From AGROVOC to the Agricultural Ontology Service / Concept Server An OWL model for creating ontologies in the agricultural domain

Boris Lauser, Margherita Sini, Anita Liang, Johannes Keizer, Stephen Katz
{Boris.Lauser, Margherita.Sini, Anita.Liang, Johannes.Keizer, Stephen.Katz} @fao.org
Food and Agriculture Organization of the United Nations, Rome, Italy
http://www.fao.org/

Abstract

This paper illustrates the conversion from a traditional thesaurus in agriculture (AGROVOC) to a new system, the Agricultural Ontology Service Concept Server (AOS/CS). The Concept Server will serve as a multilingual repository of concepts in the agricultural domain providing ontological relationships and a rich, semantically sound terminology. The Food and Agriculture Organization recently developed the underlying model for this new system in the Web ontology language OWL. In this paper, we describe the purpose of this conversion and the use of OWL and highlight in particular the core features of the developed OWL model. We go on to explain how it evolves and differs from the traditional thesaurus approach.

Keywords

Ontologies, Thesauri, Semantic Web, OWL, Classification Schemes, Metadata, AGROVOC, AOS

1. Background and Introduction

Since 2003, the Food and Agriculture Organization (FAO) has been concerned with developing a new model for the AGROVOC thesaurus that accounts for semantic and lexical relations in more refined and precise ways, with the objective of building a multilingual repository of concepts in the agricultural domain, the Concept Server (CS).

This effort fits in with FAO's overall initiative to establish an Agriculture Ontology Service (AOS) which aims to function as a tool to help structure and standardize agricultural terminology in multiple languages for use by any number of different systems around the world. It will be possible to export the traditional AGROVOC thesaurus, as well as other forms of knowledge organization systems (KOS)[1], from the CS.. It will also be possible to extract ontological concepts and use them to build domain specific ontologies..

During the research, a number of models and approaches have been studied and proposed. Initially, a relational database was considered an advantageous storage solution because of:

- its ease of management, scalability, and performance;
- its similarity to the current format and the ability to ensure backward compatibility;
- the use of RDBs to store other terminologies to be integrated into AGROVOC, such as FAOTERM, FAO Glossary.

Subsequently, investigations on using the OWL Web Ontology Language (OWL) for representing the model of the Concept Server have been carried out. OWL is eliciting increasing interest from individuals from a wide range of disciplines and domains, including medicine, defence, agriculture, biology, library sciences, and more sophisticated and better performing technologies are continually being developed for building and using OWL ontologies. Although additional OWL database and triple store repository tests need to be

[1] KOS are knowledge structures, including authority files, classification systems, concept spaces, dictionaries, controlled lists, taxonomies, gazetteers, glossaries, ontologies, subject heading sets, thesauri, etc.

done to determine their performance and scalability, there appears to be a sufficient number of advantages that argue for the transition to OWL over the creation of a new ad-hoc RDB:

First, one of the major objectives of AOS is the promotion of standards and interoperability in the agricultural domain. Designing yet another proprietary model for a terminology system in this domain would run counter to that objective. Using an established standard like OWL will provide for maximal interoperability with other systems. Existing open source tools (Protégé, SWOOP, etc.) and methods can be used to handle the model and reused and modified for local applications, thus minimizing system development efforts.

Second, a customized database schema is not directly interoperable with other storage solutions. In contrast, an established standard XML/RDF-based format such as OWL is already interoperable with any RDF triple-store, which allows for easy integration of other RDF/XML-based data sources at the storage level and straightforward data processing and visualization. But OWL is more than RDF. Using OWL, ontologies can be shared easily across the Web, since OWL is explicitly able to draw equivalences between classes and individuals across terminologies. Consistency checks can be performed on linked ontologies to identify and resolve conflicts between the ontologies and reasoning can be used to arrive at conclusions beyond those asserted.

Third, using an established standard like the OWL model will minimize training efforts. It is sufficient to refer to publicly available OWL documentation, instead of having to create heaps of documentation for a proprietary system.

