Department of
Information Engineering
and Computer Science DISI

UNIVERSITY
OF TRENTO - Italy

DISI - Via Sommarive 14 - 38123 Povo - Trento (Italy)
http://www.disi.unitn.it

TOWARDS THE INTEGRATION OF KNOWLEDGE ORGANIZATION SYSTEMS WITH THE LINKED DATA CLOUD

Vincenzo Maltese and Feroz Farazi

May 2011

Technical Report # DISI-11-466

Also: In the proceedings of the UDC seminar 2011.

Vincenzo Maltese
DISI - University of Trento, Trento, Italy

Feroz Farazi
DISI - University of Trento, Trento, Italy

Towards the Integration of Knowledge Organization Systems with the Linked Data Cloud

Abstract: In representing the shared view of all the people involved, building a Knowledge Organization System (KOS) from scratch is extremely costly, and it is therefore fundamental to reuse existing resources. This can be done by progressively extending the KOS with knowledge coming from similar KOS and by promoting interoperability among them. The linked data initiative is indeed fostering people to share and integrate their datasets into a giant network of interconnected resources. This enables different applications to interoperate and share their data. However, the integration should take into account the purpose of the datasets and make explicit the semantics. In fact, the difference in the purpose is reflected in the difference in the semantics. With this paper we (a) highlight the potential problems that may arise by not taking into account purpose and semantics, (b) make clear how the difference in the purpose is reflected in totally different semantics and (c) provide an algorithm to translate from one semantic into another as a preliminary step towards the integration of ontologies designed for different purposes. This will allow reusing the ontologies even in contexts different from those in which they were designed.

Keywords: Knowledge Organization Systems; integration; Linked Data; semantics

1. Introduction

KOS, to serve their goals in indexing and providing broad access to information content, need to express the knowledge about the domain at hand. According to the specific purpose they have to serve, there is a large spectrum of KOS, ranging from simple dictionaries to classifications, thesauri and subject headings.

In representing the shared view of all the people involved, building a KOS from scratch is extremely costly, and it is therefore fundamental to reuse existing resources at least partially meeting the organizational requirements. This can be done by progressively extending the KOS with knowledge coming from similar KOS and by promoting interoperability among them. This is typically achieved by establishing informal (purely syntactic) or formal (semantic) correspondences, called mappings, between their terms/concepts. A good survey of the projects that have dealt with mappings between KOS can be found in (Giunchiglia, Soergel, Maltese & Bertacco, 2009). They include *inter-alia* CARMEN, Renardus (Koch, Neuroth, & Day, 2003) and OCLC initiatives (Vizine-Goetz, Hickey, Houghton & Thompson, 2004). A reference scheme is sometimes used, mainly DDC (Nicholson, Dawson & Shiri, 2006) and LCSH (Whitehead, 1990) (O'Neill & Chan, 2003). Some of them are based on fully manual approaches, while others rely on automatic tools for the identification of an initial set of correspondences to be manually validated and augmented. In this case, it is fundamental to minimize the time required for validation. A technique based on the computation of the minimal mapping (that unique minimal subset of the correspondences such that all the others can be efficiently computed from them), its validation and expansion is described in (Maltese, Giunchiglia & Autayeu, 2010). On this respect open source tools for their computation, visualization and validation can be downloaded from http://semanticmatching.org/.

