Erlangen Implementation of FRBRoo

Judith Merges, Martin Scholz, Guenther Goerz

Professorship of Artificial Intelligence, University of Erlangen-Nuremberg
Haberstr. 2, 91058 Erlangen, Germany
Judith.Merges@ikgf.uni-erlangen.de

Short Paper, CIDOC June 2012

Abstract: This paper reports on on-going research implementing the FRBRoo (Object-oriented Definition and Mapping to Functional Requirements for Bibliographic Records) in the description logic language OWL-DL. This implementation is called Erlangen FRBRoo (EFRBRoo). The goal to be fully compatible to Erlangen CRM / OWL (ECRM) is achieved by applying the same implementation patterns of ECRM wherever possible. In order to preserve the FRBRoo specifications at the same time EFRBRoo keeps also as close as possible to the text of the FRBRoo definition. The EFRBRoo is made available online.
To evaluate and improve the quality of the EFRBRoo implementation, the software development method of static unit testing is applied: simplified test cases have been designed, which consist of representative datasets in OWL-DL, SPARQL queries and respective results. In addition, validators (RDF, OWL) and frameworks (Protégé, Sesame, and WissKI) have been used for validating formal correctness and class consistency.
In order to extend the evaluation to real world applications, the data of the bibliographic SQL database of the International Consortium for Research in the Humanities (IKGF) has been converted to OWL-DL. Research at the IKGF focuses on differences in prognostication and strategies to cope with destiny in China and medieval Europe. The impact of these historical differences on the immediate present and our respective way of coping with the future is analysed. The bibliographic data contain many different languages such as German, English, Chinese, Latin and Sanskrit. With more than 1400 publications the bibliography of the IKGF provides an ideal extent to sample real world applicability.

1 Background

Models help to communicate, to explain and to make predictions. They are also used to mediate among multiple viewpoints. This is accomplished by providing shared terms with well-defined semantics and by forming an abstract description that hides certain details to high-light others. To fulfil these tasks it is also advantageous for a model to be known and interpreted by many people. For example: to people who know the molecular formula - a model that expresses information about the atoms that constitute a particular chemical compound - the expression "H_2O" is known as "water". It is vital that the model reflects the appropriate expressivity, so that it is able to express everything needed, but omits everything that is not needed, to be able to focus.
In the area of cultural heritage and libraries there are two major models, which have undergone significant development in the past years and are subject to today's research.

First, in the field of the cultural heritage profession, there is the CIDOC Conceptual Reference Model (see [CRM2011]), referred to as CRM. The CRM is an object-oriented formal ontology intended to facilitate the integration, mediation and interchange of heterogeneous cultural heritage knowledge. Starting in 1996 it has been developed and maintained under the auspices of the ICOM-CIDOC Documentation Standards Working Group. The CRM is available online (see [CRM2011]) as PDF Document and its latest version 5.0.4 was issued in November 2011 and contains about 90 entities and 140 properties. In 2006 the CIDOC CRM version 3.4.9 has been certified as ISO standard 21127:2006. An implementation using the semantic markup language OWL (Web Ontology Language) for sharing ontologies on the World Wide Web has been done in Erlangen. This Erlangen CRM / OWL-DL implementation (see [ECRM2012]), referred to as ECRM, attempts to be as close as possible to the text of the CRM model specification. It is available online (see [ECRM2012]).

Second, in the area of library and information profession, there is the International Federation of Library Associations (IFLA) Functional Requirements for Bibliographic Records (see [FRBR2009]), referred to as FRBR. The FRBR model represents an attempt to establish a logical framework assisting in the understanding and further development of conventions for bibliographic descriptions. It is a conceptual model, which is based on the entity-attribute-relationship model of analysis. First published in 1998 it has been developed and maintained under the auspices of the IFLA Study Group on Functional Requirements for Bibliographic Records. The FRBR is available online as PDF document (see [FRBR2009]) and the latest version was issued in February 2009. As theoretical conceptual model it is not meant to be implemented per se, but it defines in a systematic way the expectations of the user in finding and using information within bibliographic records. It shall create a particular understanding that can be used to design and build online catalogue systems.

