

The Foundational Ontology Library ROMULUS

Zubeida Casmod Khan and C. Maria Keet

School of Mathematics, Statistics, and Computer Science,
University of KwaZulu-Natal and UKZN/CSIR-Meraka Centre for Artificial
Intelligence Research, South Africa
zkhan@csir.co.za, keet@ukzn.ac.za

Abstract. A purpose of a foundational ontology is to solve interoperability issues among domain ontologies and they are used for ontology-driven conceptual data modelling. Multiple foundational ontologies have been developed in recent years, and most of them are available in several versions. This has re-introduced the interoperability problem, increased the need for a coordinated and structured comparison and elucidation of modelling decisions, and raised the requirement for software infrastructure to address this. We present here a basic step in that direction with the Repository of Ontologies for MULTiple USes, ROMULUS, which is the first online library of machine-processable, modularised, aligned, and logic-based merged foundational ontologies. In addition to the typical features of a model repository, it has a foundational ontology recommender covering features of six foundational ontologies, tailor-made modules for easier reuse, and a catalogue of interesting mappable and non-mappable elements among the BFO, GFO and DOLCE foundational ontologies.

1 Introduction

It has been 15 years since the introduction of the notion of ontology-driven information systems [1], which entails ontology-driven conceptual data modelling [2,3,4,5]. Ontology-driven conceptual data modelling uses principles and solutions from Ontology (philosophy) and ontologies (as artifacts) to improve the quality of a conceptual data model and refine its language, which therewith improves the quality of the information system. This can be applied to individual aspects by devising solutions to modelling problems in conceptual data models—e.g., part whole relations in conceptual models, aided by a foundational ontology [4]—as well as more broadly to modify a conceptual data modelling language’s metamodel thanks to a foundational ontology (e.g., [3]). However, one solution or extension may use, say, the DOLCE foundational ontology [6] for refining UML’s aggregation association [4], another could be informed by the UFO foundational ontology [3], and yet another by GFO [7], but it is not clear whether DOLCE, UFO, and GFO are compatible. Proliferation of modelling improvements, then, end up to be incompatible if the improvements rest on different philosophical assumptions represented in different foundational ontologies. Moreover, besides DOLCE, GFO, and UFO, other foundational ontologies have been developed over the years, such as SUMO [8], YAMATO [9], and BFO

[<http://www.ifomis.org/bfo>]. This potential for incompatibilities for ontologically well-founded conceptual data models has been recognised in the field of ontologies where they serve integration of domain ontologies, and it has been observed already within the Semantic Web setting, where ontology developers use their preferred foundational ontology that differ in various aspects, which therewith exhibit a semantic interoperability problem for domain ontologies.

A solution to such semantic issues has been proposed as the “WonderWeb Foundational Ontologies Library” (WFOL) in 2003, so that one should be able to commit to different but systematically related (modules of) foundational ontologies [6]. However, this library has not been implemented due to theoretical and implementation limitations. The main theoretical hurdle is alignment of foundational ontologies. Implementation limitations were primarily due to the absence of a common representation language, and there was scant stable software infrastructure for ontologies. Thanks to a range of advances in the meantime, the solvability of the implementation issues is within reach but has not yet been realised, whilst foundational ontology alignment and mapping is at the early stages. The creation of a software-based model repository is a necessary first step to manage these issues so as to have a one-stop shop for foundational ontologies and therewith foster coordinated, or at least interchangeable, ontology-driven conceptual data modelling with broadly usable results, as well as enabling examination of interchangeability of a foundational ontology that is mapped to a domain ontology.

We propose to solve practical shortcomings through the creation of the first such online library of machine-processable, aligned and merged, foundational ontologies: the Repository of Ontologies for MULTiple USEs ROMULUS. ROMULUS has features typical of Open Ontology Repositories [10], such as browsing the ontology and standardized metadata. Moreover, it incorporates a new web-based version of ONSET [11] that helps selecting a foundational ontology, it contains the included foundational ontologies’ OWLized version in whole and as various types of modules, and it contains both the logic-based pairwise alignments of DOLCE, BFO, and GFO, as well as a catalogue of individual alignments that are not mappable due to other axioms. ROMULUS is online accessible at <http://www.thezfiles.co.za/ROMULUS/>.