The linked data initiative (Bizer, Heath & Lee, 2009) goes exactly in the direction of fostering people to share and integrate different datasets into a giant network of interconnected resources. The fundamental property required is that the datasets must be codified in RDF, a simple generic enough representation language to express triples of the form source-relation-target. Correspondences are established through same-as links between individuals and through equivalent-class links between classes/concepts in different datasets. Potentially, this is at the same time relatively simple and powerful since it enables different applications to interoperate and share their data. However, integrating different datasets is intrinsically difficult because they may differ in purpose, structure, terminology and language used, coverage, level of formality and conceptualization (Giunchiglia, Soergel, Maltese & Bertacco, 2009). Even more problematic, there is often no clear explicit semantics associated to the data. Expressing something in a language - regardless if it is natural language, RDF or SKOS - provides the syntax, i.e. the grammatical rules to write correct sentences, but it is not enough to provide the semantics, i.e. the precise meaning of the sentences. In this paper we show, with some concrete examples, how the difference in the purpose is reflected in the difference in the semantics and how, without taking into account these differences, it is at least inappropriate to integrate them. Making explicit the semantics allows converting schemes into formal ontologies (see for instance (Giunchiglia, Marchese & Zaihrayeu, 2007), (Giunchiglia & Zaihrayeu, 2008), and (Giunchiglia, Zaihrayeu, & Farazi, 2009)) and consequently automating complex tasks. To do that it is fundamental to distinguish between schemes built to classify documents (the main goal of KOS), called classification ontologies, from schemes built to generically describe a domain, called descriptive ontologies (Giunchiglia, Dutta & Maltese, 2009).

However, integrating undifferentiated ontologies into a single data cloud can be potentially dangerous or at least superficial. Before integrating two or more schemes one should establish the purpose of the final ontology. If the purpose is to classify, one should codify both schemes into classification ontologies. Conversely, if the goal is to describe a domain, one should codify them into descriptive ontologies. This may require converting the schemes from one semantic to another. Along this paper we explain how this can be done. While the translation from descriptive to classification ontologies can be fully automated, the inverse translation always requires manual intervention. This is due to the underspecified relations between nodes and has precise implications on the applications that can make use of them. For this reason, storing ontologies as descriptive would serve both the purposes (describing and classifying) with no loss of information and with clear advantages in maintenance. Particularly, during the definition and the translation process we emphasize the necessity to pay attention to the distinction between transitive and non-transitive relations.

The difference between the two semantics is not currently stressed by representation languages. Actually, to overcome expressiveness limitations of SKOS, recent updates allow defining non-transitive NT/BT relations. Conversely, subset/superset relations are always transitive. This may cause an undesired mixture of semantics. In this respect, one of the limitations of RDF is that classes/concepts can be represented as individuals, thus increasing the ambiguity.

To sum up, with this paper we (a) highlight the potential problems that may arise from indiscriminate attempts of integrating datasets in an undifferentiated tangle of ontologies with no clear purpose and semantics, (b) make clear how the difference in the purpose is reflected in totally different semantics and (c) provide an algorithm to translate from one semantics into another as a preliminary step towards the integration of ontologies designed for different purposes. This will allow reusing the ontologies even in contexts different from those in which they were designed.

The remainder of this paper is organized as follows. Section 2 and 3 describe descriptive and classification ontologies respectively, their purpose and some of the limitations of the languages used for their representation. Section 4 provides a simple algorithm to convert descriptive into classification ontologies. Section 5 provides a concrete use case of how to deal with ontologies with different semantics and how to use them in practice for different purposes. Section 6 concludes the paper by summarizing the main message.

2. Descriptive ontologies

Schemes built to describe a domain, called descriptive ontologies, are in real world semantics (Giunchiglia, Dutta & Maltese, 2009) where terms at nodes represent either individuals or classes of real world entities.

Consider the example in Fig. 1. It shows a scheme to describe some organizations and where they are located. White nodes represent classes while black nodes represent individuals. The first label at the nodes is the preferred term. Additional synonymous terms are eventually provided separated by semicolon. Arrows represent relations and the direction of the arrows indicates the direction of the relation. For instance, the term *country* (in the sense of the territory occupied by a nation) denotes all the real world countries, while the term *Italy* denotes Italy the country. Under this semantics, there is an *instance-of* relation between *country* (the class) and *Italy* (the individual). Other typical relations include *is-a* between classes (connecting a subclass to a class) and *part-of* between classes or instances.