In the year 2000 the idea emerged of bringing CRM and FRBR together to create a shared perspective of accessing and using knowledge within both domains of cultural heritage and libraries. In the following process both models have been enriched. An object-oriented perspective of the FRBR, referred to as FRBRoo (see [FRBRoo2012]), extending the formal CRM ontology has been created. The first FRBRoo version 1.0.1 was made available in October 2011, and the latest version 1.0.2 from February 2012 is available online (see [FRBRoo2012]) as PDF document.

2 Implementing FRBRoo in OWL-DL / XML

The Erlangen implementation of the textual FRBRoo specification, referred to as EFRBRoo, has been implemented in the same semantic markup language OWL-DL / XML as the Erlangen CRM implementation (ECRM). The EFRBRoo attempts to both ensure the seamless integration as ECRM extension as well as to be as close as possible to the text of the FRBRoo specification. To accomplish these goals the implementation patterns and naming conventions of ECRM have been applied to EFRBRoo. The EFRBRoo is available online[1].

---

[1] http://erlangen-crm.org/efrbroo

3 Debugging Ontologies

The development of semantic models is a challenging task. As ontologies become larger, the probability raises that they contain defects. Even if defective ontologies may still be useful, problems are prone to occur, especially if the ontology is used in semantic applications. Wrong conclusions can be drawn or valid conclusions missed.

Defects in ontologies are categorised the following way: (see [LQM2011])

1. Syntactic defects
2. Semantic defects
3. Modelling defects

Syntactic defects are usually easy to detect and to solve and automatic validation for the specific semantic modelling language can be used.
More severe are semantic defects, in particular incoherent and inconsistent ontologies. Usually the size of ontologies is beyond the human capacity to overview the complete ontology with all possible conclusions. Very useful in that regard is the web ontology language OWL that uses description logic as foundation and therefore allows the detection of inconsistencies in ontology models. However, if an inconsistency is found, further work needs to be done to provide an explanation of the origin of the inconsistency and to suggest a solution for resolving the inconsistency. No mature debugging framework is available, although research in that area has been conducted (see [Stu2008], [LL2011]).
Modelling defects are commonly structural defects like missing or wrong relationships. Usually domain knowledge is required to detect and resolve those defects.

4 Syntactic Evaluation

The Erlangen Implementation of FRBRoo (EFRBRoo) is a syntactically valid RDF, OWL-DL and OWL 2 DL ontology.
The RDF syntax validity check was performed with the RDF Validator[2] of W3C. The OWL-DL syntax validity check was performed with the WonderWeb OWL Ontology validator [3] of the University of Manchester and the University of Karlsruhe and the OWL 2 DL syntax validity check was performed with the OWL 2 Validator[4] of the University of Manchester.

5 Semantic Evaluation

In order to detect inconsistencies in EFRBRoo three sets of test cases were designed:

The first set of test cases contained a single OWL-DL ontology for any FRBRoo class (F1-F33, F40-F44) containing only one instance of that class. These ontologies import the EFRBR

---

[2] http://www.w3.org/RDF/Validator/
[3] http://www.mygrid.org.uk/OWL/Validator
[4] http://owl.cs.manchester.ac.uk/validator/

Version 120219 and consist of only two statements: first that an instance is an OWL Named Individual and second that this instance is of the type of the specific EFRBRoo class.

This first set of test cases was built to ensure that further test sets containing more than one instance of one class can rule-out inconsistencies in single instances.
All test cases were loaded in Protégé 3.4.1 using the Pellet 1.5.2 reasoner and in Protégé 4.1 using the reasoner FaCT++ and HermiT 1.3.6. In all cases the computation of inconsistent concepts returned no inconsistencies as result.

The second set of test cases contained a single OWL-DL ontology for any FRBRoo property (R1-R31, CLP2, CLP43, CLP45-46, CLP104-105, CLR6, R40 R53, and CLP58) containing only the usage of that property. These ontologies import the EFRBR Version 120219 and consist of three statements: first that an instance, named subject of the property, is an OWL Named Individual and second that an instance, named object of the property, is an OWL Named Individual and third the statement that connects the subject instance to the object instance using the EFRBRoo property.
All test cases were loaded in Protégé 3.4.1 using the Pellet 1.5.2 reasoner and in Protégé 4.1 using the reasoner FaCT++ and HermiT 1.3.6. The computation of inconsistent concepts in the Protégé 3 environment returned no inconsistencies as result, in contrast to Protégé 4 that computed inconsistencies in the test cases of R4, R11, R26 and R41.