In the remainder of the paper, we first describe several motivating examples for conceptual data models in Section 2. The design and features of ROMULUS are presented in Section 3, and compared with related works on model repositories with respect to ontologies in Section 4. We discuss ROMULUS and the relevance of some mappings for ontology-driven conceptual data modelling in Section 5 and we conclude in Section 6.

2 A Selection of Motivating Examples

In the following motivating examples, the purpose is not to discuss which way is better for modelling a universe of discourse or refining a modelling language, but instead to demonstrate *i)* consequences of a modelling choice, *ii)* the need

for systematically related elements of foundational ontologies, and *iii*) that a foundational ontology library such as ROMULUS does assist with this.

Let us take as first example UML classes and EER or ORM entity types on the one hand, and a foundational ontology’s counterpart to that, which may be subsumed by **Particular** for instances or **Universal** for kinds or ‘classes’ of particulars. For instance, DOLCE is a foundational ontology of particulars that has entities (categories), such as **Endurant** (an entity wholly present at a time), **Process**, and **Amount of Matter** (stuff, like water and gold) [6]. BFO is an ontology of universals, which has, **Independent Continuant** (alike **Endurant**), **Process**, and **Fiat Object Part**. UFO [3] and GFO [7] are ontologies of particulars *and* universals: one of the first distinctions in the hierarchy is between **Particular/Individual** and **Universal**, where the former subsumes, among others, **gfo:Presential**, **gfo:Role**, **ufo:Set**, and **ufo:Quality**, and the latter **gfo:Persistent** and **ufo:RigidSortal**. Does it matter for conceptual data modelling? It can, and, moreover, does in the case where one wants to align ontological principles of a conceptual data modelling language to a foundational ontology. For instance, UFO has **Quality** (the ontological version of an attribute) as a subclass of **Moment**, and a **Quality Universal** as subclass of **Moment Universal** [3]. GFO’s **Property** is similar in idea, but is a subclass of **Individual**, and is unrelated to moments. That is, similar ideas are actually quite distinct. A different issue is that of GFO’s **Amount_of_substrate** (like water, gold) and DOLCE’s **Amount of Matter**, which convey a similar notion as Guizzardi’s “stuff universal” for which was proposed a stereotype $\ll\text{quantity}\gg$ that is a sortal that is a universal in [12], but UFO—used in another extension [3]—does not consider amounts of matter, and nor does BFO, i.e., stuff does not exist according to UFO and BFO, so we cannot identify and model it. This results in a situation where aligning EER or UML to DOLCE or GFO would permit us to create a stereotype to denote such entities and relate them with **subQuantityOf** [4,12], but not with UFO or BFO.

These examples may seem confusing to a reader, for one does not have an overview at hand of each of the foundational ontologies’ content and structure. And, perhaps, for some alignments, it does not make a difference which foundational ontology is chosen because not all its entities are used in such an alignment. However, we only can know this for sure if we have insight in the systematically assessed mappings between the entities in the various foundational ontologies. For instance, DOLCE’s and GFO’s notion of amounts of matter are comparable also when taking into account the structure of the ontology, but seemingly similar entities can end up not to be so: mapping **dolce:set** to **gfo:Set** results in an inconsistency, because DOLCE has it as a type of **Abstract** that is a **Particular** (equivalent to **gfo:Individual**), whereas **gfo:Set** is disjoint from **gfo:Individual**. UFO’s **Set** is alike DOLCE’s **set**—an **AbstractParticular**—so, by transitivity, one can infer that **ufo:Set** and **gfo:Set** are ontologically distinct entities as well.

In addition to the need for a systematic investigation into possible alignments and mappings to foster the possibility that the various extensions to conceptual data modelling languages are practically compatible, we need ways to at least quickly check that, and, where relevant for the application scenario, swap

between one or the other foundational ontology where possible. Although this can be carried out and maintained on paper, software-based mapped ontologies where the mappings are at least guaranteed not to lead to any inconsistency simplify this process. This, then, could feed into a software-based content negotiation method. Likewise, a catalogue of entities that cannot be mapped serves as an easy online reference of incompatibilities.