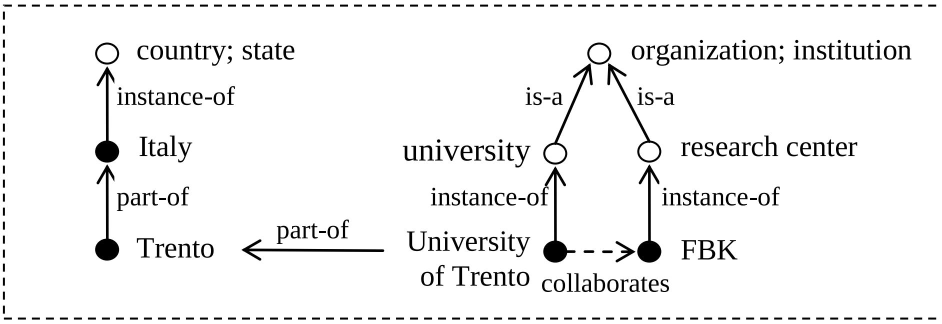

Fig. 1 – An example of descriptive ontology

These schemes represent what we know about the domain and can be used to reason about it. Typical queries can include for instance:

1. Give me all the countries
2. Give me all the organizations
3. Give me all the organizations located in Italy

By exploiting the instances of the class *country*, the output of the first query is clearly {*Italy*}. The output of the second one should consider also the classes that are more specific than *organization* (*university* and *research center*) and therefore is {*University of Trento*, *FBK*}. To respond to the third query, one should also exploit the part-of relations between the entities and therefore, by assuming the part-of as transitive, it should return {*University of Trento*}.

In order to automate tasks, one should convert these schemes into formal (descriptive) ontologies. We use Description Logics (DL) (Baader, Calvanese, McGuinness, Nardi & Patel-Schneider, 2002). With the conversion:

- classes correspond to concepts
- instances correspond to individuals in the domain of interpretation
- *is-a* relations are translated into logical subsumption (⊑)
- other relations correspond to DL roles

Specifically, the scheme in Fig. 1 can be codified with the following TBox and ABox:

TBox	**ABox**
university ⊑ organization	country(Italy)
research-center ⊑ organization	university(University of Trento)
	research-center(FBK)
	part-of(Trento, Italy)
	part-of(University of Trento, Trento)
	collaborates(University of Trento, FBK)

Deciding on the transitivity of the relations is an important choice in modelling. In DL there are ways to enforce transitivity of roles (Horrocks & Sattler, 1999), e.g. for the part-of relation above (subsumption itself is assumed to be transitive). However, this might be problematic. There are several works about the transitivity of part-of relations. As stated in (Varzi, 2006), the generic part-of relation is always transitive. However, if we start distinguishing about the different kinds of part-of then they might lose the transitivity property, in particular when we try to combine them together. The typical example is the handle that is part of the door that is part of the house that after a chain of other part-of relations ends to be part of the universe.

In our example, we may say that the part-of relation between *Italy* and *Trento* is an administrative part-of relation, while the one between *Trento* and *University of Trento* can be characterized as being a topological part-of or even just a generic associative relation. In fact, it is actually the building hosting the university as institution that is located in Trento, not the institution as such. The response to the query (3) will highly depend on whether or not we consider the combination of these relations to be transitive.

To publish the ontology as linked data, one should encode it into RDF. A fragment of a possible translation would look as follows:

```
<!--Classes-->
<rdfs:Class rdf:about="#research_center">
    <rdfs:subClassOf rdf:resource="#organization"/>
</rdfs:Class>
<rdfs:Class rdf:about="#university">
    <rdfs:subClassOf rdf:resource="#organization"/>
</rdfs:Class>
<!--Properties-->
<rdf:Property rdf:about="#collaborates"/>
<rdf:Property rdf:about="#part_of"/>
<!--Individuals-->
<administrative_division rdf:about="#Trento">
    <part_of rdf:resource="#Italy"/>
</administrative_division>
<university rdf:about="#University_of_Trento">
    <collaborates rdf:resource="#FBK"/>
    <part_of rdf:resource="#Trento"/>
</university>
```

In this representation the constructs rdfs:Class and rdf:Property are used to encode classes and properties, respectively.

