Constellation

A Prototype ISO/IEC 11179 Metadata Registry

Gramm Richardson
U.S. Department of Defense
gpricha@tycho.ncsc.mil

Elli Schwarz
SRA International
eliezer_schwarz@sra.com

Abstract—Different systems across the government, as well as in the private sector, use different country names or country codes to represent the notion of a "country" within a particular problem domain. These systems may choose to represent countries using a particular standard for county names and country codes. Often times these systems find themselves interacting with other systems that may use another standard for country representation. This makes it difficult to compare and link country-related data in a consistent fashion. We describe our work on the Constellation system using the ISO/IEC 11179 metadata standard to register the various country code sets in a common metamodel. This facilitates management, querying, updating and mapping the elements within the code sets.

Keywords: metadata, country codes, ontology

I. Introduction

There exist numerous international and national standards for country and country code representations. Some are designed to represent countries within a certain domain, such as the ITU-T e.164 [1] codes to represent telephone dialing codes for countries, or the ICAO [2] codes to represent country prefixes for airplane tail numbers. Other codes are attempts at international or national standardization, such as ISO 3166 [3] codes and NGA Geopolitical Codes [4]. Each of these standards has its own terminology and criteria for inclusion in its list.

Unfortunately, there is no unambiguous, standard definition of the term "country" [5]. Many country code sets contain entries for entities that might not be thought of as countries in the common usage of the word. A code set may consider a semi-autonomous or dependent entity to be a country in its own right, or it may include non-country placeholders such as "reserved" or "unknown". Some code sets may list a region or entity for practical, political, or diplomatic considerations, notwithstanding the entity's precise legal status.

To further complicate matters, these country lists are not static. Dependent territories may become independent, civil wars may split countries, two countries can unify, or a country may simply decide to change its official name. To keep up with changing realities, many of these code sets or standards organizations publish updates to their lists from time to time. This adds a chronological dimension to the maintenance of county code sets.

All of the above factors make it necessary to maintain these code sets together in one registry that can facilitate the management, querying and updating of these code sets. This registry can also provide a framework for tackling the challenge of mapping entities from one code set to another.

This rest of this paper describes the Constellation metadata registry system, which uses the ISO/IEC 11179-3 Edition 3 registry metamodel [6] standard to register and map country code sets. We will describe in more detail the nuances of common country code management challenges. We will discuss our approach to designing a country code registry using an OWL ontology based on the ISO/IEC 11179 metamodel, and explain how we handle updates. We will also describe our algorithm used to match countries across code sets.

II. Country Code Management Challenges

The complex nature of country data poses several challenges for its management in a registry:

- A country/geopolitical entity may have an official name and several alternate names, and some of these names may be in multiple languages.
- In some country code standards, there may be multiple code formats for each country. For example, in ISO 3166-1, each country has trigraphs, digraphs, and numeric codes, whereas other standards may have only one code format per country.
- One country may have multiple codes in one format, such as in the ICAO Nationality Marks code set. In that code set, South African aircraft can bear the nationality marks "ZS", "ZT", or "ZU".
- Multiple countries in a single code set may share the same code, such as in ITU-T e.164, where 25 countries share the country dialing code "1".
- A geopolitical entity may be a dependency of another country, like a state, territory, province, or outlying area. In ISO 3166, these entities are listed in a separate code set for dependencies, ISO 3166-2. The code set ISO 3166-1 is used for what it considers to be "top-level" (usually independent) countries. In ITU-T e.164, the dependency may be explicitly written out as part of the country name in parenthesis, as in the case of "Greenland (Denmark)". In other code sets, the administrator is ignored.
- Some code sets may have entries for regions (such as Europe or Asia) or transnational groups (such as EU, UN, or NATO) which are not traditionally thought of as countries.

- Code sets change over time. New versions of code sets might be released, and updates to individual entities in the code set, like code or name changes or even spelling corrections, might be issued.

