.2013.04.003