Finally, having attained the status of a W3C recommendation means that it has become a stable specification that has achieved a high level of technical quality, and is meant for widespread deployment in service of the goal of interoperability of the Web.

Based on these considerations, we developed a new model in OWL that will serve as a skeleton for building ontologies in the agricultural domain. In this paper we will present the most important features of this model, which will serve as a basis for the future AOS Concept Server. It also goes into details concerning the problems of multilingualism and shows our solutions. We will not explain OWL basics in this paper, but assume the reader to be familiar with ontologies and the OWL basics. For more details on OWL you can refer to [1]. As a modelling tool we used Protégé 3.2, a now widely used, Java-based open source ontology editor developed at Stanford University [2]. The screenshots used for illustration purposes in this paper have been created with this tool.

2. Expressing the semantics of AGROVOC in OWL

The purpose of re-engineering AGROVOC into an OWL model with a more ontology-like structure is

- to facilitate its use for developing agricultural domain terminologies, including ontologies, without requiring the terminologist to start from scratch;
- to enable the development of applications using semantic technologies; and
- to enable interoperability between applications using these ontologies.

As a starting point, AGROVOC is highly suitable for re-engineering into an ontology. Compared to ordinary wordlists or glossaries, it contains explicit semantics of a hierarchical structure between elements (terms representing agricultural concepts). It also contains generic associative relations that indicate a semantic relation between a pair of entities, and that can be further refined into more specific relations. Plant and animal species lists, geo-political entities, and chemical substances form natural taxonomies whose semantics can readily be expressed in OWL. Likewise, attributes (e.g., number of legs, size of land mass) and non-hierarchical relations (e.g., membership, plant part) can also be expressed.

3. The multilingual issue

To prepare AGROVOC for use as an ontology, it is essential to represent concepts by minimizing bias towards a given language or family of languages. That is, to the extent possible, meaning is considered independently of its realization in a particular language. Each language would then be able to express the domain concepts for which it had lexicalizations and for which others may not. A terminology that simply translated the terms in a given language, such as English, would miss out on concepts that were not lexicalized in that language. For example, the Italian word *loculo* ‘walled niche containing a coffin or cinerary urn’ [3] has no lexicalized counterpart in English. More examples of similar problems are encountered if we consider for example Asian languages for representing rice or mango related concepts. A multilingual terminology that was English-centric, as is arguably the case with AGROVOC, would fail to include these meanings. Thus, the proposed revision of the AGROVOC terminological structure will result in a domain model that will conceptually be richer than one that was based on a single language and translations. In addition to accommodating concepts from diverse languages (and hence cultures), the domain model should represent lexical relationships, both within and across languages. This would enable accurate lexical equivalences (e.g., translations, synonyms) to be made and allow efficient processing of terms and concepts as well as maximizing the value of the ontology for a variety of applications.

The three levels of representation that we are aiming to express in this model are

- concepts (the abstract meaning), for example ‘rice’ in the sense of a plant,
- terms (language-specific lexical forms), for example ‘Rice’, ‘Riz’, ‘Arroz’, ‘稻米’, ‘□□□□’, or ‘Paddy’,
- term variants (the range of forms that can occur for each term), for example ‘O. sativa’ or ‘Oryza Sativa’, ‘Organization’ or ‘Organisation’.

The abstract concepts build the actual hierarchy and semantic structure of the ontology. Terms are no longer arranged in a hierarchy or related via semantic relationships, as is currently done in AGROVOC. Each term is a separate entity in every language that can be linked to concepts, to other terms and to term variants of the same term.

These distinctions allow us to posit the following inter-level relations:

Concept to Term	has_lexicalization (links concepts to their lexical realizations);
Term to String	has_acronym, has_spelling_variant, has_abbreviation link terms to term form variants

String here simply means that the term variants do not constitute a new term, but are simply variant strings of the same term.