By linking the RDF code to a knowledge base such as WordNet (Miller, 1998), we can disambiguate the meaning of the classes university and organization to university sense #3 (*a large and diverse institution of higher learning created to educate for life and for a profession and to grant degrees*) and organization sense #1 (*a group of people who work together*), respectively. This assignment is consistent with the code above since in WordNet university sense #3 *is-a* organization sense #1.

Nevertheless by construction RDF cannot prevent the modeller to add a new relation between the class university with a new class artefact (*a man-made object taken as a whole*) to enforce that a university as artefact can be part-of a location, e.g. that the University of Trento is part-of Trento. This makes the meaning of the class university ambiguous. In fact, university as artefact would rather match with university sense #2 (*establishment where a seat of higher learning is housed, including administrative and living quarters as well as facilities for research and teaching*). This could be prevented by making artefact and organization disjoint, but RDF does not support the use of disjointness. An immediate consequence is that if we have two RDF ontologies, one of them codifying university as organization and the other codifying university as artefact, nothing prevents to merge them into one single class when integrating them.

Another well-known limitation of RDF is that, even if it distinguishes between classes and instances, a class can be treated as an instance (Brickley D. & Guha R.V., 2004). Moreover, in RDF transitivity cannot be enforced at the level of instances. In the example, this pertains in particular the transitivity of the part-of between University of Trento and Trento.

3. Classification ontologies

Schemes built to classify documents, called classification ontologies, are in classification semantics (Giunchiglia, Dutta & Maltese, 2009) where terms at nodes always represent classes of documents. In this respect in these schemes the instances are the documents themselves.

Consider the example in Fig. 2. It shows a thesaurus built with the purpose of classifying documents by country and by organization. Similarly to Fig. 1, labels at the nodes denote the preferred term, optionally followed by synonymous terms separated by semicolon, while arrows represent relations. Documents at nodes are denoted with the letter *d* followed by an index. NT/BT relations (where the direction of each arrow goes from the narrower to the broader term) - being hierarchical - mainly serve the purpose of facilitating the indexing and search tasks, while the RT relations - being associative - are mostly used for navigational purposes or for query expansion (to increase recall). In particular, following NT relations will allow identifying progressively more specific concepts (thus decreasing the extension, i.e. the set of documents about the concept) while following the inverse direction using the BT relation will allow identifying progressively more general concepts (thus increasing the extension).

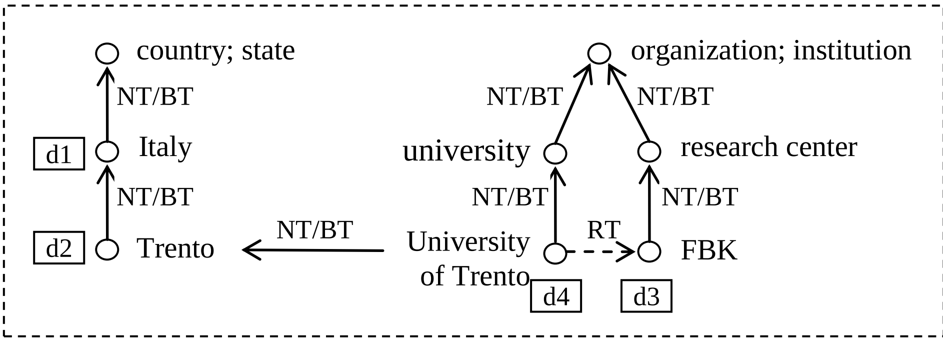

Fig. 2 – An example of classification ontology

In the example, the term *country* denotes all documents about countries. Under this semantics NT/BT relations represent subset/superset relations (where NT and BT are one the inverse of the other). For instance, if the node *Italy* is connected to *country* through a BT relation, then the semantics of the node *Italy* is the set of documents about Italy the country.