Using an ontology can be the first step toward managing some of the above complexities. The UN FAO (Food and Agriculture Organization) ontology [7] illustrates one approach to add some degree of structure to the attributes of a country or region. It provides an OWL ontology with properties such as fao:nameOfficial and fao:nameShort for the different forms of a country name (with a language tag to indicate the language of the name), fao:validSince and fao:validUntil for valid dates for a particular country, and fao:isAdministeredBy to represent the administering country. It also provides many other additional properties of importance to countries, such as fao:sharesBorderWith, fao:predecessorOf, fao:memberOf, and other useful properties.

Additionally, SKOS [8] can be used to provide some level of abstraction to the concept of a country and its name and code representations. Using the SKOS vocabulary in OWL provides the skos:Concept class, and instances of this class can represent countries, with properties such as skos:prefLabel to represent the preferred name, and skos:altLabel to represent other names (with language tags on the literal to represent the language of the name). SKOS Mapping Properties such as skos:closeMatch and skos:broadMatch can be used on these country instances to map similar countries or country relationships. SKOS Documentation Properties such as skos:note or skos:changeNote can be used to further describe a country and changes to a country.

Methods of supplying the country code for a SKOS country concept have also been proposed in [9]. One possibility mentioned there is adding new properties for the different types of codes (iso3166:twoLetterCode or iso3166:numericalCode), or using a skos:prefLabel with a special private language tag to indicate the code type (such as using the skos:prefLabel property with "FR"@x-notation-twoletter as the literal).

SKOS-XL [10] has been proposed to further extend SKOS. It provides a class skosxl:Label to further abstract the notion of a name from the country it represents, so the name can have its own properties independent of the country itself. Thus, a date or other provenance information pertaining to the name can be accommodated [11]. The Library of Congress proposed an additional ontology, MADS/RDF [12], which builds on SKOS but provides additional classes and properties designed to model geographic and other kinds of names, as well as thesauri and other controlled value lists. The Library of Congress MARC [13] codes use the MADS/RDF ontology to represent its list of geographic areas.

Using these ontologies are a good start toward registering country code metadata in a way that manages many of the complexities listed above. However, we cannot expect that each country code set we want to register will provide their data in this fashion. Some existing code sets are provided as CSV files, with columns mapping country names to country codes, without any schema at all. Many other code sets are available only as tabular data embedded in web pages or text documents that we converted to CSV. Therefore, it is important that we allow any vocabulary or data format to be used in each particular code set, and rely on our own internal metamodel to accommodate all of these diverse data models in a uniform fashion.

Furthermore, it is important that whatever internal metamodel we use not be proprietary, and be able to handle updates to the data without losing the data contained in earlier versions. Using a standard metamodel would enable a more widespread use and understanding of our system, and would also enable it to be used by other kinds of data besides country codes, to facilitate integration with a wider range of problem domains. Maintaining a version history of the data would be of great use if the system were to integrate with other systems that contain data from an earlier point in time. To accommodate all these issues, we chose to develop the Constellation system using the ISO/IEC 11179 metamodel standard [6] to register our country code metadata. This standard, with some of our own minor extensions, enables us to build a system that can not only register countries, codes, and mappings among these countries, but also handle different versions of the various code sets and updates.

III. Implementing the ISO/IEC 11179 Metamodel in OWL for Constellation

The goals of the Constellation country code metadata registry are to represent the metadata using a consistent terminology, provide a uniform way of querying the data, manage updates without disrupting previous versions of the data, and facilitate storing relationships between data elements.

The ISO/IEC 11179 metamodel describes a variety of classes, attributes, and associations between classes useful for representing metadata about country objects. In Constellation, we implemented these classes and attributes in an OWL ontology. We represent the set of all countries in a code set as an instance of the Conceptual_Domain class, and the set of country codes in that code set as a Value_Domain. Each Value_Domain can represent one country code format (e.g., digraph or numeric). In most code sets we registered, there is only one code format for each country, so there would be one Value_Domain. In other code sets, for example ISO 3166-1, there are three code formats for each country – the trigraph, digraph, and numeric codes. Each of these formats would be a separate Value_Domain within the Conceptual_Domain for ISO 3166-1. The Value_Domain is made up of a set of Permissible_Values that contain the code (known as the "permitted value") for a country.

