FORMAL ONTOLOGY IN INFORMATION SYSTEMS

Frontiers in Artificial Intelligence and Applications

FAIA covers all aspects of theoretical and applied artificial intelligence research in the form of monographs, doctoral dissertations, textbooks, handbooks and proceedings volumes. The FAIA series contains several sub-series, including “Information Modelling and Knowledge Bases” and “Knowledge-Based Intelligent Engineering Systems”. It also includes the biennial ECAI, the European Conference on Artificial Intelligence, proceedings volumes, and other ECCAI – the European Coordinating Committee on Artificial Intelligence – sponsored publications. An editorial panel of internationally well-known scholars is appointed to provide a high quality selection.

Series Editors:
J. Breuker, R. Dieng-Kuntz, N. Guarino, J.N. Kok, J. Liu, R. López de Mántaras,
R. Mizoguchi, M. Musen, S.K. Pal and N. Zhong

Volume 183

Recently published in this series

Vol. 182. H. Fujita and I. Zualkernan (Eds.), New Trends in Software Methodologies, Tools and Techniques – Proceedings of the seventh SoMeT_08
Vol. 181. A. Zgrzywa, K. Choroś and A. Siemiński (Eds.), New Trends in Multimedia and Network Information Systems
Vol. 180. M. Virvou and T. Nakamura (Eds.), Knowledge-Based Software Engineering – Proceedings of the Eighth Joint Conference on Knowledge-Based Software Engineering
Vol. 179. A. Cesta and N. Fakotakis (Eds.), STAIRS 2008 – Proceedings of the Fourth Starting AI Researchers’ Symposium
Vol. 178. M. Ghallab et al. (Eds.), ECAI 2008 – 18th European Conference on Artificial Intelligence
Vol. 177. C. Soares et al. (Eds.), Applications of Data Mining in E-Business and Finance
Vol. 176. P. Zaraté et al. (Eds.), Collaborative Decision Making: Perspectives and Challenges
Vol. 175. A. Briggle, K. Waelbers and P.A.E. Brey (Eds.), Current Issues in Computing and Philosophy
Vol. 174. S. Borgo and L. Lesmo (Eds.), Formal Ontologies Meet Industry
Vol. 173. A. Holst et al. (Eds.), Tenth Scandinavian Conference on Artificial Intelligence – SCAI 2008
Vol. 172. Ph. Besnard et al. (Eds.), Computational Models of Argument – Proceedings of COMMA 2008
Vol. 171. P. Wang et al. (Eds.), Artificial General Intelligence 2008 – Proceedings of the First AGI Conference
Vol. 170. J.D. Velásquez and V. Palade, Adaptive Web Sites – A Knowledge Extraction from Web Data Approach

ISSN 0922-6389

Formal Ontology in Information Systems

Proceedings of the Fifth International Conference (FOIS 2008)

Edited by

Carola Eschenbach

University of Hamburg, Germany

and

Michael Grüninger

University of Toronto, Canada

IOS Press

Amsterdam • Berlin • Oxford • Tokyo • Washington, DC

© 2008 The authors and IOS Press.

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without prior written permission from the publisher.

ISBN 978-1-58603-923-3
Library of Congress Control Number: 2008936296

Publisher
IOS Press
Nieuwe Hemweg 6B
1013 BG Amsterdam
Netherlands
fax: +31 20 687 0019
e-mail: order@iospress.nl

Distributor in the UK and Ireland
Gazelle Books Services Ltd.
White Cross Mills
Hightown
Lancaster LA1 4XS
United Kingdom
fax: +44 1524 63232
e-mail: sales@gazellebooks.co.uk

Distributor in the USA and Canada
IOS Press, Inc.
4502 Rachael Manor Drive
Fairfax, VA 22032
USA
fax: +1 703 323 3668
e-mail: iosbooks@iospress.com

LEGAL NOTICE
The publisher is not responsible for the use which might be made of the following information.

PRINTED IN THE NETHERLANDS

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.

Preface

Since its inception ten years ago, the International Conference on Formal Ontology in Information Systems (FOIS) has explored the multiple perspectives on the notion of ontology that have arisen from such diverse research communities as philosophy, logic, computer science, cognitive science, linguistics, and various scientific domains.

As ontologies have been applied in new and exciting domains such as the World Wide Web, bioinformatics, and geographical information systems, it has become evident that there is a need for ontologies that have been developed with solid theoretical foundations based on philosophical, linguistic, and logical analysis. Similarly, there is also a need for theoretical research that is driven by the issues that have been raised by recent work in the more applied domains. FOIS is intended to be a forum in which to explore this interplay between the theoretical insights of formal ontology and their application to information systems and emerging semantic technologies. The papers appearing in this year's conference exemplify this interaction in very interesting ways, with papers covering the range from foundational issues and generic ontologies to methodologies for ontological engineering, reasoning, and ontology integration.

Themes emerging from the papers give us a snapshot of current issues within the fields of formal ontology and ontological engineering, as well providing us with a glimpse of future research directions.

Although ontologies were originally motivated by the need for sharable and reusable knowledge bases, the reuse and sharing of ontologies themselves is still limited because the ontology users (and other designers) do not always share the same assumptions as the original designers. It is difficult for users to identify implicit assumptions and to understand the key distinctions within the ontology and whether or not disagreements reflect fundamentally different ontological commitments. The challenge therefore still stands to propose ontological engineering methodologies that emphasize ontology reuse and that identify the characteristics of an ontology that enhance its reusability.

The interaction between ontology and science has emerged as an interesting new theme. First, ontologies can be treated as scientific theories, rather than as engineering artefacts. Second, there are ontologies for scientific theories, such as biology and chemistry, which play a role in integrating multiple data sets. Finally, there is work on developing ontologies of scientific theories; such ontologies play a supporting role in scientific research by providing explicit representations for alternative theories. This is also interesting insofar as one can consider ontologies for scientific domains to be a proposed solution for the sixth of twenty-three challenge problems posed by David Hilbert in an address to the International Congress on Mathematicians in 1900:

> *Mathematical treatment of the axioms of physics:*
> *The investigations on the foundations of geometry suggest the problem: To treat in the same manner, by means of axioms, those physical sciences in which mathematics plays an important part.*

The identification and logical formalization of fundamental ontological distinctions continues to be an impetus for current research. Fundamental distinctions, such as universals vs. particulars, features vs. substrates, and artefacts vs. roles are still a source of many challenging problems. Many of these issues converge in unexpected ways, particularly in the treatment of collective objects. Linguistic expressions have also motivated several areas in formal ontology, particularly in the areas of vague predicates and geographic terminology.

Ontology evaluation, both from a logical and empirical perspective, has also been recognized as a critical phase in ontological engineering. On the one hand, this leads to a deeper understanding of the relationships between ontologies. As ontologies are increasingly being deployed on the web, users are faced with the dilemma of selecting amongst multiple possible ontologies for similar domains. On the other hand, ontology evaluation is based on the relationship between the ontology and its initial conceptualization and intended application. We can rigorously characterize the relationship between the intended models of the ontology and the models of the axiomatization of the ontology, but it is more difficult to evaluate the correspondence between the intended models and their adequacy for the intended application of the ontology.

Finally, the widespread deployment of ontologies also raises the challenge of managing discrepancies that arise between ontologies. Although this is most evident in applications that require the integration of multiple domain (and possibly upper) ontologies, the problem must also be addressed by end users who want to merge concepts from multiple ontologies to create new ontologies that meet the specific needs of some domain. Integrating sets of independently designed ontologies also has ramifications for supporting automated reasoning with the ontologies.

The success of FOIS-08 has been the result of a truly collaborative effort. We would like to thank the members of the Programme Committee for their diligent work and constructive feedback, which have contributed to an excellent conference programme. We would like to thank the three invited speakers, Johanna Seibt, York Sure, and Mike Uschold, for providing their interesting perspectives on formal ontology in information systems. Finally, we would like to thank the Conference Chair, Nicola Guarino, and the Local Chairs, for managing all of the details that have made the conference a productive interaction of researchers from the diverse disciplines that contribute to formal ontology.

Carola Eschenbach
Michael Grüninger

Conference Organisation

Organising Committee

Conference Chair

Nicola Guarino	Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy

Programme Chairs

Carola Eschenbach	University of Hamburg, Germany
Michael Grüninger	University of Toronto, Canada

Local Chairs

Paul Buitelaar	DFKI GmbH, Saarbrücken, Germany
Massimo Romanelli	DFKI GmbH, Saarbrücken, Germany
Michael Sintek	DFKI GmbH, Kaiserslautern, Germany

Programme Committee

Bill Andersen	Ontology Works, USA
John Bateman	Department of Applied English Linguistics, University of Bremen, Germany
Brandon Bennett	School of Computing, University of Leeds, UK
Stefano Borgo	Laboratory for Applied Ontology, ISTC-CNR, Italy
Vladimir Borschev	University of Massachusetts, USA
Joost Breuker	Leibniz Center for Law, University of Amsterdam, The Netherlands
Paul Buitelaar	DFKI GmbH, Saarbrücken, Germany
Roberto Casati	Jean Nicod Institute, CNRS, Paris, France
Werner Ceusters	New York State Center of Excellence in Bioinformatics and Life Sciences, SUNY at Buffalo
Hans Chalupsky	University of Southern California, USA
Philipp Cimiano	Universität Karlsruhe (TH), Germany
Anthony Cohn	School of Computing, University of Leeds, UK
Matteo Cristani	University of Verona, Italy
Carola Eschenbach	Department for Informatics, University of Hamburg, Germany
Jérôme Euzenat	INRIA Rhône-Alpes, France

Christiane Fellbaum	Cognitive Science Laboratory, Princeton University, USA and Berlin Brandenburg Academy of Sciences and Humanities, Berlin, Germany
Roberta Ferrario	Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy
Antony Galton	School of Engineering, Computer Science and Mathematics, University of Exeter, UK
Aldo Gangemi	Laboratory for Applied Ontology, ISTC-CNR, Roma, Italy
Pierdaniele Giaretta	Department of Philosophy, University of Verona, Italy
Michael Grüninger	University of Toronto, Canada
Nicola Guarino	Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy
Udo Hahn	Jena University, Germany
Jerry Hobbs	University of Southern California, USA
Rüdiger Klein	Fraunhofer IAIS, Germany
Werner Kuhn	IFGI, Muenster
Terry Langendoen	University of Arizona, USA
Alessandro Lenci	University of Pisa, Italy
Leonardo Lesmo	Department of Informatica, University of Torino, Italy
Bernardo Magnini	Centre for Scientific and Technological Research, ITC-irst, Trento, Italy
David Mark	Department of Geography, State University of New York, Buffalo, USA
Claudio Masolo	Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy
William McCarthy	Department of Accounting & Information Systems, Michigan State University, USA
Chris Menzel	Department of Philosophy, Texas A&M University, USA
Simon Milton	Department of Information Systems, University of Melbourne, Australia
Philippe Muller	Research Institute for Computer Science, University of Toulouse III, France
John Mylopoulos	Department of Computer Science, University of Toronto, Canada
Daniel Oberle	SAP Research, CEC Karlsruhe
Leo Obrst	The MITRE Corporation, USA
Barbara Partee	University of Massachusetts, USA
Yun Peng	University of Maryland, USA

James Pustejovsky	Department of Computer Science, Brandeis University, USA
Massimo Romanelli	DFKI GmbH, Saarbrücken, Germany
Riccardo Rosati	University of Rome, Italy
Tom Russ	University of Southern California, USA
Johanna Seibt	University of Aarhus, Denmark
Michael Sintek	DFKI GmbH, Kaiserslautern, Germany
Barry Smith	National Center for Ontological Research and Department of Philosophy, University at Buffalo, USA; Institute for Formal Ontology and Medical Information Science, University of Saarland, Germany
John Sowa	Vivomind Intelligence Inc., USA
John Stell	School of Computing, University of Leeds, UK
Veda Storey	Department of Computer Information Systems, Georgia State University, USA
Ludger van Elst	DFKI GmbH, Kaiserslautern, Germany
Achille Varzi	Department of Philosophy, Columbia University, USA
Laure Vieu	Research Institute for Computer Science, CNRS, Toulouse, France
Chris Welty	IBM Watson Research Center, USA

In addition to the programme committee, the following individuals provided reviews for FOIS-2008, and we gratefully acknowledge their contributions:

Krzysztof Janowicz

Edmund Kazmierczak

Manuel Moeller

Gavan Newell

Alessandro Oltramari

Antoine Zimmermann

This page intentionally left blank

Contents

Invited Talks

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-3

Ontology-Driven Information Systems: Past, Present and Future

Invited Paper by Michael USCHOLD
Reinvent Technology, Vancouver, B.C. Canada

Abstract. We trace the roots of ontology-drive information systems (ODIS) back to early work in artificial intelligence and software engineering. We examine the lofty goals of the Knowledge-Based Software Assistant project from the 80s, and pose some questions. Why didn't it work? What do we have today instead? What is on the horizon? We examine two critical ideas in software engineering: raising the level of abstraction, and the use of formal methods. We examine several other key technologies and show how they paved the way for today's ODIS. We identify two companies with surprising capabilities that are on the bleeding edge of today's ODIS, and are pointing the way to a bright future. In that future, application development will be opened up to the masses, who will require no computer science background. People will create models in visual environments and the models will *be* the applications, self-documenting and executing as they are being built. Neither humans nor computers will be writing application code. Most functionality will be created by reusing and combining pre-coded functionality. All application software will be ontology-driven.