3 A Repository of Ontologies for MULTIPLE USES

In line with our goal of creating a foundational ontology repository, we have designed and implemented a web-based software system ROMULUS in order to allow modellers to publicly access and benefit from all the functionality of the repository. In this section, we describe the requirements, design, and features of the Repository of Ontologies for MULTIPLE USES.

Before the actual design, requirements were formulated. The functional requirements are briefly described here, of which the first three are adapted from the original WFOLE proposal in [6]. The library must provide a high-level view of the foundational ontologies with only the most general entities common to all implemented foundational ontologies, it must provide a comparison of implemented foundational ontologies, and ontology metadata must be available [6]. In addition, to serve interoperability and interchangeability, basic ontology mediation must be present, including alignment, mapping, and merging of the foundational ontologies. To facilitate usability, the foundational ontologies in the repository must be modularised, there must be easy and effective online ontology browsing and searching, renderings in human-readable views of each foundational ontology module must be available, and there must be an ontology download facility.

3.1 Design of ROMULUS

The three main components of ROMULUS are the web server (used to execute HTML pages and PHP scripts), the Tomcat server (used to execute JSP pages) and the MySQL database (used to store ontology alignments, ontology metadata and users' ontology selection results and to assist with search functions). The interaction of the components in ROMULUS is shown in Fig. 1, and we describe here the four conceptually different aspects of the design: the front-end features, ontology metadata, the ontologies themselves that have been modularised to meet and anticipate further requests for partial foundational ontologies suitable for a task, and ontology selection.

Front-End Features. The modular design of the foundational ontology library is met through different tabs in the user interface of the repository. For online ontology browsing, WebProtégé [13] is used, which requires a tomcat server. There are separate HTML pages with tables and lists for the comparison of foundational ontologies for the different categories of criteria¹, for the

¹ There is a page for each category: ontological commitments, representation languages, software engineering properties, subject domains and applications.

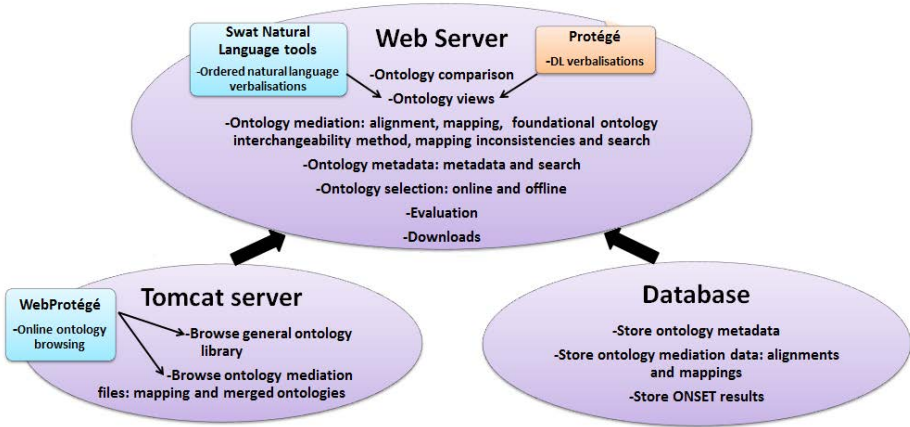


Fig. 1. The interaction of ROMULUS's components

user-readable version of the alignments of the ontologies and, for the metadata. SWAT Natural Language tools [14] was used to generate the HTML pages of the verbalisation of each ontology module in a structured natural language format. The Protégé-generated Description Logic axioms of each ontology module are available as pdf files. ROMULUS's alignments and mappings are stored in a database and rendered as HTML tables. Similar to the ontology browsing page, WebProtégé is used to easily access all the mappings and the merged ontologies. The foundational ONtology Selection and Explanation Tool ONSET [11] has its own tab where it may be executed online and downloaded for offline usage. The experimental foundational ontology interchangeability method listed in Fig. 1 can be accessed at ROMULUS's foundational ontology interchangeability page.