Each country entry is modeled as a Value_Meaning within a Conceptual_Domain. The Conceptual_Domain is thus made up of a set of Value_Meanings. Each country can contain several names (official names or other forms of the name), in multiple languages. In order to separate the concept of "country" from that of its name, we use the 11179 Designation class to represent a label or name for a country Value_Meaning. This Designation contains a "sign" property containing the actual country name, and a language identifier property to represent the language used for that name. We use a Designation_Context to describe the "acceptability" of a Designation within the context of a Conceptual_Domain. The acceptability ratings are described in ISO/IEC 11179 as being on a scale of: preferred, admitted, deprecated, obsolete and superseded. Only one Designation per language is “preferred” in a given Context; we use “admitted” to represent the other forms of the name.

Value_Meanings and Permissible_Values each contain a property for begin_date and optional end_date. This is used to represent the time period when the code set considers that value to be part of its official list. Instances of these classes without an end_date are considered to be the latest valid entry. We extended the 11179 standard to add these date fields to the Designation_Context as well. If a code set has several versions (such as when new countries are added, names or codes change, etc.) we can represent this with multiple instances of the class, each with a different date range. A diagram depicting an example of some instances of these classes can be found in Fig. 1.

The 11179 standard also provides a way to depict relationships among concepts. We use this feature to represent relationships among countries, such as when an entity is part of another country or is administered by another country. We also use this feature to represent relationships among countries that are likely to be close matches (i.e. the country named “United States” in the different code sets). These matches can be generated manually or by machine. Constellation’s semi-automated country matching algorithm [14] suggests matches based on the similarity of the names of countries in different code sets. The suggestions are then evaluated by a person who marks them as either correct or incorrect. These human judgments are recorded as rules that are used when automatically aligning entities in different code sets. We explain our approach to store these relationships in more detail later.

The Constellation system can thus be used to keep track of countries, country names, country codes, relationships among countries, and different versions of all of these pieces of information. This system has been successfully applied to over 15 different code sets, and it is easy to add additional ones. Table 1 shows some of the code sets we’ve used along with a brief description of how the code set is used.

TABLE I. Code Sets Registered in Constellation

Code Set	**Description**
International Organizations	
International Civil Aviation Organization	Aircraft nationality marks based on the Chicago Convention on International Civil Aviation, as reported to ICAO by national administrations. Used as the prefix of an aircraft tail number.
International Olympic Committee	Codes identifying the National Olympic Committees/National Teams participating in the Olympics
ISO 3166-1, ISO 3166-2	Entities which are members of the UN or one of its specialized agencies and parties to the Statute of the International Court of Justice, or registered by the UN Statistics Division. Part 2 of the standard includes dependencies of the entities in Part 1.
UN FAO Geopolitical Ontology	AGROVOC, FAOSTAT, FAOTERM - code sets used for agricultural statistics and projects purposes
UN M.49 Area Codes	Used by the United Nations for statistical purposes
U.S. Government	
Census Schedule C	Used by the US Census Bureau as well as the Army Corps of Engineers
Treasury International Capital Reporting	Designations identifying countries in data files on international portfolio capital movements reported to the US Treasury Department via the Treasury International Capital reporting system.
GSA Geographic Locator Codes	Used by US federal agencies for reporting data to the Federal Real Property Profile.
NGA Geopolitical Codes (and dependencies)	Codes for political entities in the NGA GEOnet Names Server (Formerly FIPS 10-4).
Industry	
ITU-T e.164	Recommendation that defines structure for telephone numbers, including country dialing codes
ITU-T e.212	Defines the code used in the Mobile Country Code portion of an IMSI (International Mobile Subscriber Identifier)
International Union of Railways	Standard numerical country coding for use in railway traffic. Used as the owner’s code (3rd and 4th position) of a 12-digit wagon identification number.