Keywords. Ontology-driven information systems, model driven architecture, knowledge-based software engineering, automated software engineering, visual programming, automatic programming, ontology, semantic web, semantic technology, domain modeling.

Introduction

Ever since people started creating software, they sought more and better tools to support the process. One of the best ideas is *raising the level of abstraction*. It has a long history in software engineering and continues today. Assembler languages were an abstraction over machine code. Then came higher level languages like FORTRAN and COBOL. Two additional themes in programming languages were introduced, each further raising the level of abstraction, though in different ways. One was object-oriented languages such as Simula and Smalltalk, and more recently C++ and Java; the other was logic programming languages. In current times, we have model-driven software development whose central tenant is raising the level of abstraction [1].

Another central idea for improving software engineering is to *increase the level of formal methods and structure* in various phases of the software lifecycle. Using ontologies as the models in model-driven software development represents the joining together of these two key ideas and the subject of this talk: ontology-driven information systems (ODIS).

Neither raising the abstraction level nor increasing structure and formality is inherently beneficial. If abstraction is too high, it will have little use. While formality and structure are not always appropriate, their wise use can lead to advantages related to speed of development, reuse, reliability and reduced maintenance costs.

In 1986, a collection of seminal papers was published by Morgan Kauffman called "Artificial Intelligence and Software Engineering" [2]. One very influential paper reported on an ambitious vision of a "Knowledge-Based Software Assistant" (KBSA) [3,4]. The KBSA idea gave rise to a series of annual conferences that evolved into the Knowledge Based Software Engineering conference series, and later the International Conferences on Automated Software Engineering (ASE). Below is a summary introduction from one of the KBSE conference proceedings where I have highlighted the key goals of the project:

> "KBSA: This annual conference provides a forum for exchanging technical and managerial views on ... developing a *knowledge-based life cycle paradigm* for large software projects, the Knowledge Based Software Assistant. ... Software developed using the KBSA is expected to be *more responsive to changing requirements*, *more reliable*, and *more revisable* than software produced using current practices. The KBSA will improve software practices by providing *machine-mediated support to decision makers, formalizing the processes associated with software development and project management*, and providing *a corporate memory for projects*. The KBSA will *utilize artificial intelligence, automatic programming, knowledge-based engineering, and software environment technology* to achieve the [project] goal[s]. (my emphasis)"

In the past quarter-century, we have seen a lot of change, and much progress. Yet in many ways, the original KBSA vision remains largely just that, a *vision*. An early concept demo was created in 1992. There was never any grand synthesis into a single powerful widely used software application development platform that meets most of the original KBSA goal. Why is that? Where are we instead? What lies on the horizon?

In this talk, we examine the somewhat more modest and therefore more attainable vision of *ontology-driven information systems* (ODIS). This idea goes further than ontology-driven software engineering. The latter only implies that ontologies are used in the application development process (e.g. for requirements analysis); the end application need not be driven by an ontology. This distinction is critical for a variety of reasons that we will explore – the main one is that it allows the ontologies and the application to evolve in lock-step.

We consider how ODIS evolved from its early roots in knowledge-based software engineering. We review some key developments that gave us the current state of our art, and we articulate a vision for the future. We believe that the idea of ontology-driven information systems is in the early stages of becoming a practical reality, and that before too long, it will become mainstream.

This paper and corresponding will be akin to a stroll in the park, stopping along the way to smell the flowers that catch our attention, not a botanical expedition to examine and catalog everything in view.

1. Early Roots and Current Technology

The roots of ontology-driven information systems go back to the early work in artificial intelligence and software engineering. Some of the key ideas that originated and evolved from this work include:

1. modeling the domain of the software
2. automated reasoning
3. automatic programming & executable specifications
4. model-driven software development
5. semantic technology (ontologies and the semantic web)

1.1. Domain Modeling

The idea of domain modeling [5] was that there should be explicit models of the domain for which the software application was intended. LaSSIE [6] focused on improving comprehension of software by reverse engineering a domain model and linking it to the code. It was one of the early efforts in linking explicit knowledge models to software. More and more, the domain models for software engineering are becoming formal ontologies using rigorous logic languages.

1.2. Automated Reasoning

Automated reasoning is a very general technology used in conjunction with most of the other ideas to support many phases of the software engineering lifecycle. These include requirements elicitation and analysis, checking consistency of designs, intelligent program analysis, assisting program planners, and proving the correctness of code, to name a few. This is a huge topic that we will not consider further here.

1.3. Automatic Programming & Executable Specifications

Early results in automatic programming were on toy problems and fairly unimpressive. It was a grand challenge, so this was understandable. Today, we have industrial-strength systems that leverage automated reasoning in the form of theorem proving to develop realistic-scale schedules [7] and provably correct avionics software[1], to name two examples I'm familiar with. The dream of full automation is still a ways off in the general case. Often, a human is still needed in the loop.

The idea of saving the human from having to directly write program code evolved in different ways. Above we mentioned the evolution from machine code to assembler to higher-level languages, and beyond. Another way to reduce human coding effort was to create and reuse formal models as a first step in the process of software development. After a series of refinements and transformations with the human in the loop, executable code was created. Some approaches used category theory. [8,9] This approach results in software that is linked to a domain model and is also provably correct.

A related idea is executable specifications. Logic programming and Prolog came along and the idea of having a specification and a program be one and the same was revolutionary at the time. One could write programs that looked like models of a domain, or simple rules in an English-like notation and just execute them. This was similar to the idea of automatic programming, except that there was no intermediate step where code was generated, the specification is executed directly.

---

[1] Automatic Theorem Proving Research Group, http://www.cs.utexas.edu/users/moore/atp/

A precursor to the idea of a model-driven software development (see below), is the idea of using knowledge representation and ontologies to assist in the requirements specification stage of software development [10]. One example is KADS [11], a methodology for developing a special class of software: knowledge-based systems. A key idea was the development of an ontology to specify the key things that the knowledge based system will address. The completed application is closely aligned to, but wholly disconnected from the ontology (i.e. ontology-driven development, but not ontology-driven information systems). The applications evolve and the requirements and other design documents don't - business as usual.

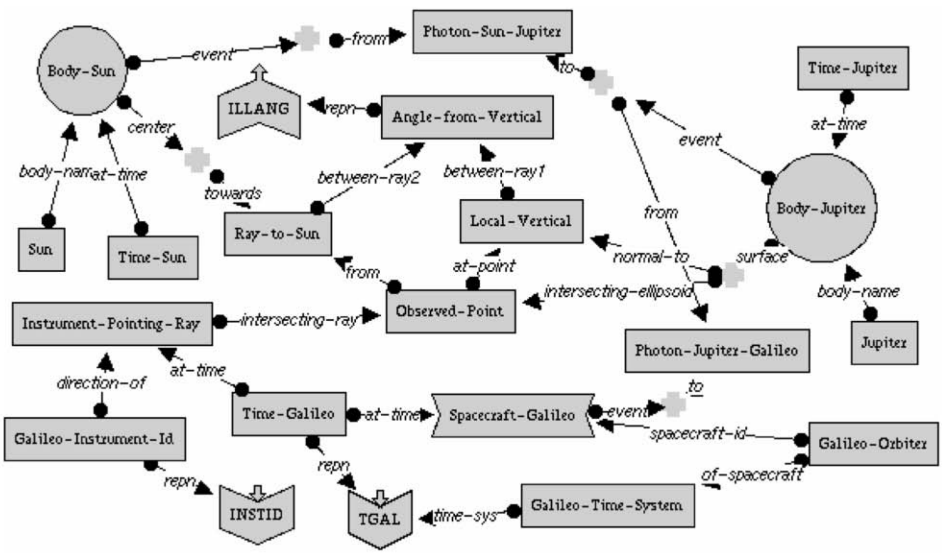
A variation on executable specifications, and the use of domain models is the idea of creating an environment that allowed domain experts that are non-programmers to specify and create software in a specialized domain. This was the subject of my own PhD research [12,13]. I created a prototype environment that allowed an ecologist to create models of the objects and processes in a real or imagined ecological system. The model was converted into executable code in FORTRAN or Prolog. Along these same lines, a much more impressive system was [independently] created and used at NASA: AMPHION (Figure 1) [14]. A large library of FORTRAN routines was exposed to the user as functions in a visual modeling environment. The core of the end software consisted of a series of calls to pre-coded subroutines, as specified by the functional connections in the model. In both of these examples, the users were programming without even knowing it. From their perspective, they were creating models. They were in effect, executable models. There was a very tight connection between the domain model and the code, they evolved together. These are two early examples of model-driven software development and ODIS-like capabilities.

1.4. Model-Driven Software Development

Next, we have the idea of model-driven [software] development (MDD) which attempts to use models as the basis for creating software in a variety of domains, rather than limited to single domains, as in the above examples. Much of the following discussion is drawn from an excellent paper on MDD [1]. The idea is that models raise the level of abstraction; this is intended to simplify and formalize various activities in the software lifecycle. There are various approaches to MDD, including agile model-driven development, domain-oriented programming and Microsoft's software factories as well as the more widely known and used *model-driven architecture* (MDA).

Model-driven architecture is characterized by having a platform independent model (PIM) that specifies the required functionality of the end application. The PIM is then transformed into one or more platform-specific models which are much closer to programming code. UML is the most often used modeling language for MDA. There are different ways to use UML. Some use it to sketch out ideas and get agreement before. Some build grand designs that are detailed blueprints that are handed off to programmers. These two uses both keep a clear separation between the models and the code objects. That distinction is significantly blurred in a third way to use UML, as a kind of high level language that can at least in part, be compiled into executable code. Programmers fill in any missing pieces. This is sometimes called model programming.