Ontology Metadata. Additional data pertaining to foundational ontologies are required to assist ontology developers with reusing an ontology effectively. Metadata values for each original, modularised, mapped and merged ontology module are provided in ROMULUS. In order to facilitate interoperability with other ontology repositories, we considered the Ontology Metadata Vocabulary (OMV) [15], which is a general OWL-formalised metadata vocabulary, and OM²R metadata model [16], which is aimed at ontology mapping reuse. ROMULUS uses OMV and OM²R, as well as its own metadata for each foundational ontology. The additional metadata not available in OMV and OM²R concerns modularity with module type (e.g., OWL 2 profile; see below) and original ontology, and mediation with original ontologies that are mediated and alignment type. For this first version of ROMULUS, we store the metadata in ROMULUS's centralised back-end database together with the alignments and mappings (Fig. 2), which we render to human-readable pages describing the metadata for each ontology. Storing the metadata in a database makes it easier to search through it, and we hope to minimise duplication. We intend to integrate the OWL version of the metadata in the ontologies in a future version.

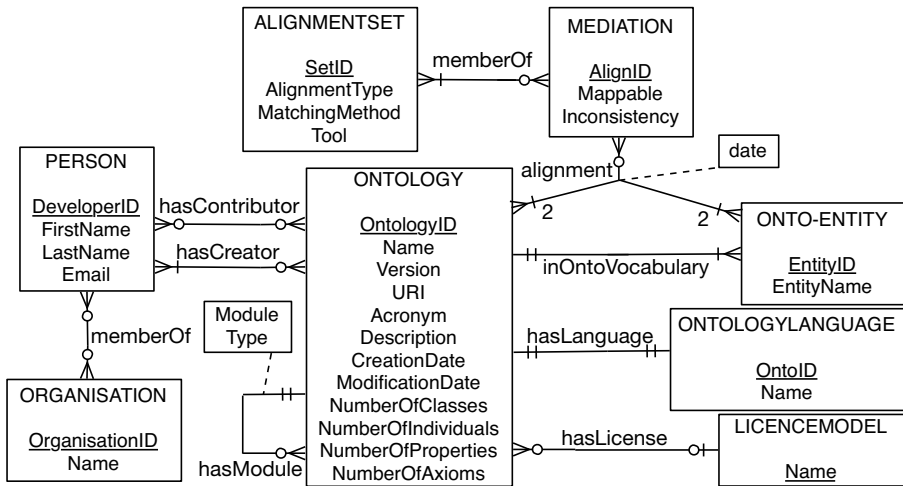


Fig. 2. ER diagram of ROMULUS's database regarding ontology mediation, extending (a subset of) OMV and OM²R on, notably, mediation, entity, and module

Modularised Ontologies. Ontology modularisation deals with creating or altering an ontology to be broken down into modules for specific functions. The idea behind it is to hide unnecessary detail when not required. Modularity is important in that it aids in ontology maintenance, publication, validation and processing. Factors pertaining to modularization are discussed elsewhere [17,18], and modules are created to organise and manage domain coverage, isolate branches of a taxonomy, extract a particular subject domain and/or theory, to isolate patterns, assist with scalable automated reasoning, or to reduce the cognitive overload. Such modularization ideas have been incorporated in ROMULUS on an experimental basis.

The following types of modules have been created:

- *Separate branches of 3D and 4D entities in the ontologies:* For when one wants to keep the entities that exist as a whole at all times (3D entities) separate from entities with temporal parts that unfold over time (4D entities);
- *Isolated branches of taxonomies of the ontologies for available subject domains support:* A module can be used for a specific subject domain; e.g., biomedical, business.
- *More/less detailed versions of the ontologies:* For when one does not need the entire functionality of the ontology, variants of an ontology with fewer/more entities, properties, and axioms; e.g., gfo-basic and gfo-full.
- *OWL 2 profiles:* For when one wants to improve the efficiency of reasoning, modules in different expressive fragments of OWL 2; e.g., OWL 2 EL.

