MEDINFO 2010
C. Safran et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-588-4-1060

The DebugIT Core Ontology: semantic integration of antibiotics resistance patterns

Daniel Schober[a], Martin Boeker[a], Jessica Bullenkamp[a], Csaba Huszka[b], Kristof Depraetere[b], Douglas Teodoro[c], Nadia Nadah[d], Remy Choquet[d], Christel Daniel[d], Stefan Schulz[a]

[a] *Institute for Medical Biometry and Medical Informatics, Freiburg University Medical Center, Germany*
[b] *AGFA Healthcare, Gent, Belgium*
[c] *Medical Informatics Service, University Hospitals of Geneva, Switzerland*
[d] *INSERM, UMR_S 872, Eq. 20, Paris, F-75006 France; Université René Descartes, Paris, F-75006 France*

Abstract

Antibiotics resistance development poses a significant problem in today's hospital care. Massive amounts of clinical data are being collected and stored in proprietary and unconnected systems in heterogeneous format. The DebugIT EU project promises to make this data geographically and semantically interoperable for case-based knowledge analysis approaches aiming at the discovery of patterns that help to align antibiotics treatment schemes. The semantic glue for this endeavor is DCO, an application ontology that enables data miners to query distributed clinical information systems in a semantically rich and content driven manner. DCO will hence serve as the core component of the interoperability platform for the DebugIT project. Here we present DCO and an approach thet uses the semantic web query language SPARQL to bind and ontologically query hospital database content using DCO and information model mediators. We provide a query example that indicates that ontological querying over heterogeneous information models is feasible via SPARQL construct- and resource mapping queries.

Keywords:

Ontology, Knowledge sharing, Antibiotics, Semantic heterogeneity, Systems integration, Information storage and retrieval

Introduction

After fifty years of unreflected and abundant use of antibiotics, the emergence of resistant and potentially untreatable pathogens has led to increased healthcare costs and patient risks. Comparison of antimicrobial resistance data across Europe led to the discovery of a wide diversity in practices. For instance, in Urinary Tract Infection (UTI) by uropathogenic E.coli, Trimethoprim is used as first line medication, while Fluoroquinolones are preserved as backup for patients with contraindications, allergies and where first line drugs fail. Although increasingly widespread use of Fuoroquinolones will promote bacterial resistance, an uncontrolled prescription can be observed in some areas. While resistance to Fluoroquinolones averages 5% in Europe, it can be as high as 24% in Spain [1].

The **DebugIT** project (**D**etecting and **E**liminating **B**acteria **U**sin**G** **I**nformation **T**echnology, http://www.debugit.eu/), a large scale data integration project funded within the 7th EU Framework Program, intends to analyze these practices and their outcomes across Europe and to exploit this knowledge to detect patient safety related patterns in hospital data, i.e. to discover indicators for better treatments and antibiotics resistance prevention.

In this project, a semantic infrastructure allowing bidirectional communication between locally distributed Clinical Data Repositories (CDR) and the DebugIT knowledge mining services is being built[1]. Most of the required semantics are provided by the DebugIT Core Ontology (DCO), which represents the formal and explicit computer-interpretable meaning throughout the project using semantic web technologies. DCO focuses on patients, diseases, pathogens, their analyses and medications.

We present DCO's current state of development and demonstrate how DCO is used within DebugIT to bridge the semantic gap between two heterogeneous clinical information systems. In order to do so, we briefly introduce some core aspects of the DebugIT interoperability platform which enables the semantic query integration over different hospital CDRs via DCO. The overall DebugIT knowledge mining architecture is described in [2].

Materials and Methods

Querying within the DebugIT interoperability platform

In order to understand how DCO is used for building cross-hospital queries, we here describe the query building process and the involved modules:

1. A data miner receives a clinical question and determines the needed datasets in the list of different hospital CDRs by iterating through steps 2 to 4 for each of the targeted CDRs. Soon the system will eventu-

[1] D. Teodoro, R. Choquet, E. Pasche, J. Gobeill, C. Daniel, P. Ruch, C. Lovis, Biomedical Data Management: a Proposal Framework. MIE 2009

ally possess a large battery of solved questions and their queries, which in turn give rise to the needed datasets for certain type of questions, therefore simplifying any subsequent query making process.