The model programming approach is a variation of the executable specification idea, where the specification is a model. With the model programming approach, there is an explicit connection to the end software. Often all that is generated is stub code,




```
Only the procedural statements are listed below. "State" means position and
velocity.
```

`SUBROUTINE ILLUM ( INSTID, TGAL, ILLANG )`	
`CALL BODVAR ( JUPITE, 'RADII', DMY1, RADJUP )`	`RADJUP`: Jupiter's radii.
`CALL SCS2E ( GALILE, TGAL, ETGALI )`	`ETGALI`: `TGAL` converted to an internal time system.
`X0 = I2SC ( INSTID ) X0 = Galileo's ID.`	
`CALL SPKSSB ( X0, ETGALI, 'J2000', PVX )`	`PVX`: the state of Galileo at time `ETGALI`.
`CALL SCE2T ( INSTID, ETGALI, TKINST )`	`TKINST`: `ETGALI` converted to another system.
`TJUPIT = SENT ( JUPITE, GALILE, ETGALI )`	`TJUPIT`: the time when light left Jupiter.
`CALL BODMAT ( JUPITE, TJUPIT, MJUPIT )`	`MJUPIT`: the rotation matrix to Jupiter's frame.
`CALL ST2POS ( PVX, PPVX )`	`PPVX`: extract Galileo's position.
`CALL SPKSSB ( JUPITE, TJUPIT, 'J2000', PVJUPI )`	`PVJUPI`: the state of Jupiter at time `TJUPIT`.
`CALL CKGPAV ( INSTID, TKINST, TIKTOL, 'J2000', C, DMY2, DMY3, DMY4 )`	`C`: rotation matrix to the instrument's frame.
`TSUNN = SENT ( SUNNAI, JUPITE, TJUPIT )`	`TSUNN`: the time when light left the Sun.
`CALL ST2POS ( PVJUPI, PPVJUP )`	`PPVJUP`: extract Jupiter's position.
`CALL MATROW ( C, 3, V )`	`V`: the instrument pointing vector.
`CALL SPKSSB ( SUNNAI, TSUNN, 'J2000', PVSUN )`	`PVSUN`: the state of the Sun at time `TSUNN`.
`CALL VSUB ( PPVX, PPVJUP, DPPPP )`	`DPPPP`: the vector from Jupiter to Galileo.
`CALL MXV ( MJUPIT, V, XV )`	`XV`: `V` in Jupiter-centered coordinates.
`CALL ST2POS ( PVSUN, PPVSUN )`	`PPVSUN`: extract the Sun's position.
`CALL MXV ( MJUPIT, DPPPP, XDPPPP )`	`XDPPPP`: `DPPPP` in Jupiter-centered coordinates.
`CALL VSUB ( PPVSUN, PPVJUP, DPPPP0 )`	`DPPPP0`: the vector from Jupiter towards the Sun.
`CALL SURFPT ( XDPPPP, XV, RADJUP(1), RADJUP(3), P, DMY5 )`	`RADJUP(2), P`: The vector from the center of Jupiter to the observed point. (`SURFPT` does intersection)
`CALL MXV ( MJUPIT, DPPPP0, XDPPP0 )`	`XDPPP0`: `DPPPP0` in Jupiter-centered coordinates.
`CALL SURFNM ( RADJUP(1), RADJUP(2), RADJUP(3), P, PP )`	`PP`: surface normal vector at the observed point.
`CALL VSUB ( XDPPP0, P, DPXDPP )`	`DPXDPP`: vector from observed point to the Sun.
`ILLANG = VSEP ( PP, DPXDPP )`	`ILLANG`: the angle between PP and `DPXDPP`.

Figure 1. Visual Programming using AMPHION: *A planetary scientist draws diagrams that are close to their own way of thinking to specify the objects relationships and functions to get the desired results. Components on the diagram and their relationships specify the requirements that are automatically complied into FORTRAN code. The final program largely consists of calls to pre-coded subroutine. The diagram is an executable model; this is 'model programming' for a narrow domain of applications. Courtesy of NASA.*

which means that the main body of code is still largely disconnected from the model. If the model changes, it is possible to re-generate the stub code, but then all the 'real code' needs to be re-written. So the potentially large gain in maintainability, is only realized in small part. Some of the UML models have executable semantics which allows code to be generated directly from the models (e.g. activity diagrams). This goes a big step beyond the sketch and blueprint uses because for the first time in a general application development paradigm, there is a chance for some [however small] portion of the system requirements and design documents to evolve in lock step with the end application, rather than be separated from it and become out of date.

While MDD ideas do get significant use in the commercial world, they are far from mainstream, many issues remain. There is no consensus as to how far the model programming idea can go, or how often it is desirable under what circumstances.

One of the complaints about UML as a modeling language is that it is too informal, it is just a diagramming notation, initially without even a text version. We now have UML 2.0, [15] and the meta object facility (MoF) to address some of these concerns[2].

1.5. Semantic Technology

A "semantic technology" community concerned with ontologies and the semantic web emerged independently from both the OMG/MDA and software engineering communities. The semantic web community emerged from the collaboration between the W3C and the ontology community that produced RDF[3] and OWL[4]. The ontology community had evolved [chiefly] from the knowledge representation and reasoning community within artificial intelligence. The focus was on how to use ontologies based on languages with formal semantics and automated inference. KADS represented early work related to software engineering of knowledge based systems; the used an ontology much as UML is used in the sketching and blueprinting modes, as discussed above.

For the most part, the ontology and semantic web community was not concerned about software engineering in general, only how to engineer software systems that were going to use ontologies and inference in some way. There was a solution looking for a problem. The community searched and searched, built more and more ontologies, and more and more tools, and after a dozen or so years, they were still looking for a way to make a commercial impact. Mostly, what we saw was lists of potential benefits and research prototypes. Eventually, it all started coming together in the mid-double-ohs when the long-awaited explosion of commercial deployments looked like it might have begun for real.

In the past few years, there has grown a substantial commercial awareness and take-up of semantic technologies. The community had spent years hypothesizing, discovering, articulating and demonstrating the *potential* benefits and value propositions of semantic technology. Finally these were being reported as real experiences from deployed systems. The ones most frequently quoted are software engineering benefits, such as being easier or faster to build or maintain the same software, compared to conventional approaches. One example is the creation of the Data Patrol web service for tracking sensitive personal information on the web by the company, Garlik [16]. The reason it is easier to maintain systems is that semantic technologies seem to offer inherently more flexibility in a changing world. Conventional approaches are more suited for relatively static environments where change is slow or non-existent. Yahoo has reported similar software engineering benefits in its use of semantic web technologies to roll out various services.[17].

---

[2] OMG's MetaObject Facility. http://www.omg.org/mof/

[3] Resource Description Framework (RDF). http://www.w3.org/RDF/

[4] OWL Web ontology Language Overview. http://www.w3.org/TR/owl-features/

Inevitably, people who became aware of both the semantic web and OMG/MDA communities started asking the excellent question: "what is the difference between a UML class diagram and an ontology?". They were much more alike than different, and as a result the two communities have gotten together to make UML compatible with the semantic web languages OWL and RDF. An initial step was the production of two notes by the W3C Semantic Web Best Practices and Deployment Working Group:

- Ontology Driven Architectures and Potential Uses of the Semantic Web in Software Engineering[5]
- A Semantic Web Primer for Object-Oriented Software Developers[6]

This was followed by a formal collaboration between W3C and OMG, resulting among other things in the production of the Ontology Definition Metamodel (ODM). The authors "believe that this specification represents the foundation for an extremely important set of enabling capabilities for Model Driven Architecture (MDA) based software engineering, namely the formal grounding for representation, interoperability, and automation of business semantics." [18]. Tools for converting UML to OWL are also available. The coming together of the OMG and Semantic Web communities is a natural evolution of model- to ontology-driven development of software. In this case, the model is a formal ontology.

2. Today's Bleeding Edge

Progress is happening, at an accelerating rate. Research results are maturing and being transitioned to industry. Many companies are in the market, growing new capabilities. In some cases companies that have been beavering away for years have recently come forward to describe some remarkable capabilities.

Good examples of being smart about using an ontology to directly drive an application are map mashups and ontology-driven forms. This is being done by TopQuadrant, a leading semantic web technology vendor. They are evolving their platform so that one can develop ontologies in the ontology editor, and the ontologies directly drive portions of the applications that run off a client-server platform. Consider a map mashup. A geolocation ontology has properties for latitude and longitude. This can then be directly linked to data sets and sent to Google maps for displaying things like restaurants or kayaking put-in and take-out points along your favorite whitewater rivers (Figure 2). Ontologies can also directly drive the content of forms. The ontology serves as the metadata for the information that is input into the forms by the user. If a new property is added to the ontology, or one is removed, the form in the running application is automatically updated. Customers love this particular feature. The approach is summarized below:

> "In contrast to conventional Model-Driven Architecture known from object-oriented systems, semantic applications use their data models not only at design time, but also as run-time components. The rich declarative semantics of ontological data models can be exploited to drive user interfaces and to control an application's behavior [19]."

[5] http://www.w3.org/2001/sw/BestPractices/SE/ODA/
[6] http://www.w3.org/2001/sw/BestPractices/SE/ODSD/

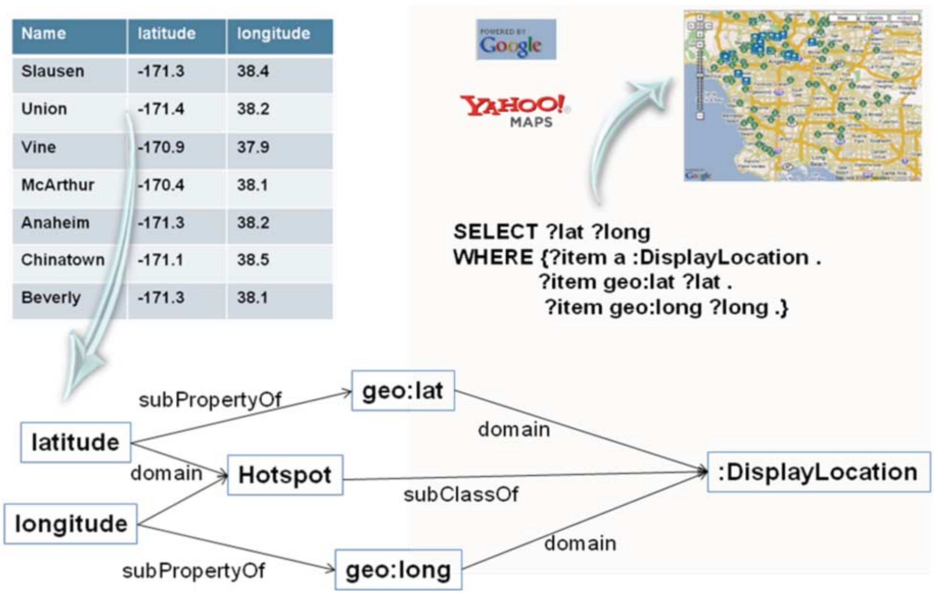


Figure 2. Ontology-Driven Map Mashups. *A local geolocation ontology is used as a data model for storing lat/long coordinates for various locations The local ontology is linked to a public geolocation ontology (with namespace 'geo') that drives public mapping tools. This enables locations from several datasets to be placed on the same map. Figure courtesy of TopQuadrant.*

Note that for these examples, the application developer is not writing any code, but they are developing application functionality, just by using an ontology integrated with an application development support environment. Another example where the emphasis is on modeling, not programming is the ACUIty approach. The idea is to "use semantic technology to drive adaptive user interfaces based on models of users, their work and human computer interaction [20]." Their talk is aptly titled "Stop Programming, Start Modeling...". Automatically generating UI code from models is a great start. The ACUITy approach is not purported to be a general ontology-driven application development environment.

Next we introduce some breakthrough systems and capabilities that were recently introduced to the semantic technology community by two separate companies. Unique to these systems is that they are *primarily* model-driven (as opposed to only isolated parts like the UI, being model-driven), the majority of the application is developed without humans writing code, and they are fairly general purpose application development environments (i.e. not limited to niche domains).

2.1. Mission Critical IT

The first company is Mission Critical IT. Their system and approach appears to be the most advanced in the area of ontology-driven software development with automatic code generation. In the words of the CEO:

> "ODASE (tm) [is] an ontology-driven development approach and platform, which cleanly separates the Business Domain knowledge (at the description level) from the Software

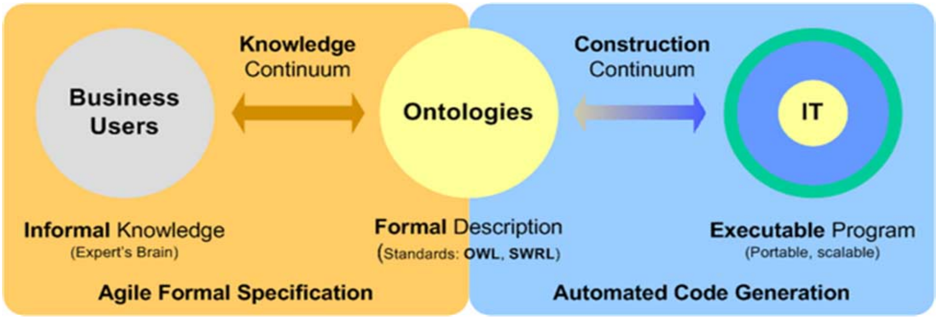


Figure 3. Ontology-Driven Architecture for Software Engineering (ODASE™) *The ontology and other formal models specify a requirements contract with business users, and are the basis for automatic code generation. The breakthrough is a combination of several capabilities: 1) the majority of the code is generated automatically from formal models 2) the models drive the application, and thus 3) the models and the application evolve in sync and 4) there is a warranty on the code. This results in dramatic cost reductions in application development and maintenance. Figure courtesy of Mission Critical IT. [22]*

> Engineering knowledge (at the execution level). The process for transferring the Business knowledge from the ontology to the programming language is by automatic generation of source code. The power of ODASE is that the model specification, the code generation, and the runtime reasoning use the same formal description. [21] (Figure 3)."