OWL Module extractor [19], Swoop [20] and Protégé have been considered for ontology modularisation of the foundational ontologies. OWL Module extractor and Swoop use a logic based analysis of the axioms only and this resulted in large modules similar to the original ontologies. For this reason, we could not apply

OWL Module extractor and Swoop to create modules. In DOLCE, *endurant* and *perdurant* are linked by a participation relation, making it difficult to separate them into separate hierarchies. In order to create modules of these types, it was necessary to manually remove some of the axioms relating the two entities. We encountered a similar problem when modularising DOLCE to be a module without *quality* and *qualia*. Protégé generated smaller modules according to the user's input, in most cases, but some unnecessary entities were still present after using Protégé and they were manually modularised.

In addition to the 'full' versions of the three ontologies, the following ones were newly created:

- New DOLCE modules: DOLCEEndurants, DOLCE-Perdurants, DOLCENoQualityAndQualia, DOLCE-EL (trimmed to what can be represented in OWL 2 EL and DOLCE-QL);
- New BFO modules: BFOContinuants, BFOOccurrents, BFO-EL-QL-RL;
- New GFO modules: GFOATO (based on the Abstract Top Level layer), GFOACO (based on the Abstract Core Level), GFONoOccurrents, GFONoPersistantsAndPresentials, GFOBasicEL, GFOBasicQL.

Given that these modules are proper fragments of the original ones that were aligned, mapped, and merged, their respective remaining mappings for the modules are also available in ROMULUS.

Foundational Ontology Selection. It is worthwhile to include an ontology selector tool in ROMULUS in order to bridge the process of foundational ontology selection with the features offered by the repository. ONSET is such a—and, to date, the only—foundational ontology selection and explanation tool that was designed to aid the user with the task of selecting an appropriate foundational ontology for domain ontology development [11]. Significant changes and improvements have been made to v2.0 of ONSET, with an aim to integrate it in ROMULUS. Instead of a stand-alone jar file, there is now a web-based version, which has access to ROMULUS's centralised database (Users still can download the offline version v1.2 of ONSET, without linkage to ROMULUS's features, available in ROMULUS's ontology selection page). This also provides for the new feature that users can save their ontology selection results both locally in a CSV file format, and to let ONSET store a copy of it in ROMULUS's database, which can be used for further analysis and investigation with regard to foundational ontology usage and selection. A condensed conceptual data model for this is displayed in Fig. 3 (the details of *Results* and *ConflictingReasons* are not shown due to space limitations). ONSET v2.0 also provides links to features in ROMULUS, such as its modules and metadata for a particular foundational ontology. Finally, we added the YAMATO and GIST foundational ontologies to ONSET v2.0, therewith providing the user with more possible foundational ontology choices (the other ones in v1.2 are BFO, DOLCE, GFO, and SUMO). Use cases of ONSET can be found at ROMULUS's ontology selection page.

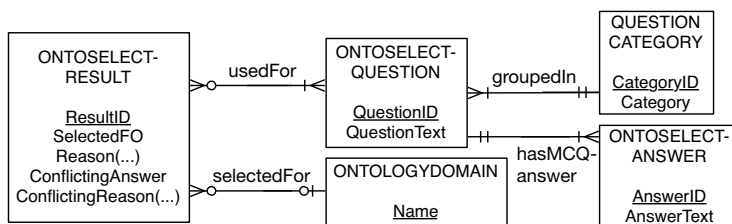


Fig. 3. Summarised ER diagram of ONSET’s saved data

4 Related Works

We compare ROMULUS to several other ontology repositories, and subsequently discuss some considerations for a foundational ontology library. A comparison was conducted with other repositories, including OOR [10], BioPortal [21], TONES [22], COLORE [23] and Ontohub [24], which is summarised in Table 1.