2. Previously stored SPARQL[2] dataset queries for the selected CDR are searched in order to be reused. Also partially matching queries may be used.
3. If no adequate query is found, a new SPARQL query is created (or an existing one adapted) by the data miner. We bridge the gap between the different CDRs by linking the specific CDR concepts via the DDO to DCO classes in a SPARQL query.
   a. First the **CONSTRUCT** clause is created using DCO according to a graph pattern template that specifies how results of the query should be returned. This CONSTRUCT clause can be reused for further CDR SPARQL queries the data miner is building and can be the same for all of the CDRs.
   b. Then the **WHERE** clause is created using an RDF file mediator called 'DataDefinitionOntology', DDO, expressing the information model and the mapping between its local concepts and DCO. The data miner needs to build a SPARQL query for each targeted CDR, because they are independent storage systems and normally have different DDOs. If a DDO concept is missing, the local CDR maintainer is notified who should fill this gap by defining the missing concepts.
4. The SPARQL query is sent to the targeted CDR and the returned RDF result is analysed to determine if it provides the needed data to solve the clinical query. If this is not the case, steps 2 to 4 are repeated to refine the SPARQL query. If the result is adequate the steps are repeated for the next selected CDR.
5. Finally, the SPARQL queries are sent to all distributed hospital SPARQL endpoints[3] to access their CDRs. The results are then aggregated into one RDF data result set, which can be exported to different formats, depending on the needs of the used data mining approach.
6. The constructed dataset SPARQL queries can be stored together with the RDF result and additional metadata in a knowledge repository for later reuse.

The gap between the different CDRs is bridged by linking the specific CDRs to DCO concepts in a mapping SPARQL query. In the query process described, we apply two kinds of ontologies to communicate between different modules of the interoperability platform. DCO classes and relations are used for formulating a hospital independent clinical query using SPARQL. It is mapped to the local IM via an RDF converted database schema[4], the DDO, acting as a query mediator to the proprietary hospital CDR. The physical IM was converted into RDF syntax by a syntax conversion tool to render it accessible to the SPARQL WHERE clause.

DCO design principles

We subscribe to a realist perspective towards biomedical ontologies as detailed in [3], however this is not in conflict with integrating information entities (see footnote 6).

Whereas according to [4] domain ontologies describe the vocabulary for a generic domain (medicine) and task ontologies describe a generic task or activity (e.g. diagnosing), DCO has to be classified as an application ontology according to this system, because DCO describes terms depending both on a particular domain (infectious disease) and task (data mining). From an engineering standpoint, we apply the normalization approach of [5] and use single asserted parenthood throughout the taxonomy. This will facilitate the orientation in the taxonomy and its maintenance. A reasoner infers multiple parenthood from the formal restrictions.

Ontology builders and users root their modeling decisions and interpretations into upper-level assumptions, whether they make it explicit in an upper-level ontology or not. We build DCO as an extension of the already existing upper level ontology BioTop [6], which renders the meaning of classes and relations explicit and less ambiguous. It helps to ensure a rigid modeling view and eases modeling decisions by providing basic constraints on a high level that can readily be exploited. To allow non-ontologist biomedical experts to view and check parts of the ontology we apply user-friendly ontology visualizations as generated by the OwlPropViz Protégé plugin[5] (see Figure 1).

Scope delineation

In order to maintain the ontology manageable and not to fall into "analysis paralysis", a restriction of the representation to an area of more immediate interest is mandatory. The top requirement for DCO is the coverage of the conceptual space for the detection of harm patterns and the exchange of clinical information, focused on infectious diseases. We decided to model the full circle for a concrete application and querying scenario first, rather than going for broad coverage. This will result in a better idea of how time series of events and branching within processes can be handled and it will contribute to test where the DCO upper level model needs to be updated in order to capture all information in the different CDRs.