IV. Data Ingestion and Updates

In order to facilitate the easy ingestion of data of all types, we have two main ingestion workflows: ingesting CSV files and RDF files. For CSV, we require some basic columns such as country name (with separate columns for preferred names, and other languages), columns for dates, and columns for country codes. The column headers need to be one of several that we have pre-defined. In order to ensure that all data is ingested into the system in a uniform fashion, we first convert the CSV into a general-purpose RDF format suited for easy conversion to our OWL representation of the 11179 format. We also take RDF country data in any format (such as UN FAO data, Library of Congress MARC codes, and country currency data, each of which uses a different ontology) and convert that to the general-purpose RDF format using SPARQL 1.1 scripts custom written for each of these RDF ontologies. Once this data is in the general-purpose RDF format, it is then ingested using another SPARQL 1.1 script to convert the general-purpose RDF to RDF conforming to our OWL implementation of the 11179 metamodel.

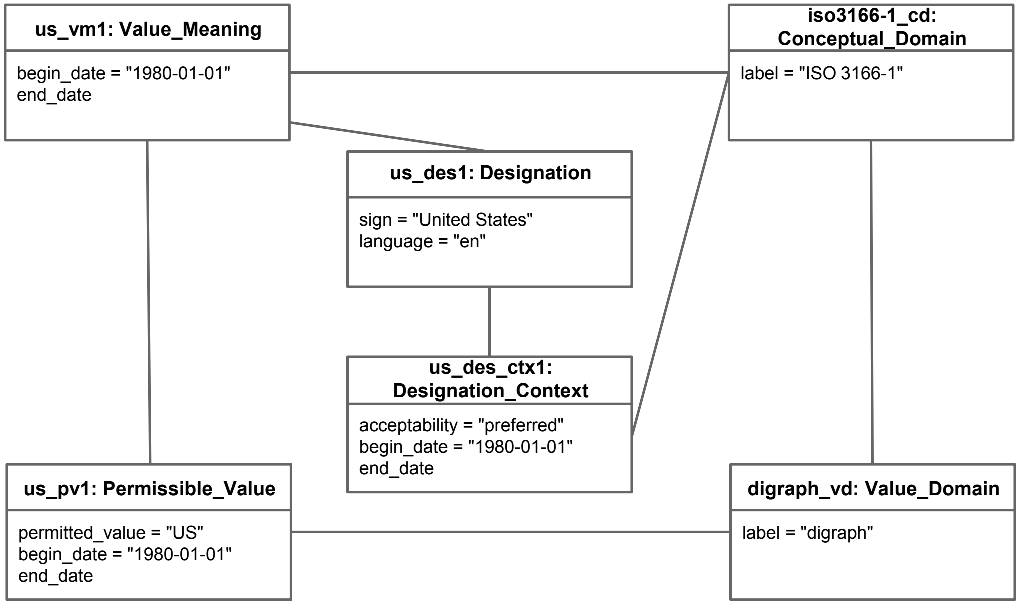
us_vm1: Value_Meaning
begin_date = "1980-01-01"
end_date
iso3166-1_cd: Conceptual_Domain
label = "ISO 3166-1"
us_des1: Designation
sign = "United States"
language = "en"
us_des_ctx1: Designation_Context
acceptability = "preferred"
begin_date = "1980-01-01"
end_date
us_pv1: Permissible_Value
permitted_value = "US"
begin_date = "1980-01-01"
end_date
digraph_vd: Value_Domain
label = "digraph"

Figure 1. UML object diagram showing an example of Constellation's use of ISO/IEC 11179 metamodel, edited for clarity