Intra-level relations occur at both the level of the concept and at the level of the term exemplified by the following:

Concept to Concept	is_a (hierarchy), pest_of, pest, etc.
Term to Term	is_synonym_of, is_translation_of

4. The OWL model

4.1 The OWL species

In this report, we present an OWL model that can capture the aforementioned conceptual and lexical distinctions while maintaining the characteristic of computational completeness. Therefore, the design of the multiple levels of lexical representation presented in this paper (classes, properties, annotations) is done in the version of OWL identified as OWL DL[2].

4.2 The basic model

The baseline of the new OWL model has three concepts at the top level, as shown in Fig. 1. Each entity of an OWL ontology has a unique URI[3]. In Fig. 1 you can see only the identifying last part of the URI. As a general convention for our model, each entity's URI is constituted by a prefix, c_ (for classes), r_ (for relationships/properties[4]), i_ (for instances), followed by a numeric or alphanumeric sequence.

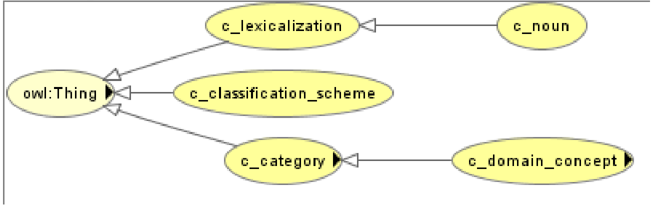


Fig. 1: Top level concepts.

The concept *c_domain_concept* is the root of all domain concepts that constitute the core hierarchical structure of the AOS Concept Server. This node subsumes all the basic structural characteristics of the domain ontology, i.e. a class hierarchy with classes and their instances along with their relations, properties, axioms, constraints and annotations pertinent to domain knowledge. Basically, all AGROVOC terms, or more precisely, AGROVOC descriptors, will be modelled under this node.

The class *c_domain_concept* is modelled as a sub-class of *c_category*, which implies that every domain concept is also potentially a category. The separate class *c_category* accounts for the need of specific categories that are not domain concepts. Categories are organized in Classification Schemes represented by the class *c_classification_scheme*. We will talk more about categories and classification schemes in section 4.5.

While the backbone structure of the domain ontology is modelled under *c_domain_concept*, the lexicalizations of the concepts will occur as instances of the class *c_lexicalization*. This modelling approach has been chosen instead of just using the rdfs:label on each concept to represent its lexicalization in a particular language. It addresses the aforementioned multilingual issue. Modelling lexicalizations as a separate concept will make it possible to establish relationships amongst various lexicalizations that describe a concept, and thus provide for much powerful semantics.

4.3 The hierarchical backbone structure

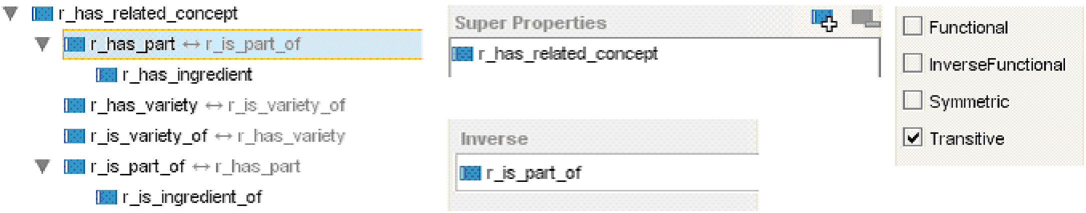
AGROVOC terms (more precisely, its main descriptors) will constitute the initial hierarchical backbone structure of the model. All AGROVOC descriptors will be modelled as sub-classes of *c_domain_concept* using the AGROVOC term code to form a class's URI (i.e. *c_208* for the concept of 'Agriculture'). The traditional thesaurus relationships *Narrower Term* and

[2] See also: http://www.w3.org/TR/owl-ref/#Sublanguages.
[3] We will not go into details of URIs here. Refer to http://www.w3.org/Addressing/ for more on URIs.
[4] Throughout this paper, we use the words property and relationship as synonyms. Property is the term used in the world of ontologies and OWL, whereas relationships is more common in the traditional thesaurus world.

Broader Term are then translated into OWL super-class and sub-class relationships and thus build the initial hierarchy of the Concept Server.