At the business domain level, OWL (for ontologies), SWRL (for complex business rules), and colored Petri Nets (for specifying behavior) are used. Domain experts are able to make substantial contributions in this phase, alongside knowledge engineers. The model is automatically converted to a logic/functional programming language (Mercury)[7]. This ensures that the formal logic basis is carried from the model to the execution. The logic program includes the user interface (e.g. for rich Internet applications) and a Semantic Service Broker for integration with back ends using different groundings (e.g. MQ, JMS, SOAP). For efficiency, the logic/functional program is compiled into one of various a backend languages (e.g. C). The resulting code is executed on a new kind of application server which includes all the key components required at run-time. This includes, among other things, UI logic drivers, workflow engine, session manager, state manager, RDF store, and run-time OWL/SWRL reasoner.

This is not a research prototype, this is an infrastructure that is used to deliver applications to paying customers. There is one documented case where their approach took 65% less time to deliver, than the next closest quote. This arises from various factors [22]:

- approximately 70% of the code is automatically generated[8]
- modeling causes better communication between business and IT people resulting in more accurate requirements,
- the formal methods are more reliable,
- changes in the model are immediately reflected in the application.

---

[7] The Mercury Project, http://www.mercury.cs.mu.oz.au/index.html
[8] Michel Vanden Bossche, email communication, August 2008

Due to the logic foundation and largely automatic code generation, they are able to do something that is unheard of: *they offer an explicit warranty on the code*. This appears to be a major advance in the state of the art. Other commercially viable systems tend to be much more narrow in their range of applicability, and degree to which they take the idea of ontology-driven information systems.

2.2. Visual Knowledge

The second company that recently announced some impressive capabilities is Visual Knowledge[9] [23,24]. This company has been around for many years developing an in-house capability for deploying model-based solutions to a variety of customers (including the power and financial sectors). The development environment is unique in that 1) is it fully driven by ontologies and metamodels, 2) the vast majority of a normal application is developed in a visual environment with no manual coding, 3) application development functionality is being exposed through a semantic wiki. Visual Knowledge was recently used to develop the computer game, Treasure Hunt[10]. After the interface joining Visual Knowledge to the 3D rendering engine (Croquet) was completed, only a few dozen lines of new code was written for the whole application. The code implemented a couple of functions that had not been needed before. All other functionality was created using a visual/graphical environment by combining and reusing capabilities that have been coded for prior applications. As in the prior example, development time was significantly reduced, compared to a traditional approach[11].

3. A Vision for the Future

3.1. How far have we come?

The emergence of the two companies with breakthrough capabilities is very exciting, and portends a bright future for ontology-driven information systems. Recall that in the 80s, the Knowledge Based Software Assistant [3,4] was an ambitious vision and project to create a “knowledge-based life-cycle paradigm”. The goals were to:

make software be:

1. “more responsive to changing requirements”
2. “more reliable”
3. “more revisable”

and to:

4. “provide machine mediated support”
5. “formalize the processes associated with software development”
6. “provide a corporate memory for [software] projects”

and to utilize:

7. “artificial intelligence”
8. “automatic programming”
9. “knowledge-based engineering” and
10. “software environment technology”

---

[9] http://www.visualknowledge.com
[10] http://www.treasurhuntthegame.com/
[11] Rob Bauman, personal communication, May 2007

In 1990, Gérard Comyn made some insightful predictions about the state of software engineering in 2010[12].

> "...Mastering such complexity on the software level will only be possible by means of a *very rigorous and disciplined intellectual penetration of the problem domains relying on powerful general tools for knowledge representation and manipulation.* I regard the following principles as decisive in this respect:
> 1. Founding any kind of programming activity on a *sound and reliable formal basis*.
> 2. Exploiting as much as possible the benefits of a *knowledge-based, declarative style of representing information*.
> 3. Investing considerable effort in much more *sophisticated methods of inference and retrieval* than known today despite of all the energy already spent we are still at a very rudimentary level of sophistication.
> 4. Designing the *procedural environment for manipulation of and interaction with knowledge* as soundly and rigorously as the declarative fundamentals of knowledge representation. [25]"

From what we have discussed so far, it is clear that we have made much progress on nearly all of the original KBSA goals. There is one major exception, and one possible minor one. The minor one is the apparent lack of any systematic and reliable knowledge-based [or any other kind of] approaches providing a corporate memory for software projects. Chaos seems to remain the state of this art – however, I have not tracked this area.

The major exception is that we still lack a robust and complete knowledge-based life-cycle paradigm that is widely agreed on and used, even by a substantial sub-community of practitioners. There have been many different approaches and methodologies for software engineering, and new ones are popping up fairly regularly. Arguably, the current trend towards ontology-driven information systems could be the basis for such a new lifecycle paradigm.

Comyn's predictions have also been right on the mark, in terms of where research and development efforts have been focused, and progress made. We do have powerful general tools for knowledge representation and inference. We are moving further towards declarative ways of representing information, and there is a strong focus on formal methods in much of today's cutting edge software engineering.

The two key ideas mentioned in the introduction, 1) raising the level of abstraction, and 2) increased use of formal structures and methods, have had a major impact on software engineering, and are particularly important for ontology-driven information systems. Wisely applied, these ideas can lead to many advantages:

Reduced conceptual gap: application developers can interact with the tools in a way that is closer to their thinking.

Reuse: abstract/general notions can be used to instantiate more concrete/specific notions, allowing more reuse.

Facilitate automation: formal structures are amenable to automated reasoning, reducing the load on the human.

Reduced development times: producing software artifacts that are closer to how we think, combined with reuse and automation enables applications to be developed more quickly.

Increased reliability: formal constructs with automation reduces human error

[12] Source: Michel Vanden Bossche, Mission Critical IT

Agility/Flexibility: Fully ontology-driven information systems are more flexible, because you can much more easily and reliably make changes in the model than in code.

Decreased maintenance costs: increased reliability and the use of automation to convert models to executable code reduces errors. A formal link between the models and the code makes software easier to comprehend and thus maintain.

While one can point to progress in all of these areas, in the big picture, the current state of software development is still well short of the ideal on all of these points. Even the most adanced systems have serious shortcomings: they may not scale well, or their generality may be limited (i.e. unsuitable for important kinds of applications). Nevertheless, we seem to be on the right road, moving in the right direction. The above benefits are almost always mentioned in reported implementations of model- and ontology-driven information systems. Where will we end up?

3.2. Where are we going? [13]

The current state of the art includes *no* fully functional computing infrastructure for general application development that has *any* of the following characteristics:

- fully visual
- fully agent-based.
- fully model driven
- fully ontology driven and declarative

Any existing systems of the above sorts, whether commercial or academic, while useful for their intended range of applications, were never designed for general application development. In the future, we will have application development platforms that have *all* of the above characteristics. In addition, applications will be ready to deliver either on or off the web with no additional work. Agents are a critical piece of infrastructure that we have not yet mentioned. They are needed to provide the necessary intelligence and flexibility for tomorrow's ODIS.

Future application development environments will be formal and declarative 'down to the metal'. Everything in an application (including models, ontologies, data, interfaces, algorithms and any other functionality) will be represented in a uniform way as one or more intelligent context-aware agents, semantically linked to other agents. All functionality will be captured in agents and will be driven from the models/ontologies. Applications will be manifest as giant networks of these "semantic agents". We will say that the models comprising the applications are *executing*, rather than *executable*. The models will not go through the usual steps of code generation, compiling and linking. Rather these models will be *already executing as they are being built.* This will be analogous to developing applications using today's interpreted programming languages. Furthermore, the 'models' will be executing efficiently, suitable for large scale commercial deployments, for virtually any kind of application.

Because the models will already be executing, there will be no need to generate code. This will dramatically speed up the program/test/debug cycle and unlock the potential of visual application development environments that has eluded us for decades. It circumvents the two main barriers to their widespread use: visual

[13] Most of the ideas in this section are from discussions with Conor Shankey.

environments are cumbersome to use for experienced developers and the generated code runs too slowly. In the future, virtually everyone will be able to use visual application development environments, not just those who aspire to be programmers. Visual programming interfaces and debuggers will be the norm for virtually all phases of software engineering. Collectively these features will significantly shorten the application development cycle, compared to traditional approaches. As noted above we are already seeing early indications of this happening with Mission Critical IT, and Visual Knowledge.

These are major advances from the perspective of general computing. We also look forward to substantial advances from the perspective of semantic computing. The known state of the art for semantic technology today includes *no*

1. seamlessly integrated infrastructure for semantic application development
2. robust solution to the problem of ontology versioning and evolution
3. ontology based data store with a full complement of reasoning and rules functionality that will scale to large commercial applications.

Point 1 is a major barrier to use of semantic technology today. Anyone who is convinced that they need an ontology for their software application soon experiences the unfolding nightmare of deciding how the application should be architected to make use of the ontology. The state of the art often seems to be: nothing integrates with anything, so you spend most of your time choosing among many different pieces and then figuring how to put them together into a working system. Even in cases where the ontology plays a prominent role in the end application (which is not the norm) the amount of time spent on developing and using the ontology can be dwarfed by the architecting and gluing work that is necessary. This was the subject of a panel at the 2008 Semantic Technology Conference [26]. Fortunately this is beginning to change. One important development is the growing use of REST [27]. People are starting to integrate the ontology development tools with the software delivery components. To help keep track of what is out there, a Semantic Web Tools Wiki has been created[14].

In the future, fully integrated semantics-based application development environments will speed up development time by avoiding the need to decide which tools to use, and how to hook them together. Under a single 'hood' developers will have ontology development and reasoning, GUI development, all application logic and functionality all stored in a transactional database with versioning. This leads us to the next item.

Versioning (point 2) is a problem of growing importance, and no-one has published any solutions. Tom Atwood highlighted the importance of versioning in an OWL-based object store that he was developing in May 2007 [28]. He outlined how agents need to keep track of the interdependencies, and it was to be a feature of a future object store product. It is hard enough when you just have UML models or ontologies that you need to keep track of. It is exponentially more challenging to deal with versioning when every element of functionality is driven from the ontology. If an ontology element changes, the application can break. Conor Shankey of Visual Knowledge believes that "ontology-driven software applications will never take off unless versioning is handled properly[15]."

[14] http://esw.w3.org/topic/SemanticWebTools.
[15] Conor Shankey, personal communication, July 2008

In the future, we will have fully transactional data stores for each model element, (e.g. a semantic agent). To do that effectively, every agent will know and keep track of its inter-dependencies with other agents. Every version of every agent will have to know when there are new versions of agents that it depends on, and act accordingly, to maintain system integrity. Because the whole application will consist of agents, and every agent will be version-controlled, there will be very fine granularity version control and change management. The unit of versioning will not be flat text files containing code, but richly structured semantic agents that comprise the application.

Point 3 is another major barrier of using semantic technology today. Triple stores don't scale to the level of relational databases, and more general knowledge stores with more sophisticated reasoning are even more challenging. Today, you either get performance and scale with limited reasoning, or vice versa. In the future, a new breed of data store will emerge that provides functionality of all kinds, not just a limited set of inferences. The scale problem will be solved by figuring out how to make use of the growing number of cores in our CPUs – i.e. algorithms will be parallelized.

4. Summary and Conclusion

We believe that the idea of ontology-driven information systems is in the early stages of becoming a practical reality, and that before too long, it will become mainstream. They have roots that go back to the beginning of software engineering. We traced some of these roots and identified a number of key ideas that contributed to where we are today. Two prominent ideas are raising the level of abstraction, and using formal representations and methods. Critically important technologies include domain modeling, knowledge representation and reasoning, automatic programming, executable specifications, model-driven software development and semantic web technologies.

We drew a distinction between ontology-driven software engineering, which may use ontologies in the process of building an application, but that may not use the ontology, vs. ontology-driven information systems (ODIS), where the ontology also plays a significant role in the end application. We summarized some of the benefits of ODIS, chief of which are allowing developers to interact with tools that are closer to their way of thinking, reuse, automated reasoning, flexible and agile development, faster development times, increased reliability and decreased maintenance costs.