Updates to the country code sets are performed in a purely additive fashion. No statements are actually removed from the RDF store when performing update operations on country, country code, or country name data. Each of these entities may be updated separately, allowing for incremental updating of code sets. In the case of ISO 3166-1, updates are issued on an irregular basis every few months as update newsletters. The last full version of ISO 3166-1 was published in 2006, and keeping that code set current requires implementing the updates described in the newsletters. These newsletters might correct a spelling mistake in a name, change one numeric code to another, add a new country, or describe other changes. As stored in the Constellation metadata registry, country entities, codes, and country names each have begin_dates and optional end_dates associated with them. In the case of country names, the dates are associated with the acceptability of its usage in a particular Designation_Context. If a code set removes an entry, it is not actually deleted from our database, but it is marked with an end_date reflecting the date this entry was removed from the code set. Any data that has an end_date is not considered part of the current set of values but as part of an earlier version of the code set.

This use of dates on Designation_Contexts is an extension to the ISO/IEC 11179 metamodel being used in Constellation. With this extension we can record a country name change in a particular standard. For example, Libya in ISO 3166-1 has changed its name. In 2006, the country was identified in ISO 3166-1 by its official long-form English name, "the Socialist People's Libyan Arab Jamahiriya", in addition to a short form of the name. Following that country's civil war in 2011, the ISO 3166 Maintenance Agency issued an update to the country's name in a November, 2011 newsletter, which removed the long-form English name from the entry for the country.

To reflect this change in Constellation, an end_date value is added to the Designation_Context relating the former name to the code set. A new Designation_Context reflecting the name's new status (in this case, "deprecated") is added and given a begin_date. The RDF statements express the fact that a given country name ceased to be accepted and began to be deprecated on a particular date. If, rather than simply being removed, the name was changed, new statements would be added to relate the new name to the existing country and describe its usage acceptability, context, and the dates when it was used. Fig. 2 shows an RDF diagram using date fields to deprecate the old long-form name of Libya.

V. Country Matching and Relations in ISO/IEC 11179

When choosing a metamodel, there are many ways to model the relationships between countries across code sets. Our first approach was a country-centric approach, where we would define a unique URI for each country. Constellation's semi-automated country matching algorithm [14] was used to determine which countries were the same or similar across code sets. That URI would be used in all code sets as the Value_Meaning representing the notional country.

However, that approach proved problematic for many reasons. First and foremost, two different code sets may not have the same complement of values, so a given URI might not have statements in each code set. Additionally, we don't know that each standard refers to the exact same country, even if the same name is used. For example, one code set may have an entry for United States, which would include all states and dependent territories. Another code set may have separate entries for the United States, excluding territories, and separate entries for each of the territories. A code set may even include the territories as part of its definition of United States yet still have separate entries for some of these territories. For these reasons, having one URI for United States that would be shared across code sets clearly would not be appropriate, since each code set may have a slightly different interpretation of what is indicated by the country name.