We can use the scheme to classify documents and to search or browse a document collection. Typical queries can include for instance:

1. Give me all documents about Italy
2. Give me all documents about countries

What is the output of the first query? Actually this is a bit tricky. Assume we always apply query expansion, but without using RT relations. Somebody may argue that it should correspond to the set {d1, d2, d4}, while some others may rather propose {d1, d2}. This depends on the nature of the NT relation between *Trento* and *University of Trento*. If transitive, then the output should be the former, otherwise the latter. This is even more evident by looking at the second query. One may expect as output the set {d1}, but actually according to the transitivity or not of the NT relations below the node *country*, one may have {d1, d2, d4} (if all the relations are transitive), {d1, d2} (if the relation between *Trento* and *University of Trento* is not transitive) or {d1} (if none of them is transitive). For what said in the previous section, if not transitive they should not even be encoded as NT relations, but rather as RT relations.

To make explicit the intended semantics and automate tasks one should provide a formal representation of the schema. Once again we use DL to convert the schema into the corresponding formal (classification) ontology. With the conversion:

- classes correspond to concepts
- documents correspond to individuals in the domain of interpretation
- transitive NT/BT relations are translated into logical subsumption ($\sqsubseteq$)
- RT and non-transitive NT/BT relations correspond to DL roles

Specifically, assuming all NT/BT to be transitive, the scheme in Fig. 2 can be codified with the following TBox and ABox:

TBox	**ABox**
university $\sqsubseteq$ organization	italy(d1)
research-center $\sqsubseteq$ organization	trento(d2)
university-of-trento $\sqsubseteq$ university	fbk(d3)
fbk $\sqsubseteq$ research-center	university-of-trento(d4)
italy $\sqsubseteq$ country	
trento $\sqsubseteq$ italy	
university-of-trento $\sqsubseteq$ trento	
university-of-trento $\sqsubseteq \exists$RT.fbk	

As it can be noticed, those elements of the scheme that in a formal descriptive ontology would be codified as individuals (e.g. Trento, see Fig. 1) in formal classification ontologies correspond to concepts (denoting the set of documents about Trento, see Fig. 2).

To publish the classification ontology above we can use SKOS. If we want to publish the scheme only (without the documents) we can use in particular the RDF exchange syntax, for instance as in the fragment below:

```
<skos:Concept rdf:about="#research_center">
    <skos:broaderTransitive rdf:resource="#organization"/>
<skos:Concept>
<skos:Concept rdf:about="#university">
    <skos:broaderTransitive rdf:resource="#organization"/>
</skos:Concept>
<skos:Concept rdf:about="#Trento">
    <skos:broader rdf:resource="#administrative_division"/>
    <skos:broaderTransitive rdf:resource="#Italy"/>
</skos:Concept>
<skos:Concept rdf:about="#FBK"/>
<skos:Concept rdf:about="#University_of_Trento">
    <skos:broaderTransitive rdf:resource="#Trento"/>
    <skos:related rdf:resource="#FBK"/>
</skos:Concept>
```

As it can be noticed, consistently with the TBox above, in SKOS both real world classes and individuals are codified as concepts, or better as instances of skos:Concept. However, since there is no distinction between concepts and instances, we cannot represent corresponding documents in SKOS.

Similarly to RDF, there is no support for disjointness in SKOS (Miles A. & Bechhofer S, 2009). On the other hand, differently from RDF, transitivity can be enforced using skos:broaderTransitive or skos:narrowerTransitive properties, while for non transitive part-of (e.g. membership or containment) skos:broader or skos:narrower can be used.

4. Converting, integrating and reusing ontologies

From what was discussed in the previous two sections, it should now be clear how the difference in the purpose is reflected in two totally different semantics - in terms of individuals, classes and relations - and therefore it is obviously not appropriate to integrate a classification with a descriptive ontology (exactly because the semantics is different). However, this does not mean that it cannot be done, but that we rather need to preliminary convert them such that they have the same semantics. If the purpose is to classify, one should codify both schemes into classification ontologies. Conversely, if the goal is to describe a domain, one should codify them into descriptive ontologies.