4.3.1 Relating concepts: the concept-to-concept interface

AGROVOC (as well as other classical thesauri) provides only one type of non-hierarchical, conceptual relationship, namely *related term*. In our model we want to provide the opportunity to relate concepts with more specific relationships. Therefore, we introduce a relationship hierarchy for concept relationships. Each specific conceptual relationship (like *is part of*, *is infected by*, etc.) is modelled as a sub property of *r_has_related_concept* as shown for a few examples in Fig. 2.


r_has_related_concept
r_has_part ↔ r_is_part_of
r_has_ingredient
r_has_variety ↔ r_is_variety_of
r_is_variety_of ↔ r_has_variety
r_is_part_of ↔ r_has_part
r_is_ingredient_of
Super Properties
r_has_related_concept
Inverse
r_is_part_of

□ Functional
□ InverseFunctional
□ Symmetric
☑ Transitive

Fig. 2: Hierarchical organization of the relationships between concepts.

Since we are using these relationships to define a concept, i.e. to relate it to other concepts, the domain and range of all these relationships are set to *c_domain_concept*. The relationship hierarchy is important for backward compatibility to classic thesaurus exports, i.e. all concept-to-concept relationships can always be resolved to the most generic *has related concept* (equivalent to *related term* for thesauri) relationship. We are proposing an initial list of refined concept-to-concept relationships that can be revised in the future [4].

Furthermore, we introduce *r_domain_specific_relationship* in order to provide the opportunity to create conceptual relationships, which are only valid in a specific domain of interest. This might be useful for applications in order to filter out such specific relationships. Every such property is both sub property of *r_has_related_concept* and *r_domain_specific_relationship*..

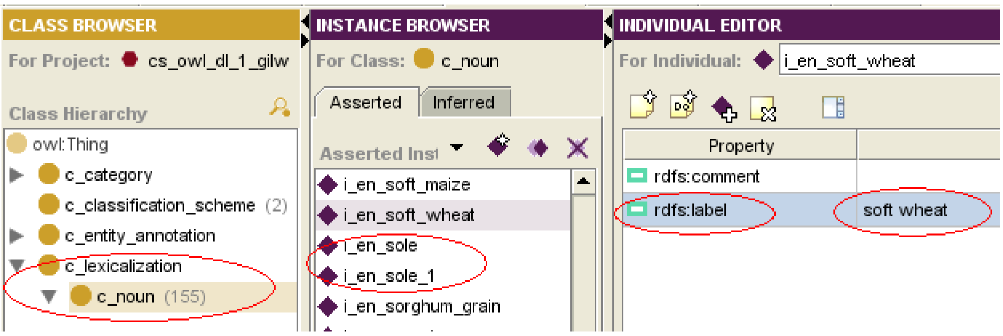
4.4 The lexicalizations

In the previous chapter, we introduced the model to create the conceptual backbone of the Concept Server. We now need to introduce lexicalizations in order to represent this structure in multiple languages. All this lexical information is subsumed by the concept *c_lexicalization*. Each term (i.e. lexicalization or word)[5] that describes a concept in a specific language is modelled as an instance of this concept.

The instance URI is composed of i_ followed by the ISO639 two-letter language code of the term, followed by the actual term (using underscores to replace spaces and special characters). If a given word form turns out to be a homonym in a given language, an additional underscore is added followed by a number, e.g., *en_sole_1* (of the shoe), *en_sole_2* (fish). The annotation rdfs:label is used to provide the actual label of the term for display purposes. Fig. 4 shows a screenshot of Protégé with a few instances of *c_lexicalization*.

The instances are actually instances of *c_noun*, a sub concept of *c_lexicalization*. This leaves the model open enough to include other forms like verbs, adjectives, etc. in the future.

[5] We will use all these three forms synonymously, i.e. it is a term/lexicalization/word that represents a concept.



For Individual: i_en_soft_wheat

Property	
rdfs:comment	
rdfs:label	soft wheat

Fig. 3: Representation of terms and URI disambiguation.