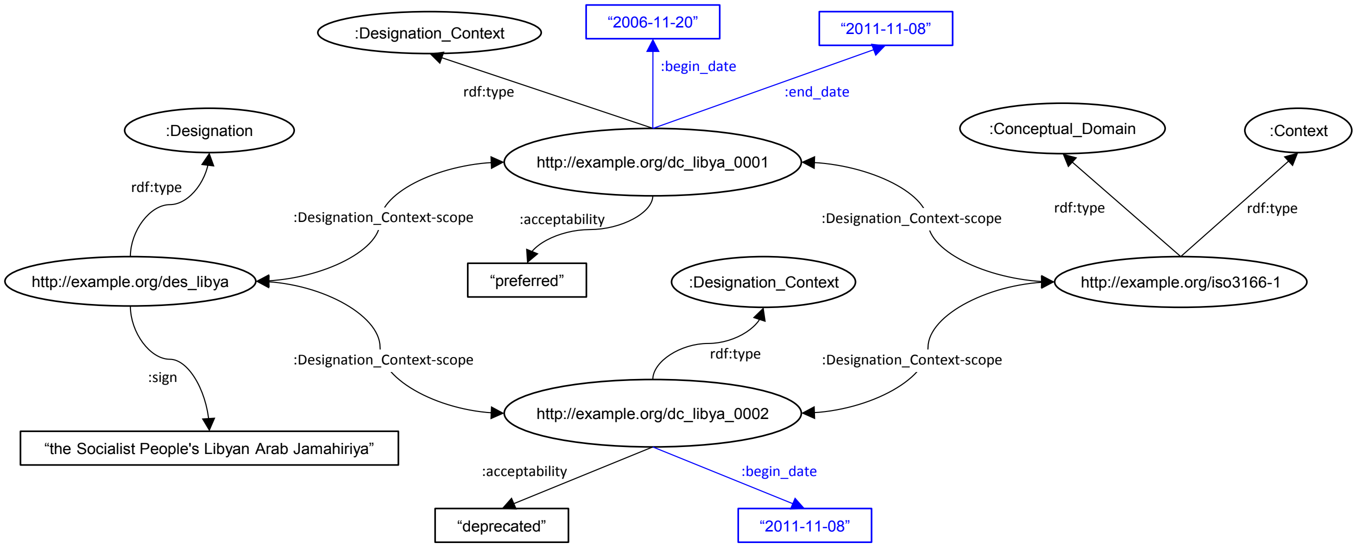
:Designation_Context
"2006-11-20"
"2011-11-08"
rdf:type
:begin_date
:end_date
:Designation
http://example.org/dc_libya_0001
:Conceptual_Domain
:Context
rdf:type
:Designation_Context-scope
:acceptability
:Designation_Context-scope
rdf:type
rdf:type
http://example.org/des_libya
"preferred"
:Designation_Context
http://example.org/iso3166-1
rdf:type
:Designation_Context-scope
:Designation_Context-scope
:sign
http://example.org/dc_libya_0002
"the Socialist People's Libyan Arab Jamahiriya"
:acceptability
:begin_date
"deprecated"
"2011-11-08"

Figure 2. RDF diagram showing how Constellation handles deprecated country names

Another example of this problem is that in some standards the country China includes Hong Kong and Macau, whereas in other standards each one has its own disjoint representation. If we had one URI for China, there would be ambiguity as to what is meant by that URI—is that the URI of all of China and its dependencies, or of just mainland China? Another example is Sudan and South Sudan—one code set might have a separate entry for South Sudan (which recently became independent from Sudan), as well as for Sudan itself. However, another code set may contain one entry for Sudan, meaning both Sudan and South Sudan. This may be based on the different dates of the code set, if one code set wasn't yet updated after South Sudan's independence, or the code set may not recognize South Sudan's independence.

Another issue with using a unique URI for each country is that two code sets may use completely different names for the same country. The reason that different names may be used in a given code set may be politically motivated. The country identified in the international ISO 3166 standard as "Myanmar" is referred to by the name "Burma" in official U.S. Government documents. The entity identified as "Taiwan, Province of China" in ISO 3166 is called "Chinese Taipei" by the International Olympic Committee. Although these entries have different names, technically they are referring to the same entity.

In all of the above cases, it is debatable whether it makes sense to use the same URI for the notional country across all code sets. Since each code set has its own idea of what an entry actually refers to, it is very difficult to determine if two code sets are using a country name in exactly the same way [15]. Therefore, we decided that each code set would use its own set of URIs (unique Value_Meanings) for its own values. Instead of relying on a common URI to map countries from one code set to another, we use 11179 Relations, which provide a way to link countries across code sets. For the names of the relationships, we use the SKOS vocabulary terms where appropriate (such as skos:closeMatch or skos:broadMatch). Use of skos:exactMatch and owl:sameAs was avoided for the same reasons we chose not to use the same URI. The 11179 standard doesn't provide date properties for these Relations, but we can add these fields to keep track of versions just as we did for countries above.