The conversion from a descriptive to the corresponding classification ontology can be done as follows:

- convert instances into classes
- convert instance-of, is-a and transitive part-of into NT/BT relations
- convert other relations into RT relations

Note that this is in line with what was postulated by Ranganathan when he says that hierarchies are constructed on the basis of *genus-species* (is-a and instance-of) and *whole-part* (part-of) relations (Ranganathan, 1967).

As it can be noticed, modulo the indexed documents, the classification ontology in Fig. 2 corresponds to the conversion of the descriptive ontology in Fig. 1.

The translation process can be easily automated. However, with the translation we have a clear loss of information. Real world classes and instances collapse into document classes. Similarly, instance-of, is-a and transitive part-of relations become undifferentiated hierarchical relations, while all other relations become associative relations. For this reason it is clear that the opposite conversion cannot be automated, but it strictly requires manual intervention:

- each class has to be mapped to either a real world class or instance
- each NT/BT relation (assuming all of them to be transitive) has to be converted to either an instance-of, is-a or transitive part-of
- each RT relation has to be codified into an appropriate real world associative relation

Distributing schemes as descriptive ontologies ensures maximum reusability. In fact, this would directly serve those applications that need to reason on a domain and at the same time it would require a minimum effort to convert them into classification ontologies when needed. When a scheme is available as classification ontology, a significant human effort will be necessary instead to reconstruct its real world version.

5. A concrete use case

In our research group we have been developing a platform to serve multi-purpose applications ranging from indexing, searching and browsing of real world entities as well as to search, browse and classify documents. For instance, it can respond to the following questions at the same time:

1. Give me all the organizations located in Italy
2. Give me all the lakes in Trento with an altitude greater than 500 m
3. Give me all documents about Italy
4. Give me all documents about countries
5. Give me all documents about the University of Trento

It is clear that in order to serve these diverse applications we need both descriptive and classification ontologies. However, to minimize maintenance costs we codify only the real world version of the semantics and efficiently compute at run time (it is a matter of seconds), whenever needed, the corresponding classification semantics. This is done by computing the corresponding transitive closure (i.e. the set of all possible relations derived because of the transitivity). According to the application, we can specify if we want to include or not part-of relations between entities in the closure (i.e. depending on whether we want to enforce their transitivity or not). Of course the time necessary for the computation of the closure heavily depends on the amount of available relations between entities.

The current core is constituted by the GeoWordNet ontology (Giunchiglia, Maltese, Farazi & Dutta, 2010). GeoWordNet is a multilingual descriptive ontology obtained from the integration of GeoNames[1] with the English WordNet and the Italian section of MultiWordNet[2]. Overall it currently contains 110,459 classes, 6,927,078 instances, 6,927,078 instance-of, 89,266 is-a and 5,325 transitive part-of relations. It also includes 98,907 associative relations. Since we consider them as non-transitive, among them we include the member meronym and substance meronym (part-of) relations. A substantial fragment of GeoWordNet is freely available both in RDF and WordNet format[3].
Fig. 3 provides a glimpse of the kind of information available in GeoWordNet in its descriptive version.

The platform is extensible, in the sense that new sources can be integrated with existing data. However, to integrate knowledge coming from classification ontologies, they need to be preliminary processed in order to reconstruct their descriptive version. This is what has been done for instance to enrich GeoWordNet with the knowledge about the spatial domain defined in (Dutta, Giunchiglia & Maltese, 2011).