The decision to treat terms as instances rather than as annotations (e.g., rdfs:label) is mainly motivated by the fact that relations in OWL DL can be defined only between two individuals, or between an individual and a literal. In order to be able to also express relationships between terms (like translation and synonym relationships), terms must consequently be realized as instances. We will introduce such term-to-term relations shortly, but let's first have a look at how to link the terms to the domain concepts.

4.4.1 Linking lexicalizations to concepts: the Concept-to-Term Interface

Terms are related to the concept whose meaning they lexicalize via two OWL object properties, *r_has_lexicalization* and its inverse relationship, *r_means* as shown in Fig.4.

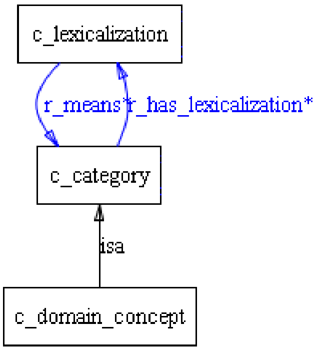


Fig. 4: Assigning terms to concepts.

We modelled the relationships on *c_category*, since we treat the lexicalizations of categories and domain concepts alike. The class *c_domain_concept* inherits the relationships from *c_category*.

Each instance of *c_lexicalization* is linked to exactly one instance of *c_category* or *c_domain_concept*. One category or domain concept will usually have several instances of *c_lexicalization* linked to it; at least one for every available language and others for synonyms and scientific names.

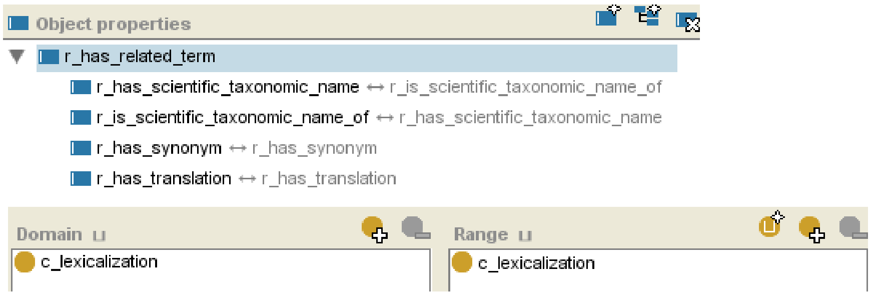
It remains to be determined what effects this will have on performance of applications that use the terminology.

4.4.2 Interlinking lexicalizations: the Term-to-Term Interface

In order to link one term (or lexicalization) to another, we introduce the property *r_has_related_term*. This property is the super property of all term to term relationships. It is important to note that this relationship does NOT correspond to the classic thesaurus relationship *related term*, since this describes a conceptual and not a term relationship. Initially, we identified three possible relationships between terms. A term can have:

- one or more translations;
- one or more synonyms per language;
- one or more scientific taxonomic names.

Fig. 5 shows the property hierarchy as modelled in Protégé. The OWL domain and range of all properties is set to *c_lexicalization*.



r_has_related_term
r_has_scientific_taxonomic_name ↔ r_is_scientific_taxonomic_name_of
r_is_scientific_taxonomic_name_of ↔ r_has_scientific_taxonomic_name
r_has_synonym ↔ r_has_synonym
r_has_translation ↔ r_has_translation

Domain: c_lexicalization
Range: c_lexicalization

Fig. 5: Hierarchical organization of the term properties.

r_has_synonym and *r_has_translation* are symmetric relationships, whereas *r_has_scientific_taxonomic_name* is a unidirectional relationship for which we introduced an inverse property. The traditional thesaurus relationships *USE* and *USE FOR* will initially be translated into *r_has_synonym* relationships when migrating AGROVOC to the new model.

This model provides highest flexibility on the lexical level. It is, for example, possible to express that an English term (corn) has a synonym in the English language (maize). The French term (maïs) describes the same concept but is a translation only of the English maize. Corn has no translation in French. Our model is able to express this, together with the ability of providing more than one translation per term.