We looked at a variety of systems over the past few decades, and reported on two rather surprising companies that have been beavering away for years, and recently announced some apparently breakthrough capabilities. Finally, we peered through a crystal ball and imagined that the future might look like this:

- Every element of an application (including models, ontologies, data, interfaces, algorithms and any other functionality) is represented in a uniform way as one or more intelligent context-aware agents.
- Applications are manifest as giant networks of semantic agents that represent the models.
- The semantic agents are fully version-controlled and stored in a transactional database.
- There is no code generation step, the models are already executing as they are being built.

- Virtually no lines of code are hand-written for normal application development.
- Virtually all functionality is specified in a visual/graphical environment by creating and linking various kinds of semantic agents with pre-coded functionality (similar to Figure 1).
- No computer science background is required to develop applications.

This is truly a world where application development opens up to the masses. The distinction between user and developer will blur, as will the distinction between conventional computing and semantic computing. All information systems will be ontology driven.

Acknowledgments

I thank Michael Gruninger and the rest of the FOIS08 organizing committee for the opportunity to write this paper and give an invited talk. I am indebted to Conor Shankey for introducing me to Visual Knowledge and for providing most of the ideas in the future vision section. I am grateful to Michel Vanden Bossche for introducing me to his breakthrough system, ODASE. Dean Allemagne explained how TopQuadrant tools are used for ontology-driven information systems. I am grateful to my PhD supervisor, Alan Bundy who supported me when I most needed it. Finally, I am grateful to the dozens of wonderful colleagues and friends that I have worked with over the years – you know who you are.

References

[1] B. Hailpern and P. Tarr, *Model-driven development: The good, the bad, and the ugly*. IBM Systems Journal, 45(3), 2006

[2] C. Rich and R. C. Waters, Eds. 1986 *Readings in Artificial Intelligence and Software Engineering*. Morgan Kaufmann Publishers Inc.

[3] Green, C., Luckham, D., Balzer, R., Cheatham, T., and Rich, C. 1986. Report on a knowledge-based software assistant. In *Readings in Artificial intelligence and Software Engineering*, C. Rich and R. C. Waters, Eds. Morgan Kaufmann Publishers, San Francisco, CA, 377-428.

[4] KBSA (Knowledge-Based Software Assistant) Concept Demo final report (1992) by M DeBellis, K Miriyala, S Bhat, B Sasso, O Rambow.

[5] Domain Modeling - Evolving Research, Iscoe, N. Knowledge-Based Software Engineering Conference, 1991. Proceedings., 6th Annual Volume , Issue , 22-25 Sep 1991 Page(s):234 - 236

[6] Premkumar T. Devanbu, Ronald J. Brachman, Peter G. Selfridge: LaSSIE: A Knowledge-Based Software Information System. Commun. ACM 34(5): 34-49 (1991)

[7] T. Emerson and M. Burstein, "Development of a Constraint-Based Airlift Scheduler by Program Synthesis from Formal Specifications," *Proc. 1999 Conf. Automated Software Eng.,* IEEE CS Press, Los Alamitos, Calif., 1999.

[8] D. Smith, KIDS: a semiautomatic program development system *IEEE Transactions on Software Engineering*, Vol. 16, pp 1024-1043, 1990

[9] Mike Uschold, Mike Healy, Keith Williamson, Peter Clark, Steven Woods; Ontology Reuse and Application, in *Proceedings of the 1st International Conference on Formal Ontology in Information Systems* (FOIS'98) 1998

[10] Borgida, A., Greenspan, S., and Mylopoulos, J. 1986. Knowledge representations as the basis for requirements specifications. In *Readings in Artificial intelligence and Software Engineering*, C. Rich and R. C. Waters, Eds. Morgan Kaufmann Publishers, San Francisco, CA, 561-570.

[11] G. Schreiber, B. Wielinga, and J. Breuker (eds.): *KADS. A Principled Approach to Knowledge-Based System Development, Knowledge-Based* Systems, vol 11, Academic Press, London, 1993.

[12] M. Uschold, *The Use of Typed Lambda Calculus for Comprehension and Construction of Simulation Models in the Domain of Ecology*, PhD Thesis, The University of Edinburgh, 1990

[13] D Robertson, A Bundy, R Muetzelfeldt, M Haggith, *Eco-Logic: Logic-based Approaches to Ecological Modeling* MIT Press Cambridge MA, 1991

[14] M. Lowry, A. Philpot, T. Pressburger, and I. Underwood. *A formal approach to domain-oriented software design environments*. In Proc. 9th Knowledge-Based Software Engineering Conf., pp. 48--57, 1994.

[15] Harald Störrle, Jan Hendrik Hausmann: *Towards a Formal Semantics of UML 2.0 Activities*. Software Engineering 2005: 117-128

[16] T. Ilube, *The Semantic Web, Social Graphs and Social Verification*, presented at the 2008 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/session/1068/

[17] D.Beckett, *Yahoo! Semantic Web In Production*, Presented at the 2008 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/session/733/

[18] OMG, *Ontology Definition Metamodel*, http://www.omg.org/docs/ad/05-01-01.pdf

[19] H. Knublauch, *From Ontology Design to Deployment: Semantic Application Development with TopBraid,* presented at the 2007 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/2007/sessions/l5.html

[20] A. Crapo & M. Garbiras, *Stop Programming, Start Modeling: Developing Work-Centered Semantic Applications* http://www.semantic-conference.com/2007/sessions/q3.html

[21] M. Vanden Bossche, *Ontology-Driven Software Development for Business-Critical Enterprise Applications*, presented at the 2008 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/session/586/

[22] Michel Vanden Bossche, Peter Ross, Ian MacLarty, Bert Van Nuffelen, Nikolay Pelov, Ontology Driven Software Engineering for Real Life Applications, *, 3rd International Workshop on Semantic Web Enabled Software Engineering (SWESE 2007)* Innsbruck, Austria, 2007.

[23] C. Shankey, *Building Semantic Applications in a Semantic Wiki*, presented at the 2007 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/2007/sessions/a5.html

[24] R. Bauman, J. Huckenpahler & R. Centers, *SEX: Semantic user Experience,* presented at the 2007 Semantic Technology Conference, San Jose, CA.

[25] G. Comyn *Programming in 2010 – A scientific and industrial challenge*, Computational Logic, J.W. Lloyd (Ed.), November 1990, Springer-Verlag

[26] H. Story, D. Allemang, J. Aasman, E. Miller, & J. Wester, *Developing Semantic Web Applications: Current Tools, Best Practices and Future Directions* Panel Discussion at the 2008 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/session/482/

[27] Battle, R. and Benson, E. 2008. Bridging the semantic Web and Web 2.0 with Representational State Transfer (REST). *Web Semant.* 6, 1 (Feb. 2008), 61-69.

[28] T. Atwood, OWL DBMS, presented at the 2007 Semantic Technology Conference, San Jose, CA. http://www.semantic-conference.com/2007/sessions/k5.html

1. Foundations

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-21

Ontology (Science)

Barry SMITH[1]
Department of Philosophy, Center of Excellence in Bioinformatics and Life Sciences, and National Center for Biomedical Ontology, University at Buffalo, USA

Abstract. Increasingly, in data-intensive areas of the life sciences, experimental results are being described in algorithmically useful ways with the help of ontologies. Such ontologies are authored and maintained by scientists to support the retrieval, integration and analysis of their data. The proposition to be defended here is that ontologies of this type – the Gene Ontology (GO) being the most conspicuous example – are a *part of science*. Initial evidence for the truth of this proposition (which some will find self-evident) is the increasing recognition of the importance of empirically-based methods of evaluation to the ontology development work being undertaken in support of scientific research. Ontologies created by scientists must, of course, be associated with implementations satisfying the requirements of software engineering. But the ontologies are not themselves engineering artifacts, and to conceive them as such brings grievous consequences. Rather, ontologies such as the GO are in different respects comparable to scientific theories, to scientific databases, and to scientific journal publications. Such a view implies a new conception of what is involved in the authoring, maintenance and application of ontologies in scientific contexts, and therewith also a new approach to the evaluation of ontologies and to the training of ontologists.

Keywords: scientific method, expert peer review, ontology engineering, biomedical informatics, Gene Ontology, OBO Foundry

1 Introduction

For some time now the Gene Ontology (GO) [1] has enjoyed the status of a *de facto* standard vocabulary for the annotation of experimental data pertaining to the attributes of gene products. The GO has been widely applied to data drawn from experiments involving organisms and biological processes of many different types. It has also been subject to a series of logical reforms, which have enhanced the degree to which it can be exploited for algorithmic purposes. The GO is now routinely used in gene expression analyses of a wide range of biological phenomena, including phenomena relevant to our understanding of human health and disease.

The thesis to be defended here is that *the GO and its sister ontologies are a part of science*. This means (i) that these ontologies themselves are properly to be understood as results of scientific activity, analogous to journal publications (in some ways also to textbooks and databases), and (ii) that the processes involved in authoring, maintaining and evaluating them are a part and parcel of the activity of science.

In what follows I shall draw out some implications of this thesis, focusing my attentions on the GO and on the other biomedical ontologies participating in the OBO

[1] Corresponding author: Barry Smith, Department of Philosophy, 130 Park Hall, University at Buffalo, Buffalo, NY 14260, USA; E-mail: phismith@buffalo.edu.

Foundry initiative [2,3]. These provide the most conspicuous examples of ontology (science) in the sense here intended. The views expressed will appear to many to be self-evident; in their detail, however, they are still exploratory in nature (and thus they do not represent any settled policy of the Foundry initiative).

2 The OBO Foundry

The Open Biomedical Ontologies (OBO) repository was created in 2001 by Michael Ashburner and Suzanna Lewis as a means of providing convenient access to the GO and its sister ontologies at a time when resources such as the NCBO BioPortal [4,5] did not yet exist. The OBO Foundry was initiated by Ashburner, Lewis and Smith in 2005 as a collaborative experiment designed to enhance the quality and interoperability of life science ontologies from the point of view of both biological content and logical structure [3]. The Foundry initiative is based on the voluntary acceptance by its participants of an evolving set of principles designed to maximize the degree to which ontologies can support the needs of working scientists. The developers of nearly all of the ontologies within the OBO repository have committed themselves to participate in this initiative, which has spawned also the establishment of a number of new ontology projects within the Foundry framework.

2.1. The OBO Foundry Principles

The principles of the Foundry can be summarized, in their current version, as follows.

First, are syntactic principles to the effect that an ontology submitted to the Foundry must employ one or another common shared syntax, possess a unique identifier space, and have procedures for identifying distinct successive versions.

Second, are principles involving definitions: the Foundry requires that textual definitions (and, by degrees, equivalent formal definitions) be provided for all terms; that terms and definitions be composed using the methodology of cross-products (see below); and that ontologies use relations that are unambiguously defined according to the pattern set forth in the OBO Relation Ontology (RO) [6].

Third, ontologies are required to be open (available to be used by all without any constraint), to have a clearly specified and clearly delineated content, to have a plurality of independent users, and to be subject to a collaborative development process involving the developers of Foundry ontologies covering neighboring domains.

Finally, the Foundry embraces a principle of orthogonality. This asserts that for each domain there should be convergence upon a single ontology that is recommended for use by those who wish to become involved with the Foundry initiative. If an ontology is submitted which overlaps substantially with an existing Foundry ontology, then the two sets of developers will be invited to collaborate in the creation of a common, improved resource, through application of the sorts of evidence-based strategies applied in other parts of science to resolve the problems that arise where alternative theories of a single phenomenon are advanced by competing groups.

2.2. The Problem of Data Silos

The primary rationale for our insistence upon the principle of orthogonality is that it offers a potential solution to a pressing problem facing researchers in information-driven areas of biology and biomedicine, namely the problem of the siloing of data. In part in response to NIH mandates, many such researchers have recognized the need to find ways to present their data in a form that will make them more easily combinable and also accessible to wider communities of researchers. Ideally, this would be achieved by constraining terminologies and data schemes so that they converge on commonly accepted standards [7]. Unfortunately, however, there is normally still no clear answer as to what, in any given case, should serve as basis for such constraint. Many researchers therefore find that they have little choice but to create their own local schemes for description of their data.

The OBO Foundry proposes a solution to this problem that is incremental, modular, empirically based, and incorporates a strategy for motivating potential developers and users. Briefly, the ontologies in the Foundry are being built as orthogonal, interoperable modules which together form an incrementally evolving network. Scientists are motivated to commit themselves to developing ontologies falling within their domains of expertise because they themselves will need to use these ontologies in their own work in the future. Users are motivated by the assurance that the ontologies they adopt from the Foundry will be maintained in the future by scientists with the relevant sorts of expertise. Both arms of this strategy for motivation can be realized effectively only against a background in which the principle of orthogonality is accepted by all involved.