In terms of repository vision, ROMULUS is a repository of foundational ontologies. It is closed in that users cannot upload their own ontologies or data, but they are encouraged to download the ontologies and data on the repository. BioPortal, OOR and Ontohub are an open repositories where users are encouraged to upload their ontology projects, contributions, and download resources. BioPortal is specifically a repository of biomedical ontologies. TONES is aimed at being a central location for ontologies that will be helpful for application developers for testing purposes. It is a closed repository where users are only allowed to download the ontologies and view some metadata. COLORE aims to be an open repository of Common Logic ontologies to aid in ontology evaluation and integration techniques, and to support the design, evaluation, and application of ontologies in first-order logic.

From the comparison of functionality, ROMULUS provides advanced functionality in most of the criteria used in this evaluation. It also provides features that were not available in other repositories, which therefore merited the development of a new repository. These include complete metadata for each ontology that also includes metadata about modularity and ontology mediation beyond the standard metadata vocabularies, carefully analysed alignments and merged ontologies, and a foundational ontology selection and explanation tool for guidance to select the most relevant one.

5 Discussion

ROMULUS combines various technologies to provide a range of features and it is the first realisation of the “WFOL” envisioned since 2003. It meets the main goals of the WFOL [6], described in Section 3: 1) it provides a higher-level ontology, FFO, containing only the common entities of DOLCE, BFO and GFO ontologies, which could be used as a starting point for ontology development;

Table 1. Comparison of ROMULUS's features with those of other repositories

	ROMULUS	BioPortal and OOR	TONES	COLORE	Ontohub
Browse	Uses WebProtégé. Hierarchical ontology view. Displays classes and class local neighborhoods, individuals and annotations of an ontology.	Hierarchical ontology classes and class local neighborhood. Has advanced visualization support.	Browsing is currently unavailable.	No support.	Sequential ontology view. Lists of classes, properties, individuals and sentences.
Mediation	Alignments and mappings between its ontologies, merged and higher-level foundational ontologies, mapping inconsistencies and a method for foundational ontology interchangeability.	Some mappings between ontologies. Users may specify mappings.	No support.	No support.	No support.
Metadata	Metadata lists for ontologies based on OMV, OM ² -R, and own model.	Metadata lists for ontologies based on OMV, but some missing metadata.	Metadata for metrics of each ontology.	Metadata exists in the ontology files.	Metadata lists for ontologies, but some missing metadata.
Ontology selection	ONSET, a foundational ontology selection tool, available to download and execute locally.	An ontology recommender system that allows one to calculate which ontologies are most relevant for a corpus.	No support.	No support.	No support.
Search	Allows a user to search for alignments, mappings and metadata.	Advanced search. Allows a user to search for entity names, ids, synonyms, properties and filter search.	No support.	No support.	Search for ontology name and ontology symbol.
Ontology view	Views in description logics and natural language.	No support.	No support.	Views in common logic.	No support.
Comparison	Ontology comparison in terms of different categories of criteria.	No support.	No support.	No support.	No support.
Ontology access	Users can view and download ontologies and tools.	Users can upload, edit own ontologies and download ontologies.	Users can download ontologies.	Users can view and download ontologies.	Users can upload, edit own ontologies and download ontologies.

2) it is a reference point for comparisons between different ontological approaches of selected foundational ontologies; 3) it provides a common framework for analyzing, harmonizing and integrating existing ontologies and metadata standards thanks to its criteria comparison of selected foundational ontologies, alignments, mapping and merged ontologies, and extensive metadata for each ontology. Also, it is rigorous, including a logic-based approach, and extensively researched. In addition to these WFOL requirements, ROMULUS provides modular and mediated foundational ontologies, online browsing of the ontologies, online and offline foundational ontology selection and download facilities.

The results of the ontology mediation are partially based on a preliminary ontological analysis and partially based on the formalizations of the respective foundational ontologies. Nevertheless, there were still unexpected inconsistencies of entities and of relationships that one would consider well-established ontologically, such as a mathematical set. We do not aim to solve such inconsistencies here. Instead, ROMULUS with its ontology mediation results is meant as a systematic foundation for such an investigation, and it provides starting points for deeper ontological analyses to possibly resolve them. To aid such investigations, additional features could be added to ROMULUS, such as a wiki-like discussion page for each ‘alignment with issues’.