Fig. 6 visualizes the complete picture, i.e. the link of the lexical model to the backbone structure using the ‘corn/maize’ example. The visualization is done with OntoViz (a Protégé plug-in). The upper part of the image shows the conceptual model, whereas the lower part displays the instantiations with their relationships. Since OWL DL allows only to link two instances with a relationship, in order to link a lexicalization instance to the concept it describes, we need to create an instance of the concept. The URI of a domain concept’s instance is identical to the concept name using initial i_ instead of c_. The picture shows the concept corn/maize (that has the AGROVOC termcode 12332) linked to its two English lexicalizations corn and maize via the *r_has_translation* / *r_means* relationship pair (concept-to-term interface). The two terms are then linked with the symmetric *r_has_synonym relationship* (term-to-term interface). The blue arrows in the lower part of the picture are therefore instances of the properties modelled on the concepts in the upper part of the picture.

4.4.3 Managing term variants: the ‘Term-to-String’ Interface

Terms themselves can be represented in varying ways. For example, the term University of California at Berkeley has the following variants:

- UCB (acronym);
- Cal (shortened form);
- UC Berkeley (abbreviation);
- University of California at Berkeley (official name).

A given term is related to its variants through datatype properties such as rdfs:label, and custom-defined ones such as has *acronym*, *spelling_variant*, and *abbreviation*. Following our hierarchical organization of properties, we model these relationships as sub properties of the datatype property *r_has_term_variant*.

The domain of all these properties is set to *c_lexicalization*, whereas the range is a simple string. This implies that no further relationships can be established between acronyms, abbreviations or spelling variants. So far, we do not consider this as a limitation to the lexical expressivity of our model.