2.3. Benefits of Orthogonality

Further benefits brought by acceptance of the principle of orthogonality include:

First, it helps those new to ontology who need to know where to look in finding ontology resources relating to their subject-matter for which they can have reasonable assurance that they have been validated and will be used and maintained in consistent fashion by subject-matter experts; that they will work well with other established ontologies; and that the expertise acquired in adapting these resources to specific local needs will potentially be of general and lasting utility.

Second, it obviates the need for ‘mappings’ between ontologies, which have proved not only difficult (and expensive) to create and use, but also error-prone and hard to keep up-to-date when mapped ontologies change.

Third, it ensures the mutual consistency of ontologies, and thereby also the additivity of the annotations created with their aid by different groups of annotators describing common bodies of data. In this way, orthogonality contributes to the cumulativity of science and allows new forms of unmanaged collaboration.

Fourth, it rules out the sorts of simplification and partiality which may be acceptable under more pluralistic regimes, and thereby brings an obligation on the part of ontology developers to commit to strive for scientific accuracy and domain-completeness in their work.

Fifth, orthogonality provides support for the Foundry’s strategy of utilizing *cross-products in composing terms and definitions* [8,9]. This strategy is designed (i) to reduce the degree of arbitrariness typically involved in term composition in complex ontologies, and (ii) to ensure that Foundry ontologies are developed in tandem in such a way as to constitute a progressively more well-integrated modular network. The idea is that, where ontologies need to include complex representations (for example of: *effects of viral infection on cell function in shrimp*), these should be built up compositionally out of component representations (here: *virus*, *infection*, *cell*, *function, shrimp*) already defined within other, more basic feeder ontologies. By enforcing orthogonality (and the use of relations derived from the RO for term combination), we can go far towards ensuring a unique choice for such composition that serves at the same time to bind the more specialized ontologies to the benchmark feeder ontologies from which constituent terms are drawn.

Finally, orthogonality helps to eliminate redundancy and it serves the division of ontological labor in ontology development work. It allows communities from different biology disciplines to address the tasks of ontology building to different levels of detail and under different timetables. It makes possible the establishment of clear lines of authority, whereby experts in each domain are able to take responsibility for creating and maintaining a single, high-quality ontology that is tailored for that domain, adjustments to which are then passed on to those other ontologies which have used its resources in composing terms and definitions via cross-products, thereby bringing further benefits of cross-ontology synchronization.

2.4. What Orthogonality is Not

When ontologies are seen as analogous to scientific theories [10], then orthogonality is a principle which arises naturally. This is because it is a pillar of the scientific method that scientists should strive always to seek out and resolve conflicts between competing theories. This is why scientists have over centuries made a huge investment in intra- and interdisciplinary synchronization, illustrated for example in the use of common standard systems of measurement units. In order to allow detection of conflicts and testing of proposed resolutions, scientists must work as far as possible within a single universe of collaborators and scientific results must be (by definition) available to all.

Where, however, ontologies are conceived as engineering artifacts, then orthogonality is neither practically achievable nor, from the perspective of ontology creators, intrinsically desirable. Here there prevail quite different disciplinary mores, in some respects comparable to those obtaining in the world of commercial enterprise.

When the orthogonality principle has been subjected to criticism in engineering circles, this has sometimes been because the principle itself has been misinterpreted as resting on a view to the effect that there can be only one correct way to represent the entities in each domain of life science research. In fact, however, all of those involved in the Foundry initiative are aware that the Foundry ontologies represent nothing more than initial attempts to address difficult problems, and that they rest on hypotheses that are always subject to further revision and supplementation. We are also fully aware that, in this as in all other domains, scientific advance rests on the to-and-fro of criticism between the advocates of competing hypotheses. We thus see considerable benefit in the development of alternative sets of ontologies by other groups, even if at the same time we warn of a shared need for strategies to counter potential dangers of silo formation. We also envisage scenarios under which externally developed ontologies would be incorporated into the Foundry because they are of superior quality to those which they would then replace.

2.5. The Strategy of Reference Ontologies

Another criticism raised by engineers against the orthogonality principle is that it will cause problems for ontology users who require their own purpose-built ontologies to address specific needs. In fact, however, the Foundry offers a strategy to address such special purposes in ways that do not contribute to the formation of silos.

This strategy rests on a view of ontologies used in science as being divided into two kinds. On the one hand are the so-called *reference ontologies* [11], arranged orthogonally within the Foundry itself. On the other hand is a larger edifice of different types of *application ontologies* constructed on this foundation, the whole being connected together, prospectively, through application of the methodology of cross-products, and employing strategies for networking of the sort now being tested within the framework of the Semantic Web [12].

Reference ontologies are analogous, in different ways, to both scientific theories and textbooks. Each has its own subject-matter, which consists of the entities in reality addressed by the corresponding branch of biomedical science. Each seeks to maximize descriptive adequacy to this subject-matter by being built out of representations which are correct when viewed in light of our best current scientific understanding.

Application ontologies, in contrast, are comparable to engineering artifacts. They are constructed for specific practical purposes such as management of data in a multi-institution clinical trial [13]. The problems arise because such artifacts are still normally built afresh for each new trial or study. What results may then serve immediate needs perfectly well, but it creates snowballing obstacles as researchers need to reuse the associated data for other purposes – for example to share them with colleagues working on cognate phenomena, or to perform meta-analyses.

Our proposal is that application ontologies should as far as possible be developed from the start in alignment with a common set of reference ontologies such as are provided by the OBO Foundry [14]. This is achieved by employing terms residing in these reference ontologies (thus preserving their existing identifiers) and using them to build new terms via composition. Requests should be submitted to the relevant Foundry ontologies where needed terms are not available. Only in this way, we believe, can the tendency towards silo formation be counteracted and the associated obstacles to the retrieval, reuse and integration of data thereby be prospectively reduced.

A final criticism of the orthogonality principle, made by Musen, turns on the question of whether it is in fact possible to find, for each domain of reality, a single perspective that 'is deserving of being canonized as a reference ontology'. From the standpoint of the 'engineering faction', as Musen sees it:

> [i]t would not make sense to talk about, say, an overarching ontology for biomedical investigations when, for example, the distinctions required to describe an experiment for publication are different from those required to describe an experiment to assist scientists in the execution of the experiment, which are different from those required to estimate the cost of the experiment to a sponsor, which are different again from those required to determine whether the experiment is ethical. (Personal communication)

This passage is interesting not least because there does in fact exist an Ontology for Biomedical Investigations (OBI) [15], which is one of the most successful new ontologies being created *ab initio* in accordance with OBO Foundry principles. Its goal is to provide controlled, structured representations for the design, protocols, instrumentation, materials, data and data analysis in biological and biomedical investigations of all types. This goal has been embraced by some two dozen communities representing different domains of high-throughput experimentation, ranging from flow cytometry to in situ hybridization and immunohistochemistry. We are learning from the experience of OBI development how difficult it is to do serious ontology work in a large and heterogeneous domain where such work necessarily involves the contributions of multiple groups of specialists with different sorts of domain expertise. To make such contributions work effectively within a single framework requires a complex process of a sort which mimics within a single ontology the process for cross-ontology coordination being applied by the Foundry as a whole.

OBI is, like every other ontology, a work in progress; it serves none of the context-specific purposes listed by Musen perfectly; but it serves all of them to some degree. OBI has been recognized as a reference ontology by the OBO Foundry because all of the communities involved recognize the advantages brought by creating a single ontology that can serve as a stable attractor for the many communities who need a framework that can already serve annotation of data describing how experimental results were obtained while being subjected to incremental improvement and expansion.

3 Science is Cumulative

Central to ontology (science) is the requirement that ontologies, like scientific theories, should be tested empirically. This requirement is realized in the context of the OBO Foundry through the work of the many biologists who contribute to the maintenance of its member ontologies from day to day [16]. This they do by aggressively using these ontologies in the annotation of new experimental results, in a development that has given rise to a new scientific profession of literature curator [17]. Because new results reported in the journal literature need to be annotated using corresponding reference ontologies, this generates new content for and corrections to these ontologies, thereby providing enhanced resources for literature curation in the future. This virtuous cycle is exemplified already in the work of a plurality of life science research communities, and the methodology has been thoroughly tested especially by the model organism research communities within the Gene Ontology Consortium [18].

Researchers in information-driven disciplines of contemporary biology are hereby realizing in a new form a pattern that has been characteristic of empirical science since its inception. Simplifying greatly, we can say that each branch of science is marked by the existence of a consensus core of established results surrounded by a changing penumbra of hypotheses that are to different degrees marked as problematic. This consensus core was earlier documented in textbooks. Increasingly, it will be documented also in ontological form.

Empirical science is *cumulative* in the sense that the consensus core of each discipline grows by absorbing hypotheses which began as problematic but have withstood attempts to refute them empirically. The process of cumulation is, of course, marked at every stage by setbacks and false starts and by the competition between theories referred to already above. Except in those rare periods in which sciences are undergoing revolutionary change, however – for example the change from Newtonian physics to special relativity – these factors will not be sufficient to dislodge the broad mass of propositions making up the consensus core.

The goal of the OBO Foundry can now be characterized as follows. *First*, and as it were on the object level, it is to provide a coherent and interoperable suite of controlled structured representations of the entities and relations described at any given stage in the consensus cores of each of the biological sciences. This framework is designed to be maximally stable, in order to provide a basis for the progressive cumulation of the scientific data described in its terms. At the same time the framework must be flexible enough to accommodate change as new experiments are performed, new results discovered, and old hypotheses refuted. *Second*, and on the meta-level, it is to establish ontology development itself as being, like statistics, a recognized part of the scientific enterprise. This brings the need to determine, incrementally and empirically, the consensus core of ontology (science), and to train a community of ontology experts who will be in a position to apply and to extend this core in their scientific work. The set of Foundry principles represents an initial glimpse of what this consensus core might contain. The overarching goal – whose significance we are only now beginning to understand – is to serve the ends of cumulativity (which means: preventing silos) in an era where the advance of scientific research is increasingly being mediated by computers, and thus increasingly subject to the influence of engineers whose incentives have sometimes been at odds with those of working scientists.

4 Ontology and Expert Peer Review

4.1. The Foundry Strategy

To become established as a properly scientific activity, ontology development must be subject to processes of evaluation of the same sort that are practiced in other parts of science. In this light, we believe that benefits can be gained from a view of ontologies as being, in crucial ways, analogous to scientific publications, and thus as subject to the discipline of *expert peer review*. The OBO Foundry has accordingly been experimenting with procedures designed to pave the way for the incorporation of the methodology of expert peer review into ontology development practice.

Progressively, each ontology submitted to the Foundry will be subject to review (1) by *Coordinating Editors* whose primary responsibility is that of harmonizing interactions (of content and of logic) between Foundry ontology development projects in neighboring domains;[2] and (2) by *Associate Editors* selected by those involved in the development and maintenance of the ontologies in the OBO repository, whose task is to provide input from these separate ontology developer communities.

The ontology peer review process will involve also ad hoc discipline-based reviewers, who will be selected on the basis of their specific scientific expertise, and who will evaluate ontologies not as computational artifacts but as representations of scientific domains. To this end it is important that there are ways to translate Foundry ontologies not only into multiple different computational formats [19] but also into something close to English [20]. In this way, ontologies such as the GO exist, and serve as objects for evaluation, in forms that are independent of specific computational implementations. In this respect, too, they are like scientific theories.

[2] Currently, the Foundry Coordinating Editors are, in addition to Ashburner, Lewis and Smith, also Christopher Mungall (a leader in the GO and model organism database communities), Alan Ruttenberg (principal scientist of Science Commons and Chair of the OWL Working Group), and Richard Scheuermann (principal investigator of the ImmPort Immunology Database and Analysis Portal and of the BioHealthBase Bioinformatics Resource Center projects).

4.2. Advantages of Expert Peer Review

As ontology engineers have criticized the principle of orthogonality, so also they have resisted the application to ontologies of the methodology of expert peer review [21]. It will thus be worth our while to summarize briefly some of the benefits that peer review has brought to the practice of science, benefits which have led to its adoption by scientific publishers, universities, and research and funding agencies in their quest for scientific quality.