From an ontology engineering viewpoint, ROMULUS is a major step toward foundational ontology interchangeability, because a prerequisite for this are the mapped ontologies. Meaning negotiation between two domain ontologies that each are linked to a different foundational ontology through a merged ontology of those two foundational ontologies has now become something within reach. Although the technologies might seem ‘pedestrian’ now, they were not until recently, and it is principally the realisation of OOR features extended for foundational ontologies that makes ROMULUS a novelty.

Revisiting FO Interoperability for Conceptual Data Models. Instead of cumbersome manual analyses as described in Section 2, we now can conduct a quick look-up in ROMULUS. For instance, *Set* is in the list of logical inconsistencies, so we do not have to look further. This is true for parthood as well, or: whichever foundational ontology one chooses, it is never truly the same with any of the others. This information is available now at one’s fingertips, compared to reading through the foundational ontology’s respective documentation: UML’s aggregation association is typically mapped into a part-whole relation, which can be parthood in the sense of mereology or meronymy [4]. Considering for the sake of example only the mereology usage of the aggregation association, we would have to specify which mereological theory it would fit in with. DOLCE uses the AGEM theory where proper parthood is irreflexive, asymmetric and transitive, BFO’s Relation Ontology [25] has only transitivity as characteristic of proper parthood, and the higher-order logic version of BFO itself is founded on a constellation of parthood theories. Then, if we were to refine UML’s metamodel, it does make a difference which foundational ontology we use for the exercise, because the (logical) implications will be different.

Conversely, DOLCE's Amount of Matter does not appear in the list of inconsistencies, but instead is in the mappings with GFO. If one is willing to be slightly lenient on the particular versus universal issue, then the foundational ontologies' version of attributes can be matched among all three foundational ontologies, using the equivalences among Quality and Property. This is now readily available with the pairwise mapped, online, ontologies and searchable mappings and inconsistencies.

6 Conclusions

We presented a core step in the direction of addressing interoperability issues with the Repository of Ontologies for MULTiple USEs, ROMULUS, software infrastructure. This is the first online library of machine-processable, modularised, aligned, and logic-based merged foundational ontologies. In addition to the typical features of a model repository, ROMULUS has a foundational ontology recommender covering features of six foundational ontologies that is integrated with ROMULUS' features and it has tailor-made modules for easier reuse. Most important for the actual ontology-driven conceptual data modelling, are its features and site content with a catalogue of interesting mappable and non-mappable elements among the BFO, GFO and DOLCE foundational ontologies, and the pairwise machine-processable mapped ontologies.

We are currently adding extended search features, and the preliminary user evaluation of the alignments (available in ROMULUS already), will be extended with community discussion pages. Also, we hope to gather sufficient voluntarily saved ontology selections to analyse them and find patterns in selection criteria.

Acknowledgements. This work is based upon research supported by the National Research Foundation of South Africa (Project UID: 80584) and the Argentinian Ministry of Science and Technology.

References

1. Guarino, N.: Formal ontology and information systems. In: Proc. of FOIS 1998. IOS Press (1998)
2. Guarino, N., Guizzardi, G.: In the defense of ontological foundations for conceptual modeling. *Scandinavian Journal of Information Systems* 18(1), 9 (2006)
3. Guizzardi, G., Wagner, G.: Using the unified foundational ontology (UFO) as a foundation for general conceptual modeling languages. In: *Theory and Applications of Ontology: Computer Applications*, pp. 175–196. Springer (2010)
4. Keet, C.M., Artale, A.: Representing and reasoning over a taxonomy of part-whole relations. *Applied Ontology* 3(1-2), 91–110 (2008)
5. Keet, C.M.: Ontology-driven formal conceptual data modeling for biological data analysis. In: *Biological Knowledge Discovery Handbook: Preprocessing, Mining and Postprocessing of Biological Data*. Wiley (in press, 2013)
6. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: Ontology library. WonderWeb Deliverable D18 (ver. 1.0, 31-12-2003) (2003), <http://wonderweb.semanticweb.org>
7. Herre, H.: General Formal Ontology (GFO): A foundational ontology for conceptual modelling. In: *Theory and Applications of Ontology: Computer Applications*, pp. 297–345. Springer (2010)