Use Case and Competency questions (CQs)

A simple and common scenario, the antibiotic therapy of UTI with the most commonly used drugs Fluoroquinolones and Trimethoprim/Sulfametoxazol (TMP/SMX), has been chosen as the core of our first modeling iterations. We first look at a prototypical 'treatment course' of patient urine sample collection, culturing and antibiogram testing with and without intro-

[2] Simple Protocol and RDF Query Language, http://www.w3.org/TR/rdf-sparql-query/
[3] E.g. http://debugit1.spim.jussieu.fr/ for the Paris hospital
[4] E.g. a DDO with a PREFIX inserm: http://debugit1.spim.jussieu.fr/resource/vocab/ as in example query
[5] http://protegewiki.stanford.edu/index.php/OWLPropViz

ducing empirical therapy with TMP/SMX. The result of the antibiogram can then influence the empirical therapy (adaption) or directly result in a targeted antibiotic therapy.
To be able to verify whether DCO is sufficiently complete to represent our use case, we have collected a set of ten competency questions [7] from clinicians. The ontology needs to contain a necessary and sufficient set of axioms to represent these questions. As such, they will later serve as benchmarks for the DCO evaluation.

From the full set, we choose CQ #5 that we want DCO to be able to answer in the DebugIT prototype. We will use this in all examples in the remainder of this article: "select all Patients that have UTI caused by E. Coli and that are Trimethoprim resistant" The abstract formulation of this CQ is: "Select patients with treatment courses, where disease x caused by agent y, and agent y has a certain quality z (i.e. has susceptibility test result: resistant)". The formalization of this CQ in DCO is illustrated in the DCO query example below.

Term harvesting to populate DCO

We have chosen a data-driven approach in order to acquire a first set of terms for the ontology. Whereas the project seeks to reuse existing ontologies, major parts had to be built from scratch. To gain input for DCO development we harvest terms via the following channels:

- harvesting the set of CQs and abstractions thereof
- harvesting the partners hospitals' CDR schemata
- harvesting concepts of terminologies already in use in the clinical domain (e.g. SNOMED CT)

Concepts from the 'information artefact' realm were integrated via a so called information model ontology (IMO) that was build by 'ontologization' of a semiautogenerated RDF model of an HL7 v3 based information model[6]. IMO mainly amends DCO with 'information entity' concepts found in HL7.
These sources permitted a first representational scaffold to represent the domain, which since then has been incrementally refined.

Ontology modularization and imports

Besides BioTop, for which bridges to all major top level ontologies exist, the following external domain ontologies are aligned with DCO:

An Image mining Ontology IRON.owl has been created[7], which also describes an approach to handle numeric values in owl-DL.

We use an ontology of medical evidence to allow application users to choose between different sources of evidence (e.g. patient records, clinical trials, data mining results). This ontology also describes data exchange concepts like 'request' and 'response' to facilitate interoperability in message exchange systems, e.g. for querying. These operational feature descriptors will soon be factored out into a separate task ontology.

[6] A paper describing this approach has been accepted for MEDINFO 2010 by D. Ouagne et al.
[7] http://www.cs.ucy.ac.cy/itab2009/ (paper accepted)

Mapping to external vocabularies

Whereas DCO follows strict architectural guidelines it is devised to co-exist with less expressive ontologies by some of our collaborators. Specifically, we agreed to re-use the following external vocabularies within the DebugIT project:

- For **diseases** we will re-use and adapt the SNOMED CT finding hierarchy. Currently about 2/3 of the present DCO classes are mapped to matching SNOMED CT terms.
- For **anatomical entities** needed to describe disease and specimen locations we are re-using and adapting portions of the Foundational Model of Anatomy.
- For **bacteria** we are reusing the NEWT taxonomy.
- For **drugs**, we are using the WHO ATC codes.

DCO administration and access

DCO is maintained using a shared Subversion (SVN) repository[8] that allows easy detection of work progress using the log files and allows for file revision history tracking, revert to previous file states and a diff function to detect atomic changes made in single files. All more immediate exchange of ideas and progress monitoring is realized via weekly teleconferences along the SCRUM[9] project management methodology.

Administrative and editorial metadata schemes

We have developed a metadata schema optimized to the project's needs via a self-standing owl file that contains all necessary annotation properties[10]. This allows us to use the RDF:comment field for its intended purpose of capturing comments as well as action items for all entities.

To keep track of abundantly used core entities, we use the bookmark plugin[11] in Protégé 4. This helps in the selection process of ontology modules, especially for repeated evaluations and visualizations of certain views.