Expert peer review provides an impetus to the improvement of scientific knowledge over time, as authors compete for scarce funding or for occupation of prestigious journal space [22,23]. It not only improves the quality of published papers through the *ex post* revisions fostered by reviewer comments, but also helps to discipline scientific communication as a result of the fact that authors are aware *ex ante* that their results need to be formulated in such a way that they will be intelligible to unknown, critical peers with powers of sanction.

Because peer review introduces an element of expert judgment independent of authors and editors, this lends it some of the functionality of an audit process. It serves as a filter to detect duplication, fraud or distorted information, and hence it is valued by regulatory agencies, which see it as providing a partial validation of scientific results.

These filters are of course not perfect. Thus far, however, no other vetting device has been offered that would do a better job. Moreover, some of the proposed alternatives have been shown to be marked by even more severe failings [24].

Filtering based on the judgment of experts brings benefits also to readers, since they need only absorb and collate vetted manuscripts, as opposed to all the manuscripts submitted to the relevant journals and to journal-like respositories. As Bug points out, such filtering promises to be especially useful in the field of biomedical ontology:

> Until there is a reliable vetting procedure, we cannot expect to re-use and extend existing ontologies effectively or with confidence for the purpose of bringing like data together in novel ways from across the biomedical data diaspora. Without vetting, we cannot expect to provide other developers with clear advice on what are the reliable ontological shoulders to build on. [25]

The fact that we currently often have multiple ontologies covering single domains at the same scope and level of granularity generates problems:

> how [can] a bioinformatics application developer determine which one to use? Even more importantly, if users pick at random from amongst the two or more ontologies covering the same domain, who will maintain the maps and software required to make deductions or inferences across the annotated data repositories which use these different ontologies to cover the same domain? [25]

4.3. Creating Incentives for Investment of Effort in Ontology Development

The need for ontology resources on the part of scientific and clinical researchers is ever increasing. And while attempts are being made to meet this need through automatic generation of ontologies, there is an increasing recognition of the fact that successful ontology development will require a considerable contribution from human experts. Typically, however, ontology work – like its counterpart in the field of database development – brings rewards incommensurate with the effort that must be invested to yield seriously useful results. One set of incentives being brought into play within the

Foundry to address this problem rests on motivating factors relating to the exercise of *influence*. Briefly, individuals will be motivated to commit themselves to investment in ensuring the adequacy of a given resource if they know (a) that they themselves will be using that resource in the future, and (b) that they will gain benefits if that same resource is used also by watchful colleagues, some of whom will then embrace a similar commitment. The importance of this sort of motivation has been demonstrated already in open source endeavors in the field of software standards. Such endeavors are, as documented by Weber [26], subject to an ever-present danger of forking. This danger must be constantly counteracted if incentivizes for involvement are to be maintained. Weber shows that the open source process is most likely to achieve this end when addressing tasks that have the following characteristics:

1. Disaggregated contributions can be derived from knowledge that is … not proprietary.
2. The product is perceived as important and valuable to a critical mass of users.
3. The product benefits from widespread peer attention and review, and can improve through creative challenge and error correction.
4. There are strong positive network effects to use of the product.
5. An individual or a small group can take the lead and generate a substantive core that promises to evolve into something truly useful.
6. A voluntary community of iterated interaction can develop around the process of building the product.

The likelihood of success in realizing these characteristics seems to be highest where the community effort is organized on the basis of a pyramidal authority structure resting to a high degree on delegation. In each successive phase of the work, positions of authority are assigned by those already holding such positions to individuals who have demonstrated both commitment to the effort and relevant expertise. The Foundry is an attempt to realize a structure of this sort within the life science ontology domain.

4.4. The Strategy of Expert Peer Review of Ontologies

A second set of incentives is provided by bringing about a situation in which ontology developers would receive career-related credit by having their ontologies count as analogues of peer-reviewed scientific journal publications. This would allow citations of ontologies to be measured in the same way as are citations of other sorts. It would allow ontology reviewers to gain credit analogous to that currently awarded for membership in journal editorial boards. It would enable also crucial elements of a scientific career path for ontologists, given that career advance in academic institutions rests on the existence of peer review-based mechanisms for evaluation of scientific competence which, in the ontology domain, have hitherto been lacking.

As in the case of traditional journal submissions, so also in the case of ontologies, the peer review strategy which the OBO Foundry is pilot testing will be an iterative process, with recommendations for revision being addressed in successive versions of the ontology until a stage is reached where it is deemed suitable for publication.

One obvious problem for such a strategy turns on the fact that ontologies, in contrast to traditional journal publications, are subject to continuous update. This problem has however been addressed already by those publishers who have brought scientific databases within a peer review framework. The Nature Publishing Group (NPG), for example, has addressed the issue of data curation speed in relation to its Signaling Gateway[27] by employing wiki tools to allow responses submitted by users to supplement peer reviewed data. NPG is however careful to insist that, in experiments such as this, 'It must be made clear to the user … which information has been peer reviewed and which has not.' [28]

A further problem for ontology peer review turns on the special role of users. As Musen puts it, while the job of reviewing journal articles is performed 'rather well by scientists who are experts in the field and who can understand the work … described', the key question of whether an ontology makes the right distinctions about its domain

> can be answered only by application of the ontology to some set of real-world problems and discovering where things break down. The people best suited for making the kinds of assessment that are needed are not necessarily the best experts in the field, but the mid-level practitioners who actually do the work. Any effective system of peer review has got to capture the opinions of ontology users, and not just those of renowned subject-matter experts or of curators. [29]

These remarks are well taken. But we believe that they do not imply that there is some problem with the methodology of peer review as the Foundry conceives it. Expert users of ontologies are already included among the Foundry reviewers, and the OBO Consortium has established strategies for taking account of user input through a heavily utilized system of open access sourceforge trackers and email forums [30].

5 Ontology Evaluation via Democratic Ranking

5.1. A Strategy for Community Based Review of Ontologies

Scientific ontologies are often highly complex. They are subject to a high velocity of change, not only in virtue of scientific advance, but also because the associated computational technologies are themselves rapidly evolving. As new applications for ontology-based technology are identified, so new ontologies are being created, bringing problems of choice and validation to potential users. To address these problems the NCBO [31] and the Networked Ontology Consortium (NeOn) [32] are carrying out experimental tests of software-based strategies to support ontology assessment.

In essence, these strategies address goals addressed also by the Foundry editorial process. Both seek a particular kind of quality assurance of ontologies. Both rely on human reviews of ontologies. On the approaches advanced by NCBO and NeOn, however, the community of those involved in providing reviews is (potentially at least [33]) larger than on the more selective approach favored by the Foundry. This is because one key element of the NCBO and NeOn approaches is inspired by the 'open' Web-based systems for the rating of consumer goods developed by organizations such as amazon.com or eBay [34,35]. The resultant strategy for 'democratic ranking' is described by Lewen as one according to which 'everyone can write reviews about the ontologies', and 'some of the reviewers can (and should) be … experts'.

> Not only does this approach scale (everybody can review), it is also very personalizable. It is up to the user to decide whether she values the opinion of a 'mere ontology user' more than the opinion of an 'ontology expert'. [36]

5.2. Problems with Democratic Ranking

On the democratic ranking approach 'trust scores' will be dynamically assigned to the authors of reviews by a larger community of users, on the basis of numerical responses to the question: *was this review helpful to you?* [37] It is then assumed, reasonably, that users will be drawn to the reviews of those who have received trust scores which are higher than the average. But will the latter also be those who have the necessary expertise, integrity, diligence, free time, and frankness to do their reviewing job properly? The empirical data that would enable us to answer this question are not as yet available. Already, however, there are reasons to question whether experts in scientific disciplines would devote their time to making contributions to an open ranking system of this sort:

1. The very idea that scientifically relevant decisions can be made on the basis of democratic vote will seem to them absurd. The evidence that this is so is easily acquired by talking to scientists. Their instinctive rejection of the idea turns on the fact that scientific decisions – as contrasted to decisions concerning, for example, choice of consumer goods – are tied logically to myriad further decisions made by other scientists, sometimes over long spans of time, on the basis of bodies of experimental evidence that are often too complex to be comprehended by any single person. It is for this reason that the processes of scientific decision-making are so involved, and why they have led to the evolution of formal and informal institutions which may seem cumbersome and antiquated to outsiders.

2. One such institution is the practice of reviewer confidentially, which brings the benefit of enhancing the ability of reviewers to express opinions frankly. On the strategies for 'open' democratic ranking advanced by NCBO and NeOn, this benefit will be lost. Certainly, the policy of open review, too, can bring benefits of its own: some may be motivated to write more thorough reviews, and perhaps thereby gain credit and acknowledgement. The prognosis for the success of such a policy is however poor, not least because the potential hazards for authors of negative reviews (ranging from career problems to lawsuits) will lead many potential reviewers to refrain from participating [33].

3. Career-related credit would seem not to accrue under the envisaged open ranking systems. Academic institutions do not promote on the basis of rankings assigned on the Web by non-experts.

4. A strategy centered on a user-based ranking of reviewers will it seems fall short of realizing one vital purpose inherent to the methodology of expert peer review, namely that of reducing search and decision costs on the part of those involved in research. For the reasons given by Bug in the passages quoted above, we should avoid the temptation to place these costs once more into the hands of researchers for the sake of an 'openness' whose benefits in the scientific context are as yet unproven.

5. Under the mix-and-match selection procedures envisaged by NCBO and NeOn, the views of experts will not only be subject to a filtering process involving non-experts, but also potentially diluted through admixture with non-expert views. This will at least diminish those sorts of motivation for serious investment of time in ontology development that rely on those who make such investments enjoying the opportunity to play a direct role in shaping resources which they themselves will need in the future.

We should thus view with caution arguments such as those advanced by Lewen to the effect that the proposed open ranking systems will solve a 'problem with restricted reviewing systems', namely that 'they are very vulnerable to personal preferences, prejudices and reviewers' egos.' [38] For it is one important lesson of the enduring success of expert peer review in so many different fields over more than three centuries that some biases (roughly: the complex set of learned biases we call 'expertise') need to be imposed upon the mix in order to ensure even minimal coherence. The reliance on experts brings, to be sure, a certain tendency in favor of established (i.e. most commonly accepted) scientific paradigms. But it is not clear how the addition of more voices to the mix should help resolve this problem, particularly if so doing has the effect of driving away just those persons who are in the position of making contributions resting on scientific expertise.

As Sowa points out, the Web has brought about a situation in which

> [p]ublication is almost free, and we have the luxury of decoupling the reviewing process from the gatekeeping process. Metadata enables that decoupling … The metadata associated with each submission can indicate what tests were made, what the reviewers said, and what results the users, if any, obtained. Users can choose to see ontologies sorted by any criteria they want: in the order of best reviews, most thorough testing, greatest usage, greatest relevance to a particular domain, or any weighted combination. [38]

The problem, however, is that the obverse sign of these very same advances in the direction of publishing freedom is the potential for what we might call a *poisoning of the wells.* Sowa thus agrees with the Foundry on the importance of maintaining an expert peer review process having a level of rigor that is comparable to that of existing scientific journals. This view is supported also by the experience of open access journals such as *PLOS ONE* who have experimented with dual frameworks involving both expert peer review and community-based dialogue on published articles.

Certainly there is one sort of openness that is essential to the advance of science. Science progresses only if it is open to new hypotheses and to new criticisms of existing hypotheses. It is this which explains why there are multiple, independent publishers of scientific journals, and why new journals are constantly being established. It is this which explains, also, why Noy is right to warn against a situation in which reviewers would be restricted to 'the experts appointed by a closed board'.

From this, however, it does not follow that our only alternative is a situation in which users would be allowed to draw only on the expertise of those reviewers willing to have their opinions made subject to review under an 'open' democratic voting process. A middle way would draw precisely on the lessons learned from scientific publishing by allowing users of ontology selection software to draw also on the contributions of experts employing traditional methods of ('closed') peer review. The NCBO BioPortal [5] might, for example, provide its users with the option to bypass the democratically supplied rankings of reviewers and to move directly to a prevalidated list of ontologies such as is provided by the Foundry (and we note that, to the degree that users of ontologies are encouraged to take advantage of such an option, silo problems will be averted in the future). The Foundry might itself then also be able to benefit from comments of Bioportal users, for example in supporting vetting of Foundry ontologies for errors, or in assessing the degree to which the terms used in ontologies might gain consensus approval on the part of significant numbers of users.