The third set of test cases extended the second set by adding two statements: first that the instance, named subject of the property, is of the EFRBRoo class defined as property domain and second that the instance, named object of the property, is of the EFRBRoo class defined as property range.
All test cases were loaded in Protégé 3.4.1 using the Pellet 1.5.2 reasoner and in Protégé 4.1 using the reasoner FaCT++ and HermiT 1.3.6. The computation of inconsistent concepts returned in Protégé 3 and Protégé 4 the same inconsistencies as result in the test cases of R4, R11, R26 and R41.

To resolve the inconsistencies there are several possibilities that are illustrated on the basis of using R26 in the following example.

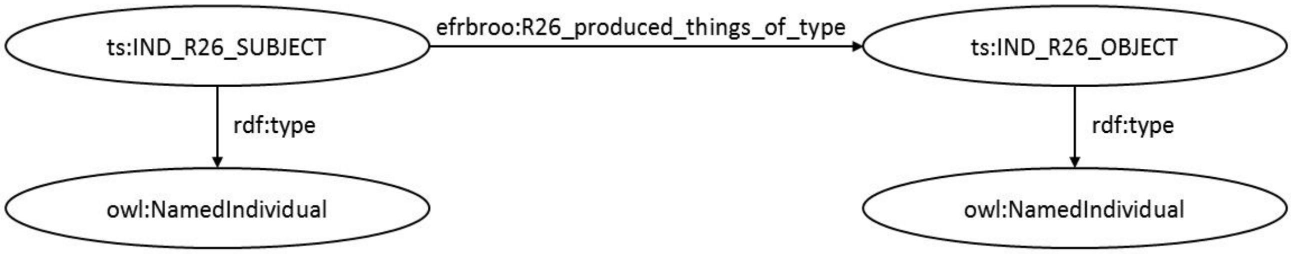

Figure 5-1[5]: Three Statements of EFRBRoo Test Case OWL Ontology of Property R26

[5] Namespace Abbreviation:

rdf:	<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
owl:	<http://www.w3.org/2002/07/owl#>
efrbroo:	<http://erlangen-crm.org/frbroo/120219/>
ts:	<http://testset.org/>

The statements contained in the test case of R26 are shown in Figure 5-1. The related class and property diagram of the EFRBR is visualised in Figure 5-2.

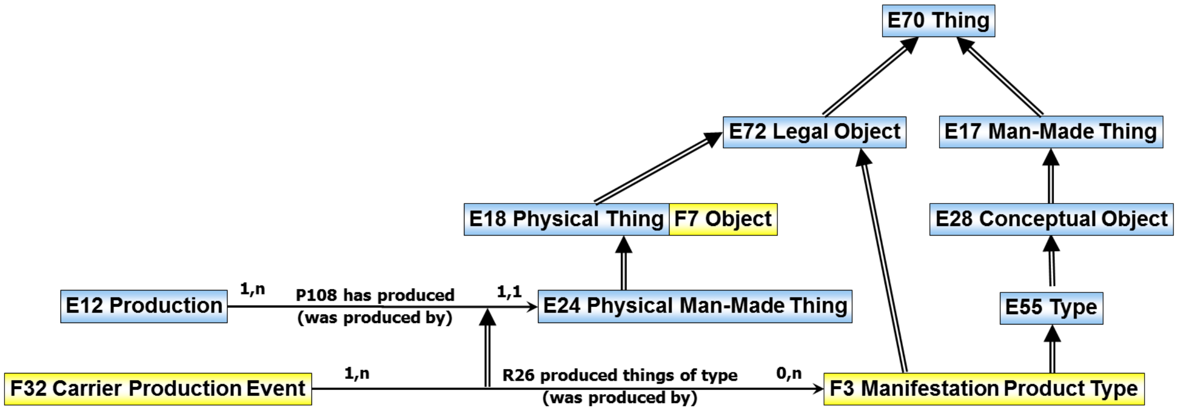

Figure 5-2: Extract of EFRBRoo Class and Property Diagram for Property R26 produced things of type