To ease DCO development and to foster a common view on use case relevant subsets of classes we have created a DebugIT specific Protégé Tab that shows

- the Bookmark view on selected DCO classes
- the OWLPropViz view to see a graph of DCO nodes linked via edges representing relations
- a cloud view on DCO classes, that displays them according to their subclass count or other criteria.

Results

The DCO ontology and the DebugIT Protégé 4 Tab are available in the project SVN. To access the ontology conveniently in a web browser, we have set up an owlDoc generated HTML serialisation[12].

[8] http://www.greeninghealthcare.org/repository/debugit/trunk
[9] http://www.scrum.org/scrumguides/
[10] http://purl.org/imbi/ru-meta.owl#
[11] http://code.google.com/p/co-ode-owl-plugins/wiki/Bookmarks, A selected set of entities is saved along with the ontology annotations for future reference
[12] http://www.imbi.uni-freiburg.de/~schober/dco_owlDoc/

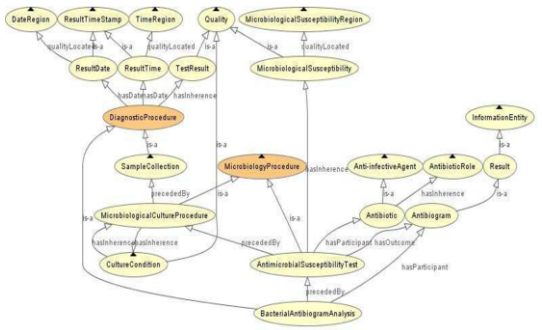

Figure 1- Graph based view on a DCO module

Figure 1 displays a use case relevant view on DCO as a graph created by the OwlPropertyViz plugin. A simple timeline of Processes, starting with the 'Bacterial Antibiogram Analysis' is shown along the *preceded_by* relation allowing for a simple model of relative time flow.

DCO Metrics

The current description logic expressivity is SRIQ(D). We are using the Hermit DL reasoner[13], which takes ~4 seconds to classify DCO including BioTop on an average PC. Table 1 illustrates the metrics of DCO and BioTop.

Table 1- Ontology metrics at submission time

Ontology Idiom	Count (all)	DCO	BioTop
Classes	1029	686	343
Object Properties (relations)	82	21	61
Datatype Properties	5	5	0
Subclass Axioms	1162	779	383
Equivalent Class Axioms	123	40	83
Disjoint Axioms	75	1	74
Sub Object Property Axioms	44	1	43
Transitive Object property Axioms	14	0	14
Object property Domain Axioms	28	0	28
Object property Range Axioms	28	0	28

DCO in a SPARQL mapping query - an example

SPARQL endpoints (see footnote 3) have been implemented in three hospitals. Mappings between the information models of local data repositories and DCO have been performed in order to run SPARQL queries. To illustrate how a mapping SPARQL query links DCO concepts to IM schema elements, we look at competence question CQ #5, which has been exemplarily modelled using DCO and BioTop concepts in the CONSTRUCT clause and entities of a particular hospitals IM schema in the WHERE clause:

```
PREFIX dco:    <http://www.debugit.eu/ontology/1.0/dco.owl#>
PREFIX inserm: <http://debugit1.spim.jussieu.fr/resource/vocab/>
CONSTRUCT {

          _:therapy             a dco:TreatingUrinaryTractInfection;
                                  biotop:hasPatient _:patient.
          _:urineSampling       a dco:UrineSampleCollection;
                                  biotop:hasParticipant _:patient;
          dco:hasOutcome _:urineSpecimen.
          _:culturing           a
dco:MicrobiologicalCultureProcedure;
                                  biotop:hasParticipant
_:urineSpecimen;
                                  dco:hasOutcome [ a dco:Result;
biotop:encodes _:bacteriaName ].
          _:bacteriaName        biotop:qualityLocated [ a
biotop:SpeciesEscherichiaColiRegion ].
          _:susceptibilityTest1 a
dco:AntimicrobialSusceptibilityTest;
                                  biotop:precededBy _:culturing;
          dco:hasParticipant [ a dco:Trimethoprim ];
          dco:hasOutcome [ a dco:Result; biotop:encodes
?susceptibility1 ] .
          ?susceptibility1      a dco:MicrobiologicalSusceptibility;
                                  biotop:qualityLocated [ a ?result1
].
          _:susceptibilityTest2 a
dco:AntimicrobialSusceptibilityTest;
                                  biotop:precededBy _:culturing;
          dco:hasParticipant [ a dco:CoTrimoxazole ];
          dco:hasOutcome [ a dco:Result; biotop:encodes
?susceptibility2 ] .
          ?susceptibility2      a dco:MicrobiologicalSusceptibility;
                                  biotop:qualityLocated [ a ?result2
].
}
WHERE {
          GRAPH<http://debugit.eu/inserm-map.n3> {
                    ?antibiotic1 a dco:Trimethoprim.
                    ?bacteria    a
biotop:SpeciesEscherichiaColiRegion.
                    ?r1          a ?result1.
?uti           a dco:UrineSampleColleciton.
?result1 rdfs:subClassOf dco:MicrobiologicalSusceptibilityRegion.
                    FILTER (!sameTerm(?result1,
dco:MicrobiologicalSusceptibilityRegion))
          }
          GRAPH<http://debugit1.spim.jussieu.fr/resource> {
                    ?susceptibility1 a inserm:CultureResults;
                                       inserm:culture_id ?culture;
                                           inserm:bacteria_analyzed
?bacteria;
                                       inserm:antibiotic_tested
?antibiotic1;
                                       inserm:antibiotic_RESULT ?r1.
                    ?culture         a inserm:culture;
                                       inserm:culture_sample_type
?uti.
          }
}
```