8. Niles, I., Pease, A.: Towards a standard upper ontology. In: Proc. of FOIS 2001, Ogunquit, Maine, October 17-19, IOS Press (2001)
9. Mizoguchi, R.: YAMATO: yet another more advanced top-level ontology. In: Proc. of AOW 2010. CRPIT, pp. 1–16. ACS, Sydney (2010)
10. Baclawski, K., Schneider, T.: The open ontology repository initiative: Requirements and research challenges. In: Proceedings of the Workshop on Collaborative Construction, Management and Linking of Structured Knowledge, Washington DC, USA, October 25. CEUR-WS, vol. 514 (2009)
11. Khan, Z., Keet, C.M.: ONSET: Automated foundational ontology selection and explanation. In: ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., d’Acquin, M., Nikolov, A., Aussenac-Gilles, N., Hernandez, N. (eds.) EKAW 2012. LNCS, vol. 7603, pp. 237–251. Springer, Heidelberg (2012)
12. Guizzardi, G.: On the representation of quantities and their parts in conceptual modeling. In: Proc. of FOIS 2010, Toronto, May 11-14. IOS Press (2010)
13. Tudorache, T., Vendetti, J., Noy, N.F.: Web-Protege: A lightweight OWL ontology editor for the web. In: Proc. of OWLED 2008. CEUR-WS, vol. 432 (2008)
14. Third, A., Williams, S., Power, R.: OWL to English: A tool for generating organised easily-navigated hypertexts from ontologies. In: Proc. of ISWC 2011(Poster/Demo), Bonn, Germany, October 23-27. Springer (2011)
15. Hartmann, J., Sure, Y., Haase, P., Palma, R., del Carmen Suárez-Figueroa, M.: OMV - Ontology Metadata Vocabulary. In: Ontology Patterns for the Semantic Web (OPSW), Galway, Ireland (November 2005)
16. Thomas, H., Brennan, R., O’Sullivan, D.: Using the OM2R meta-data model for ontology mapping reuse for the ontology alignment challenge - A case study. In: 7th Intl. Ws. on Ontology Matching (OM 2012). CEUR-WS, vol. 946 (2012)
17. d’Aquin, M., Schlicht, A., Stuckenschmidt, H., Sabou, M.: Criteria and evaluation for ontology modularization techniques. In: Stuckenschmidt, H., Parent, C., Spaccapietra, S. (eds.) Modular Ontologies. LNCS, vol. 5445, pp. 67–89. Springer, Heidelberg (2009)
18. Borgo, S.: Goals of modularity: A voice from the foundational viewpoint. In: Proc. of WoMO 2011. FAIA, vol. 230, pp. 1–6. IOS Press (2011)
19. Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Modular reuse of ontologies: Theory and practice. *J. of Artificial Intelligence Research* 31, 273–318 (2008)
20. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C., Hendler, J.A.: Swoop: A web ontology editing browser. *Journal of Web Semantics* 4(2), 144–153 (2006)
21. Whetzel, P.L., et al.: Biportal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications. *Nucleic Acids Research* 39(Web-Server-Issue), 541–545 (2011)
22. Vale, D.C., et al.: The TONES ontology repository, <http://owl.cs.manchester.ac.uk/repository/browser> (accessed on December 22, 2012)
23. Gruninger, M.: COLORE: Common logic ontology repository (2009), <http://ontolog.cim3.net/file/work/00R-Ontolog-Panel/2009-08-06Ontology-%?Repository-Research-Issues/Colore--MichaelGruninger20090806.pdf>
24. Lange, C., Mossakowski, T., Kutz, O., Galinski, C., Gruninger, M., Couto Vale, D.: The distributed ontology language (DOL): Use cases, syntax, and extensibility. In: Proc. of Terminology and Knowledge Engineering, TKE 2012 (2012)
25. Smith, B., et al.: Relations in biomedical ontologies. *Genome Biol.* 6(5), 46 (2005)