To understand the origin of the inconsistency it is descriptive to take a look to which classes the instance *IND_R26_OBJECT* belongs:

In EFRBRoo the class *F3 Manifestation Product Type* is defined as range of the property *R26 produced things of type*, therefore the instance *IND_R26_OBJECT* is of the type *F3 Manifestation Product Type*. Also the property *R26 produced things of type* is defined as subproperty of the ECRM property *P108 has produced* and because in ECRM the class *E24 Physical Man-Made Thing* is defined as range of the ECRM property *P108 has produced* the instance *IND_R26_OBJECT* is also of the type *E24 Physical Man-Made Thing*. Further the ECRM class *E24 Physical Man-Made Thing* is defined in ECRM as subclass of *E18 Physical Thing*, therefore the instance *IND_R26_OBJECT* is also of the type *E18 Physical Thing*. Next the EFRBRoo class *F3 Manifestation Product Type* is defined in EFRBRoo as subclass of the ECRM class *E55 Type*, which is defined in ECRM as subclass of *E28 Conceptual Object*, therefore the instance *IND_R26_OBJECT* is also of the type *E28 Conceptual Object*.

The inconsistency occurs, because ECRM defines that the classes *E18 Physical Thing* and *E28 Conceptual Object* are disjoint, but that contradicts the conclusions drawn by the three different reasoners shown above that the instance *IND_R26_OBJECT* is both *E18 Physical Thing* and *E28 Conceptual Object*.

Knowing the origin, the resolutions for the inconsistency suggest the following possibilities:

- changing the range of the EFRBRoo property R26
- changing the subclass relationship of the EFRBRoo property R26
- changing the statement that the ECRM classes E18 and E28 are disjoint

6 Model Evaluation

In order to take the first step towards finding modelling defects in EFRBRoo, the software development method of static unit testing was applied: simplified test cases, each including an OWL ontology based on EFRBRoo, representative SPARQL queries and text files with the expected results of those queries were implemented. The execution of the SPARQL queries were performed both in Sesame 2.6.1, an application programming interface in the programming language Java for processing RDF data including parsing, storing, inferencing and querying of such data, as well as in Jena 2.6.4, also a Java framework for building Semantic Web applications, which was approved in April 2012 as top-level Apache project.

For all test cases, first the test case ontology based on EFRBRoo including all drawn conclusions were loaded and then the SPARQL query was executed. The actual result of that query was stored as text file and compared with the former created text-file with the expected result.

The following example, in which the connection of the three classes E39 Actor, E21 Person and F10 Person was evaluated, shall give an exemplary overview about the usage of those test cases. Since the FRBRoo class F10 Person is specified as equal to the class E21 Person, an instance of F10 Person is expected to be also an instance of E21 Person. Further, since E21 Person is defined as subclass of E39 Actor, that instance is also expected to an instance of E39 Actor.
The test case ontology contained three instances of each of those classes and SPARQL queries asking for:

- all instances of type E21 Person
- all instances of type E39 Actor
- all classes that are equivalent to the class F10 Person
- all classes of which F10 Person is a subclass of
- all classes of which E21 Person is a subclass of

The test results were compared with the predicted results and the intended equivalence was shown.

7 Real World Application Evaluation

In order to extend the evaluation to real world applications, the entries of the names of all authors in the bibliographic SQL database of the International Consortium for Research in the Humanities[6] (IKGF) were converted to an OWL-DL ontology using EFRBRoo classes and properties. Research at the IKGF focuses on differences in prognostication and strategies to cope with destiny in China and medieval Europe. The impact of these historical differences on the immediate present and our respective way of coping with the future is analysed. The bibliography can be accessed online[7] and contains about 1400 publications of 1200 authors in many different languages such as German, English, Chinese, Latin and Sanskrit. This size of database is considered as a reasonable first step in testing the real world applicability of the implementation.

First an OWL-DL ontology containing only the names of all authors was implemented like shown in Figure 7-1 as incremental first step and an attempt to test for inconsistencies was performed.