Challenges

The pursued SPARQL mapping approach requiring a mediation layer is still experimental. It depends on novel formats and tools, which challenges the stability of such a complex project. Considering the large data volumes performance might become a problem[14], and it is still an open question whether the whole setup will be scalable and well-performing. The on-the-fly IM schema to RDF conversion and SPARQL querying over DDO-DCO mappings is slow on certain constructs[15].

The mapping between an ontology and a clinical data repository is not trivial as the recording of clinical data blends ontological with epistemological, pragmatic and contextual aspects. The difficulty will be to find a metamodel that can consistently deal with the rather different implicit top level assumptions in the heterogeneous information models (see footnote 6). Time modeling will be another complex problem in the near future. DCO currently includes a relation 'preceded

[13] http://hermit-reasoner.com/

[14] The clinical data from George Pompidou hospital in Paris (of a year period of time) was migrated into the clinical data repository corresponding to 59000 patients, 89000 stays, 170000 episodes of care, 28000 culture results and 9800 antibiograms.

[15] http://www.w3.org/2007/03/RdfRDB/papers/d2rq-positionpaper/

by' to link processes and allows to model relative time flows (see Figure 1). To allow for absolute time modelling we have to include date-time stamps, e.g. using xsd:dateTime.

Conclusion

Whereas earlier attempts tried to integrate CDRs via purely syntactical integration, e.g. via XML schemata as in [8], recent approaches acknowledge the benefit of a computer interpretable formally defined semantics [9,10]. Not only are the requirements for medical data integration ontologies well investigated [11], recent projects have shown their usefulness in healthcare data integration settings [12]. As in the Advancing Clinico-Genomic Trials on Cancer (ACGT) project [13], which aims at improving Post-genomic clinical trials by providing seamless access to integrated clinical, genetic, and image databases, we use IM model-derived mediator artefacts and SPARQL to resolve syntactic and semantic heterogeneities when accessing wrapped databases. Along these lines DebugIT adopts a federated data warehouse model approach for clinical data integration as described by [14].

Although it seems too early to evaluate the full potential of DCO as core communication channel for the DebugIT interoperability platform, few preliminary properties can already be evaluated. Of the four properties of an ontology that may be quality-assured [15] philosophical validity, compliance with meta-ontological commitments, fitness for purpose and content correctness, we will primarily concentrate on the latter two, because an ontology compliant with all current philosophical theories, following all necessary ontological commitments, and with entirely 'correct' content, may be too complex to be directly usable. The next steps will be identifying and fixing coverage gaps for additional competence questions. We will continue to add logical definitions for at least all bookmarked classes in order to make these accessible to automatic reasoning. We believe the application of CQs and the example given illustrates DCOs 'fitness for purpose' and its 'content correctness' has been ensured via the application of consistency checks and automated reasoning. DCO has reached a level of completeness and formality to start to interoperate data queries across clinical sites as a proof of concept. We have provided a working example for a successful query execution of a query expressed using DCO answering one given competence question.