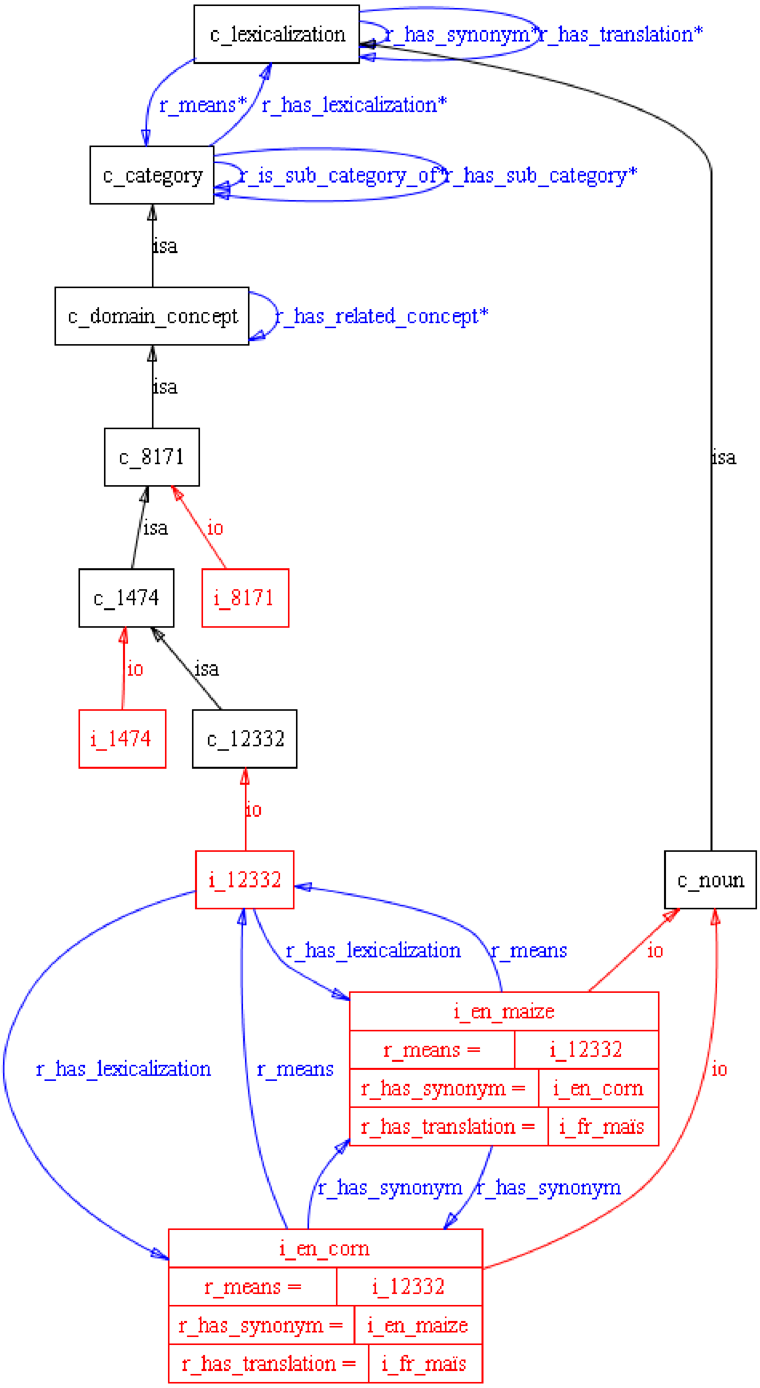


Fig. 6: overview of a concept represented with two synonyms.

4.5 Classification schemes

Another major part of our model is the concept of *c_classification_scheme*. A classification scheme is usually a shallow hierarchy (often 2 levels only) of high-level categories. A well established classification scheme in the agricultural domain is the AGRIS/CARIS classification scheme [5]. Domain concepts can be organized into a classification scheme to provide a particular view on the domain concepts. All AGROVOC terms are mapped to the AGRIS/CARIS classification scheme. Our model provides the possibility to have various classification schemes and link the categories to the domain concepts.

Fig. 9 visualizes this model on the example of the AGRIS/CARIS classification scheme. Each category is linked via the *belongs_to_scheme* / *has_category* relationship pair to at least one classification scheme it belongs to (categories can belong to several classification schemes). The *r_is_sub_category_of* / *has_subcategory* relationship pair is used for creating the hierarchy within the classification scheme. We introduce these specific relationships, because we want to keep the model open enough to use domain concepts as categories. This is why *r_domain_concept* is actually a sub class of *r_category*. The hierarchy of the domain concepts, however, might not be equivalent to a hierarchy within a particular classification scheme, so we need a specific relationship to create classification scheme hierarchies. In the example, *i_asc* represents the AGRIS/CARIS classification scheme, and *i_fao_pa* another classification scheme, called the FAO Priority Areas. The domain concept ‘Education’ (*i_2488*) is actually a category in both classification schemes, whereas the category ‘Education, Extension and Advisory Work’ (*asc:i_c*) is a specific AGRIS/CARIS subject category. The sub category relationship therefore only holds for the AGRIS/CARIS classification scheme. In reality this is reflected in the model using the property *r_has_asc_sub_category*. In the Protégé visualization tool, this has been resolved to its more generic super property *r_has_sub_category*. There will hence be a sub property or *r_has_sub_category* for each classification scheme in order to be able to model different classification scheme hierarchies on the same categories/domain concepts.