---

[1] http://www.geonames.org
[2] http://multiwordnet.fbk.eu
[3] http://geowordnet.semanticmatching.org

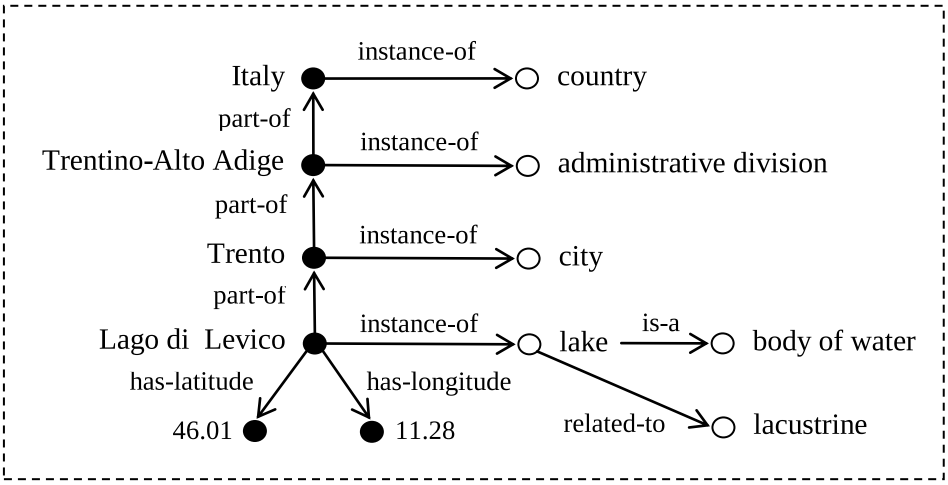

Fig. 3 – A fragment of GeoWordNet in its descriptive version

6. Conclusions

In this paper we have shown, with concrete examples, why it is fundamental to take into account the original purpose of the schemes before integrating them and how the difference in the purpose is reflected in the difference in the semantics. In particular, it is essential to distinguish between schemes built to describe a domain from schemes built to classify documents in a domain.

RDF and SKOS representation languages typically tend to leave implicit the semantics, do not support disjointness and do not force the modeller to distinguish between classes and individuals. Moreover, in RDF we cannot enforce transitive properties for relations. This makes integration tasks difficult.

We also provided an algorithm to translate from one semantics into another as a preliminary step towards the integration of ontologies designed for different purposes. We have shown, with a concrete use-case, that to maximize reusability and minimize maintenance costs, one should distribute and store the ontologies in their descriptive version.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 231126 LivingKnowledge: LivingKnowledge – Facts, Opinions and Bias in Time. We want to thank Fausto Giunchiglia for his constant guidance in our early years of study.

References

Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; Patel-Schneider, P. F. (2002). The Description Logic Handbook: Theory, Implementation and Applications. *Cambridge University Press*, 2002.

Bizer, C.; Heath, T.; Lee, T. B. (2009). Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3), pp. 1-22.

Brickley, D.: Guha, R. V. (2004). RDF Vocabulary Description Language 1.0: RDF Schema, (Editors), *W3C Recommendation*.

Dutta, D.; Giunchiglia, F.; Maltese, V. (2011). A facet-based methodology for geo-spatial modelling. *Proceedings of the 4th Int. Coference on GeoSpatial Semantics (GEOS)*, vol 6631, pp. 133-150.

Koch, T.: Neuroth, H.; Day, M. (2003). Renardus: Cross-browsing European subject gateways via a common classification system (DDC). *Proceedings of IFLA meeting*, pp. 25–33.

Giunchiglia, F.; Marchese, M.; Zaihrayeu, I (2006). Encoding Classifications into Lightweight Ontologies. *Proceedings of ESWC*, vol. 4011, pp. 80-94.

Giunchiglia, F.; Marchese, M.; Zaihrayeu, I (2007). Encoding Classifications into Lightweight Ontologies. *Journal on Data Semantics VIII*, vol. 4380, pp. 57-81.

Giunchiglia, F.; Zaihrayeu, I. (2008). Lightweight ontologies. In *Encyclopedia of Database Systems*, pp. 1613-1619.

Giunchiglia, F.; Soergel, D.; Maltese, V.; Bertacco, A. (2009). Mapping large-scale Knowledge Organization Systems. *Proceedings of the 2nd International Conference for Digital Libraries and the Semantic Web (ICSD)*, pp. 105-116.