Acknowledgements

Daniel Schober is funded by the DebugIT project of the EU 7th Framework Program grant agreement ICT-2007.5.2-217139, which is gratefully acknowledged. We acknowledge Hans Cools who has significantly contributed to the DebugIT ontology development and Djamila Raufie who helped curating DCO.

References

[1] Garcia MI, Munoz Bellido JL, Garcia Rodriguez JA, et al. In vitro susceptibility of community-acquired urinary tract pathogens to commonly used antimicrobial agents in Spain: a comparative multicenter study (2002-2004). J Chemother 2007 June; 19(3): 263-70.

[2] Lovis C et al. DebugIT for patient safety - improving the treatment with antibiotics through multimedia data mining of heterogeneous clinical data. Stud Health Technol Inform. 136 (2008), 641-6

[3] Schulz S, Stenzhorn H, Boeker M, Smith B: Strengths and limitations of formal ontologies in the biomedical do-main. RECIIS - Electronic Journal in Communication, Information and Innovation in Health, 2009; 3 (1): 31-45

[4] Guarino N (ed.), Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998. Amsterdam, IOS Press, pp. 3-15.

[5] Rector AL, Rogers JE, Zanstra PE, Van Der Haring E: OpenGALEN: open source medical terminology and tools. AMIA Annu Symp Proc 2003:982.

[6] Schulz S, Beisswanger E, van den Hoek L, Bodenreider O, van Mulligen EM: Alignment of the UMLS semantic network with BioTop: methodology and assessment. Bioinformatics 2009, 25:i69-76.

[7] Grueninger, M and Fox, M (1994). The role of competency questions in enterprise engineering. In IFIP WG 5.7, Workshop Benchmarking. Theory and Practice, Trondheim/Norway.

[8] deVos A, Widergren SE, Zhu J, XML for CIM Model Exchange, Proceedings of the 22nd International Conference on Power Industry Computer Applications, Sydney, 2001, pp. 31-37.

[9] Brailer, D. (2005). Interoperability: The Key to the Future Health Care System, Health Affairs (The Policy J. of the Health Sphere), Vol. 10 (January), 19-21.

[10] Aguilar A, Semantic Interoperability in the context of eHealth. In: HP/DERI/CIMRU Research Seminar, Galway, Ireland, 2005.

[11] Anjum A, Bloodsworth P, Branson A et al., The Requirements for Ontologies in Medical Data Integration: A Case Study, Database Engineering and Applications Symposium, International, pp. 308-314, 11th International Database Engineering and Applications Symposium (IDEAS 2007)

[12] Martin L, Anguita A, Maojo V, Bonsma E, Bucur A, Vrijnsen J, Brochhausen M, Cocos C, Stenzhorn H, Tsiknakis M, Doerr M, Kondylakis H (2008) Ontology based Integration of Distributed and Heterogeneous Data Sources in ACGT. HEALTHINF 2008, Funchal, Portugal.

[13] Weiler G, Brochhausen M, Graf N, Schera F, Hoppe A, Kiefer S: Ontology based data management systems for post-genomic clinical trials within a European Grid Infrastructure for Cancer Research. Conf Proc IEEE Eng Med Biol Soc 2007, 2007:6435-6438.

[14] Stolba N, Towards a Sustainable DWH Approach for Evidence-Based Healthcare; Dissertation, Reviewers: A. Tjoa, T. Mück; Institut für Softwaretechnik und Interaktive Systeme, 2007; Rigorosum: 20.11.2007.

[15] Rogers JE: Quality assurance of medical ontologies. Methods Inf Med 2006, 45:267-274.

Address for correspondence

Daniel Schober, schober@imbi.uni-freiburg.de,
Universitätsklinikum, Institut für Medizinische Biometrie und Medizinische Informatik (IMBI), Stefan-Meier-Strasse 26, D-79104 Freiburg, Germany, Tel: +49 (0)761 2036807