VI. Querying Challenges Using the ISO/IEC 11179 Metamodel

The generic nature of the 11179 metamodel adds a great deal of complexity and abstraction to the representation of the data. This poses a challenge for querying, since even a simple query getting all country codes for a given country name can involve traversing a large amount of RDF, resulting in a lengthy and difficult to read SPARQL query. The 11179 Relations which we used to link related concepts to each other also adds a great deal of complexity and extra statements. This is because the 11179 relations model is best suited to scale to ternary, quaternary, and higher-order relations, but it adds additional overhead when dealing with simpler binary relations, as will be explained below.

We attempted to provide shortcuts in the data we ingested, but this resulted in losing some of the benefits of 11179, particularly when it came to updates. We were able to simplify querying using shortcuts such as adding an rdfs:label directly to a Value_Meaning, instead of using Designations with a "sign" property, eliminating an extra statement traversal. However, this did not allow for dates to be provided for the label itself. Eliminating Designation_Context and adding alternate name forms directly in the Designation posed a similar problem managing the acceptability ratings. Since we don't want to actually delete any data from our system, in order to keep previous versions of data we needed these abstractions of Designation and Designation_Context, so we can maintain dates and acceptability ratings on the Value_Meaning and Designation_Context objects independently.

We experienced similar problems using shortcuts for 11179 Relations. In the 11179 metamodel, traversing the graph from one Concept to another Concept related by a Relation requires stepping through three intermediate objects rather than just a single predicate. We attempted to add convenience predicates (such as skos:broader) for these Relations to provide only one statement linking the two Concepts. As a result of this simplification, the SPARQL queries using the convenience predicates were much shorter and easier to read, but the convenience predicates lacked much of the descriptive power of the 11179 Relations. Fig. 3 shows a simple example of the way that relationships are represented in the 11179 metamodel, compared to how they are represented in SKOS.

Due to our issues with shortcuts, we determined that they were not a suitable approach for Constellation, and as a result we have some long, complex queries. We are exploring the use of SPIN [16] functions to pre-define query patterns for some of the complex parts of the 11179 metamodel. We would then call these functions in our queries. Although this may not improve query efficiency (unless the SPARQL implementation incorporates some efficiencies or caching for the SPIN functions), it should help a great deal with query readability and maintainability.

VII. Conclusion

We have shown how the 11179 metamodel can be used to register, query, and track updates to country code data. We have also demonstrated how 11179 can be used to track relationships among countries, such as country group memberships and administration. We have also shown how we can link similar countries together using 11179 relationships.

Applications of our work extend beyond just country code mapping. We have used it to model country currencies, and even to store thesaurus information, including taxonomies (such as the FAA Aviation Safety Thesaurus and the ETDE/INIS Joint Thesaurus of nuclear energy terminology). We are currently experimenting with applying this research to automated compliance challenges. The 11179 metamodel is useful for registering the metadata related to system policies and rules. We can then track changes to these rules, and relationships between different rules, in the same way we track changes and relationships in country code data. The Constellation registry, using the 11179 metamodel, can thus be used to address these challenges across a variety of metadata.

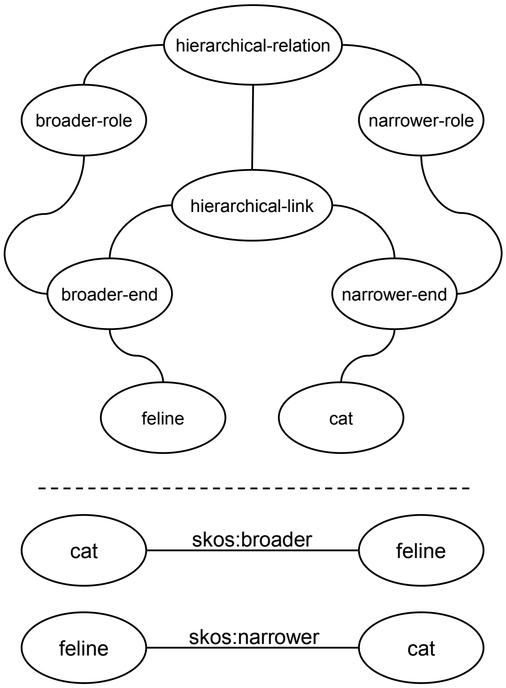

Figure 3. Top - broader and narrower relations represented via the 11179 metamodel. Bottom - broader and narrower relations represented in SKOS.