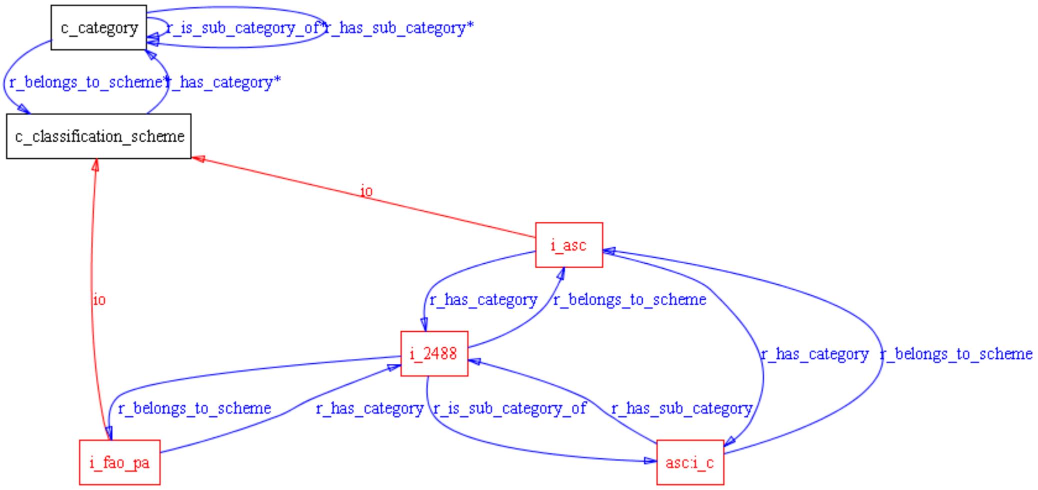


Fig. 9: Representation of Classification s Shemes and their Catagories.

4.6 Concept annotations and sub vocabularies

Concept annotations are additional information linked to domain concepts or categories, some of them coming from the traditional thesaurus world like definitions, comments, scope notes (scope of the domain concept), images and history notes (change history information). The model envisaged contains these concept annotations modelled as separate concepts linked to *c_category* or *c_domain_concept*. It furthermore contains simple annotations like date created and last updated, status and source (i.e. where the concept has been taken from).

A new notion evolving from the former ‘scope’ used in thesauri is the notion of sub vocabularies. A sub class of *c_domain_concept* called *c_geographic_concept* has been introduced in order to extract specific geographic sub structures from the Concept Server. Furthermore, *c_scientific_name* as a sub-class of *c_lexicalization* with further sub classes *c_taxonomic_name* or *c_chemical_name* will make it possible to extract specific taxonomies or sub sets of chemical names with their conceptual hierarchy and relationship structure from the Concept Server. We refer to such extractions as sub vocabularies.

4.7 Backward compatibility

One of the major concerns in moving to a new system and new formats is compatibility with current legacy systems. We cannot assume that all current AGROVOC users will suddenly stop using the traditional thesaurus. We have therefore included further annotations into our model in order to provide full backward compatibility for extracting the traditional AGROVOC thesaurus as it is used today.

5. Roadmap: Where to go from here

Having completed the basic OWL model for the AOS Concept Server, we now need to convert AGROVOC from its current relational database to the new format. A next big step is then to use the new model to completely revise the entire structure, cleaning up the hierarchy, refining relationships, providing detailed lexical information, providing definitions, comments etc. This work has to be carried out by terminologists and subject experts in various domains and languages. Protégé is an ontology editor useful for creating basic ontology models; it is, however, not applicable to above described terminology and structuring tasks, since it easily becomes too complex and cumbersome for a terminologist or subject matter expert. We therefore plan to develop a web based maintenance tool, the AOS Concept Server Workbench, which can be used by dedicated experts worldwide in order to perform the refinement and maintenance work.

Once the Concept Server is enriched with all these semantic refinements, we can explore it to enhance current applications with more semantics. An example of such an application for which we are currently specifying requirements is the new AGRIS search engine, a large bibliographical database search application currently accessing a set of about 2.5 million metadata records [6,7]. In the future the new interface shall be enhanced with semantic search result navigations and other value added features that can be provided by the AOS Concept Server.

References

[1] OWL Web Ontology Language Reference, 2004. http://www.w3.org/TR/owl-ref.
[2] Protégé Ontology Editor. http://protege.stanford.edu/.
[3] Oxford-Paravia, 2002.
[4] Proposed Concept Server relationships: http://www.fao.org/aims/cs_relationships.htm.
[5] AGRIS/CARIS classification scheme: http://www.fao.org/agris/Centre.asp?Content=DT&Menu_1ID=DT&Menu_2ID=DT1&Language=EN.
[6] AGRIS web site: http://www.fao.org/agris/.
[7] Anibaldi, Stefano, Lauser, Boris et al.: Using a full text search engine on structured metadata. Submitted to DC 2006, Mexico.