[6] http://www.ikgf.uni-erlangen.de/
[7] http://www.ikgf.uni-erlangen.de/aigaion2/index.php/publications

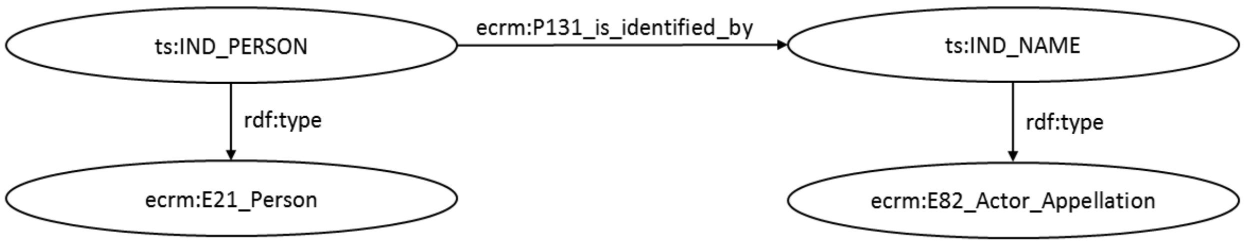

Figure 7-1: Mapping of *Name of Person* in FRBR Section 4.6.1 to FRBRoo (see [FRBRoo2012])

Already the generation of all inferred conclusions of this subset of the final ontology led to performance problems on a standard laptop.

8 Conclusion

For the first time a complete implementation of the FRBRoo specification[8] as OWL-DL ontology has been performed and is online available at http://erlangen-crm.org/efrbroo.

Further, this work shows first approaches how to evaluate the FRBRoo semantic model in using the power of description logic to systematically detect inconsistencies and in applying methods of software engineering testing principles.

These approaches should be extended in the future in collaboration with domain experts in the area of libraries. With the help of domain experts the test cases could be extended to cover all FRBRoo classes and properties. The WissKI framework provides the tool for this type of modelling of domain knowledge.

In order to take the next step towards real world application using high-performance hardware and professional software frameworks, improving the test cases to be computational more efficient and advancing the test environment application to best suit the test cases are a necessary next steps.

[8] FRBRoo Version 1.0.2 (CIDOC CRM 5.0.4)

9 References

[AH2011] Allemang, D., & Hendler, J. (2011). Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL – Second Edition. Waltham: Morgan Kaufmann Publishers.

[CRM2011] Crofts, N., Doerr, M., Gill, T., Stead, S., & Stiff, M. (December 2011). The CIDOC Conceptual Reference Model: Resources: Version 5.0.4 of the Definition of the CIDOC Conceptual Reference Model. Online available 2011-02-04 at http://www.cidoc-crm.org/docs/cidoc_crm_version_5.0.4.pdfv

[FRBRoo2012] Bekiari, C., Doerr, M., & Le Bœuf, P. (January 2012). The CIDOC Conceptual Reference Model: Working Drafts and Releases: Version 1.0.2 of the FRBRoo Model. Online available 2012-02-03 at http://www.cidoc-crm.org/docs/frbr_oo/frbr_docs/FRBRoo_V1.0.2.pdf

[ECRM2012] Goerz, G., Hohmann, G., Fichtner, M., & Scholz, M. (2012). Erlangen CRM / OWL. Online available 2012-01-14 at http://erlangen-crm.org/

[FRBR2009] IFLA Study Group on the Functional Requirements for Bibliographic Records (February 2009). Functional Requirements for Bibliographic Records: Final Report. Online available 2012-05-19 at http://www.ifla.org/files/cataloguing/frbr/frbr_2008.pdf

[LL2011] Lambrix, P., Liu, Q. (2011). Debugging is-a structure in networked taxonomies, Proceedings of the Fourth International Workshop on Semantic Web Applications and Tools for Life Sciences - SWAT4LS11, 58-65, London, UK

[LQM2011] Lambrix, P., Qi, G., & Melicke, C. (2011). Overview of defects in ontologies and ontology networks. Tutorial: Debugging Ontologies and Mappings in Ontology Networks. Bonn: ISWC 2011.

[Stu2008] Stuckenschmidt, H. (1-2. June 2008). Debugging OWL Ontologies - A Reality Check, EON-SWSC-S008. Online available 2012-02-14 at http://ceur-ws.org/Vol-359/Paper-1.pdf

[Tay2007] Taylor, A. G. (2007). Understanding FRBR: What it is and how it will affect our Retrieval Tools. Westport (USA): Libraries Unlimited.