Giunchiglia, F.; Zaihrayeu, I.; Farazi, F. (2009). Converting classifications into owl ontologies. *Presented at the Artificial Intelligence and Simulation of Behaviour Convention Workshop on Matching and Meanin*g.

Giunchiglia, F.; Dutta, B.; Maltese, V. (2009). Faceted Lightweight Ontologies. In: *Conceptual Modeling: Foundations and Applications*, A. Borgida, V. Chaudhri, P. Giorgini, Eric Yu (Eds.) LNCS 5600 Springer.

Giunchiglia, F.; Maltese, V.; Farazi, F.; Dutta, B. (2010). GeoWordNet: a resource for geo-spatial applications. *Proceedings of ESWC*, vol. 6088, pp. 121-136.

Horrocks, I.; Sattler, U. (1999). A description logic with transitive and inverse roles and role hierarchies. *Journal of Logic and Computation*, 9(3), pp. 385–410.

Maltese, V.; Giunchiglia, F.; Autayeu, A. (2010). Save up to 99% of your time in mapping validation. *Proceedings of ODBASE*, vol. 6427, pp. 1044-1060.

Miller, G. (1998). WordNet: An electronic Lexical Database. *MIT Press*.

Miles, A.: Bechhofer, S. (2009). SKOS Reference. *W3C Recommendation*,

Nicholson, D.; Dawson, A.; Shiri, A. (2006). HILT: A pilot terminology mapping service with a DDC spine. *Cataloging & Classification Quarterly*, 42 (3/4), pp. 187-200.

O'Neill, E.; Chan, L. (2003). FAST (Faceted Application for Subject Technology): A Simplified LCSH-based Vocabulary. *Proceedings of IFLA meeting*, pp. 1-9.

Ranganathan, S. R. (1967). Prolegomena to library classification. *Asia Publishing House*.

Varzi, A. C. 2006. A note on the transitivity of parthood. *Applied Ontology*, vol. 1, pp. 141-146.

Vizine-Goetz, D.; Hickey, C.; Houghton, A.; Thompson, R. (2004). Vocabulary Mapping for Terminology Services. *Journal of Digital Information*, 4(4), Article 272.

Whitehead, C. (1990). Mapping LCSH into Thesauri: the AAT Model. *In Beyond the Book: Extending MARC for Subject Access*, pp. 81.

About authors

Vincenzo Maltese received his degree in Computer Science in 2000 at the University of Salerno (Italy). After five years of experience as programmer and teacher, he joined the University of Trento (Italy) for an internship in 2007. He is currently a fourth year PhD student at the local doctorate ICT School. His main research field is Knowledge Representation and Management. He participated in several projects including Interconcept (matching large scale Knowledge Organization Systems), Live Memories (active digital memories of collective lives), Semantic Geo-Catalogue (extending geo-catalogues with semantic capabilities), and the Living Knowledge EU FET project (dealing with diversity in knowledge) as WP leader and Interdisciplinarity & Communication Manager. He is co-author of the open source tools S-Match and GeoWordNet.
He is a member of several program committees including inter-alia ESWC, ODBASE, Ontology Matching Workshop, IEEE AINA and editorial journals such as IJCIS, the Journal of Zhejiang University and DKE.

Feroz Farazi is a postdoctoral research fellow at the University of Trento (UNITN) in Italy. He obtained a PhD degree in Computer Science from UNITN in 2010. His research interests include Knowledge Management, Ontologies and the Semantic Web. He has authored several publications including a book entitled “Faceted Lightweight Ontologies: A Formalization and Some Experiments”. He has collaborated in the development of GeoWordNet knowledge base at UNITN from 2009 to 2010. He has also collaborated in the Semantic Geo-Catalogue project funded by the Autonomous Province of Trento. He served as an assistant professor from 2003 to 2006 in the department of Computer Science and Engineering (CSE) at the University of Chittagong in Bangladesh. He also served as a lecturer in the department of CSE at Shahjalal University of Science and Technology, Bangladesh from 2001 to 2003.