References

[1] "Operational Bulletin No. 991 Annex - List of ITU-T Recommendation E.164 assigned country codes." ITU-T Telecommunication Standardization Bureau, 01-Nov-2011.

[2] "Aircraft Nationality Marks and Common Marks as notified to ICAO." International Civil Aviation Organization - Air Navigation Bureau (ANB), 08-Jun-2009.

[3] "ISO 3166-1 - Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes." International Organization for Standardization, 15-Nov-2006.

[4] T. Palmer, "Geopolitical Entities and Codes (Formerly Federal Information Processing Standards Publication 10-4: Countries, Dependencies, Areas of Special Sovereignty, and Their Principal Administrative Divisions)." National Geospatial-Intelligence Agency, Apr-2010.

[5] "In quite a state," *The Economist*, vol. 395, no. 8677, pp. 62–63, 10-Apr-2010.

[6] R. Gates and K. Keck, Eds., "ISO/IEC FDIS 11179-3:2012(E) - Information technology — Metadata registries (MDR) — Part 3: Registry metamodel and basic attributes." ISO/IEC JTC1, 08-Jan-2012. Available: http://metadata-standards.org/Document-library/Documents-by-number/WG2-N1651-N1700/WG2N1675_Editors-Final-Sneak-Peek-FDIS_11179-3.pdf

[7] "FAO Geopolitical ontology," *Food and Agriculture Organization of the United Nations*, 18-Jan-2011. [Online]. Available: http://www.fao.org/countryprofiles/geoinfo/geopolitical/resource/.

[8] A. Miles and S. Bechhofer, Eds., "SKOS Simple Knowledge Organization System Reference." The World Wide Web Consortium, 18-Aug-2009.

[9] J. Voss, "Encoding changing country codes for the Semantic Web with ISO 3166 and SKOS," in *Metadata and Semantics*, New York, NY: Springer Science+Business Media, LLC, 2009, pp. 211–221.

[10] A. Miles and S. Bechhofer, Eds., "SKOS Simple Knowledge Organization System eXtension for Labels (SKOS-XL) Namespace Document - HTML Variant." The World Wide Web Consortium, 18-Aug-2009.

[11] B. DuCharme, "Improve your taxonomy management using the W3C SKOS standard," *IBM developerWorks*, 10-May-2011. [Online]. Available: http://www.ibm.com/developerworks/xml/library/x-skostaxonomy/index.html. [Accessed: 24-May-2012].

[12] MODS/MADS Editorial Committee, Ed., "MADS/RDF Primer." Library of Congress, 10-May-2012.

[13] "MARC List for Geographic Areas," *Library of Congress Authorities and Vocabularies*, 26-Apr-2011. [Online]. Available: http://id.loc.gov/vocabulary/geographicAreas.html.

[14] G. Richardson, "Automated Country Name Disambiguation for Code Set Alignment," *Research and Advanced Technology for Digital Libraries*, vol. 6273, pp. 498–501, Sep. 2010.

[15] H. Halpin, P. Hayes, J. McCusker, D. McGuinness, and H. Thompson, "When owl:sameAs Isn't the Same: An Analysis of Identity in Linked Data," in The Semantic Web – ISWC 2010, vol. 6496, P. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Pan, I. Horrocks, and B. Glimm, Eds. Springer Berlin / Heidelberg, 2010, pp. 305–320

[16] H. Knublauch, Ed. "SPIN - SPARQL Syntax" The World Wide Web Consortium. [Online]. Available: http://www.w3.org/Submission/spin-sparql/