6 Conclusion: Ontology (Science) vs. Ontology (Engineering)

We can summarize our arguments by pointing to certain special features which are possessed by reference ontologies created to serve scientific purposes. Such ontologies are developed to be (1) common resources (thus they cannot be bought or sold), (2) for representation of well-demarcated scientific domains; they are (3) subject to constant maintenance by domain experts, (4) designed to be used in tandem with other, complementary ontologies, and (5) independent of format and implementation.

Sadly, the view still predominating in engineering circles is that ontologies need possess none of these features because ontologies are of their nature engineering artifacts [39]. It is as if all ontologies, both inside and outside science, are assigned by default the status of application ontologies. This leaves no room for any foundation of application ontologies in reference ontologies, and thus undermines what we believe to be the only promising strategy for addressing the problem of silo creation. Indeed it reinforces those very expectations on the part of many ontology engineers which have done so much to cause this problem in the first place.

We believe, in this light, that if we are to have a chance of resolving the silo problem, then recognition of this fact must bring in its wake a new approach to the training of ontologists working in support of scientific research, based on a new set of expectations to the effect that the authoring and maintenance and evaluation of scientific ontologies is an incremental, empirical, cumulative, and collaborative (i.e., precisely, scientific) activity that must be carried out by experts in the relevant scientific domains. Practitioners of ontology (science) will need to learn to see ontologies in contexts in which they are required to work well not only from a logical and a technological point of view, but also from the point of view of supporting the advance of science. To bring about these changes it may be necessary also to address the degree to which educational opportunities for ontologists are still largely confined to departments of computer science, and thereby also to address the degree to which tenure decisions for ontologists are made on a basis which awards too little weight to the contributions made by ontology to scientific research.

These recommendations receive support from Akkermans and Gordijn [40], who point out that computer scientists and knowledge engineers still standardly conceive ontologies as computer science artifacts, which means that they still see an ontology developed to serve (for example) biology as 'just another application' of their own computational expertise, and thus as something that is of lesser scientific importance than core computer science issues for example in logic or in systems for ontology mapping [40]. This 'self-limiting approach' will in the end 'not be able to exploit the full potential of the ontology idea', and Akkermans and Gordijn accordingly insist that the ontologies developed for scientific purposes need to be taken much more seriously as first-class citizens by computer scientists and knowledge engineers.

Empirical evidence of the benefits to be gained from such a move has been accumulating for some time, and we are gratified that this recognition is now beginning to make itself manifest also within the framework of the Semantic Web. There, too, we can see how mature ontologies, often resting on input from the OBO Foundry, are finally beginning to be put to serious scientific use [41].

Acknowledgements. This work was funded through the NIH Roadmap for Medical Research, Grant 1 U 54 HG004028. With thanks to Robert Arp, Holger Lewen, Suzanna Lewis, Mark Musen, Natasha Noy, Mitsu Okada, Richard Scheuermann, John Sowa, and Holger Stenzhorn for helpful comments.

References

[1] http://www.geneontology.org/. Last accessed June 24, 2008.
[2] http://obofoundry.org/. Last accessed June 24, 2008.
[3] B. Smith *et al.*, The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration, *Nature Biotechnology* **25** (2007), 1251–1255.
[4] D. Rubin *et al.*, The National Center for Biomedical Ontology: Advancing Biomedicine through Structured Organization of Scientific Knowledge, *OMICS*, **10** (2006), 185–198.
[5] http://www.bioontology.org/ncbo/faces/index.xhtml. Last accessed June 24, 2008.
[6] B. Smith *et al.,* Relations in Biomedical Ontologies, *Genome Biology*, **6** (2005), R46.
[7] J. J. Cimino and X. Y. Zhu, The Practical Impact of Ontologies on Biomedical Informatics, *Yearbook of Medical Informatics*, 2006, 124-35.
[8] D. P. Hill, J. A. Blake, J. E. Richardson, M. Ringwald, Extension and integration of the Gene Ontology: combining GO vocabularies with external vocabularies, *Genome Research*, **12** (2002),1982–1991.
[9] C. J. Mungall, Obol: Integrating Language and Meaning in Bio-Ontologies, *Comparative and Functional Genomics* **5** (2004), 509–520.
[10] C. Rosse and J. L. V. Mejino, Jr. "The Foundational Model of Anatomy Ontology", in A. Burger, *et al.* (eds.), *Anatomy Ontologies for Bioinformatics: Principles and Practice*, New York: Springer, 2007.
[11] C. Rosse and J. L. V. Mejino Jr. A Reference Ontology for Bioinformatics: The Foundational Model of Anatomy, *Journal of Biomedical Informatics*, **36** (2003), 478–500.
[12] H. Stenzhorn, *et al*. "Simplifying Access to Large-Scale Health Care and Life Sciences Datasets ", The Semantic Web: Research and Applications (Lecture Notes in Computer Science 5021 (2008), 864-868.
[13] R. D. Shankar, *et al.*, Epoch: An Ontological Framework to Support Clinical Trials Management. *International Workshop on Healthcare Information and Knowledge Management*, 2006, 25–32.
[14] J. F. Brinkley, *et al*. A Framework for Using Reference Ontologies as a Foundation for the *Semantic Web*, *Proceedings of the AMIA Fall Symposium*, 2006, 96–100.
[15] http://obi.sourceforge.net/. Last accessed June 24, 2008.
[16] D. P. Hill, *et al.*, Gene Ontology annotations: What they mean and where they come from, *BMC Bioinformatics*, **9** (Suppl. 5) (2008), S2.
[17] http://tesuque.stanford.edu/biocurator.org/. Last accessed June 24, 2008.
[18] Gene Ontology Consortium, The Gene Ontology (GO) project in 2006, *Nucleic Acids Research*, **34** (Database Issue) (2006), D322-D326.
[19] http://www.berkeleybop.org/ontologies/. Last accessed June 24, 2008.
[20] Rassinoux A. M., *et al*., Coupling Ontology Driven Semantic Representation with Multilingual Natural Language Generation for Tuning International Terminologies, *Medinfo*, **12** (2007), 555–9.
[21] http://ontolog.cim3.net/forum/ontology-summit/2008-03/msg00076.html. Last accessed June 24, 2008.
[22] T. Jefferson, *et al*. Effects of editorial peer review – A systematic review. *Journal of the American Medical Association*, **287** (2002), 2784–2786.
[23] M. Ware, *Peer Review: Benefits, Perceptions and Alternatives* (PRC Summary Papers 4), 2008.
[24] Editorial: Peer Review and Fraud. *Nature* **444** (2006), 971–972.
[25] http://ontolog.cim3.net/forum/ontology-summit/2008-03/msg00070.html. Last accessed June 24, 2008.
[26] S. Weber, *The Success of Open Source*, Cambridge, MA: Harvard University Press, 2004.
[27] *Nature Molecule Pages*: http://www.signaling-gateway.org/molecule/. Last accessed June 24, 2008.
[28] B. Riley, Database Publication Presents Unique Challenges for the Peer Reviewer, *Nature* (2006), doi:10. 1038/nature04993.
[29] http://ontolog.cim3.net/forum/ontology-summit/2008-03/threads.html/. Last accessed June 24, 2008.
[30] http://www.obofoundry.org/cgi-bin/discussion.cgi.
[31] M. Musen, If We Build It, Will They Come? Available for download at: http://ontolog.cim3.net/cgi-bin/wiki.pl?ConferenceCall_2007_12_06/. Last accessed June 24, 2008.
[32] http://www.neon-project.org/. Last accessed June 24, 2008.
[33] Despite enthusiasm for the concept, open peer review was not widely popular, either among authors or by scientists invited to comment. *Nature* (December, 2006). DOI:10.1038/nature05535.
[34] N. Noy, R. Guha, and M. Musen, User Ratings of Ontologies: Who Will Rate the Raters? *Proceedings of the AAAI 2005 Spring Symposium on Knowledge Collection*. Available for download at: http://smi.stanford.edu/smi-web/research/details.jsp?PubId=1112. Last accessed June 24, 2008.
[35] B. Smith, The Evaluation of Ontologies: Editorial Review vs. Democratic Ranking, *Proceedings of InterOntology 2008* (Tokyo, Japan, 26-27 February 2008), 127–138.
[36] http://ontolog.cim3.net/forum/ontology-summit/2008-03/msg00074.html. Last accessed June 24, 2008.

[37] H. Lewen, K. S. Supekar, N. F. Noy, M. A. Musen, Topic-Specific Trust and Open Rating Systems: An Approach for Ontology Evaluation. *Workshop on Evaluation of Ontologies for the Web* (EON2006). Available at: http://smi.stanford.edu/smi-web/research/details.jsp?PubId=1142.

[38] http://ontolog.cim3.net/forum/ontology-summit/2008-03/msg00076.html. Last accessed June 24, 2008.

[39] http://ontolog.cim3.net/forum/ontology-summit/2008-03/msg00067.html/. Last accessed June 24, 2008.

[40] H. Akkermans and J. Gordijn, Ontology Engineering, Scientific Method and the Research Agenda, in S. Staab and V. Svatek (Eds.), *EKAW 2006* (Lecture Notes in Artificial Intelligence, 4248)¸ 2006, 112-125.

[41] A. Ruttenberg, *et al.* Advancing Translational Research with the Semantic Web, *BMC Bioinformatics*, **8** (Suppl 3), (2007), S2.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-36

The Ontological Square and its Logic

Luc SCHNEIDER [a]
[a] *Institut Jean Nicod (CNRS, EHESS, ENS)*
Pavillon Jardin, École Normale Supérieure, 29 rue d'Ulm, F–75005 Paris

Abstract. The Ontological Square is a four-categorial scheme that is obtained by crossing two formal distinctions which underpin conceptual modelling languages and top-level ontologies alike: that between types (or universals) and tokens (or particulars) on the one hand, and that between characters (or features) and their bearers (or substrates) on the other hand. Thus the Ontological Square consists of particular substrates, called substances, and universal substrates, called kinds, as well as particular characters, called modes or moments, and universal characters, called attributes. In this article, I try to elucidate the basic ontological assumptions underlying this four-category scheme and I propose a calculus of many-sorted second-order logic that is meant to capture these intuitions by an enrichment of standard atomic logical form. A first-order semantics can be designed for such a Logic of the Ontological Square, with respect to which the latter's soundness can be established.

Keywords. Universals, particulars, substances, modes, second-order logic

1. The Ontological Square

1.1. Overview

In *Categories 1a20–1b10*, Aristotle [1] suggests two orthogonal distinctions between entities: *types* vs. their *tokens* on the one hand and *characters* vs. their *subjects* (or *bearers*) on the other hand. The resulting four-fold categorial scheme, which is sometimes called the *Ontological Square* [2, p. 12], consists of the following items [3, chap. 2] [4]:

	Subjects	**Characters**
Types	Kinds (e.g. *Man*)	Attributes (e.g. *Wisdom*)
Tokens	Substances (e.g. *Socrates*)	Modes (e.g. *Socrates' wisdom*)

Substances, like commonsense objects such as organisms and artifacts, are tokens of *kinds*, e.g. *Man* or *Chair*. The characters or features of substances are *modes* (often also called *moments*), such as qualities (e.g. the wisdom of Socrates), bearer-specific relations (e.g. Mary's love for Sam), states (e.g. Mary's sadness), powers or dispositions (e.g. the brittleness of a glass or Mary's fickleness), events (e.g. Nelson's victory at Trafalgar) and spatial boundaries (such as surfaces and edges). Modes are tokens of *attributes* such as *Wisdom* or *Brittleness* which (may) be generic characters of certain kinds of objects like *Philosopher* or *Glass*.

The Ontological Square underlies not only conceptual modelling paradigms such as the Entity-Relationship Model or the Unified Modelling Language, but also reference ontologies like DOLCE or BFO [5]. Hence by analysing this categorial scheme I do not so much intend to develop yet another top-level ontology than to provide a rigorous account of a formal paradigm that pervades a wide range of theoretical frameworks used in conceptual modelling and in knowledge representation. For this reason the research described in this paper is primarily of foundational nature.

The four-category ontology is a conceptual scheme which is far more subtle than the one that underpins conventional predicate logic, since it is articulated around two formal ties, namely *instantiation*, which holds between types and tokens, and *characterisation*, which holds between characters and their bearers [3, pp. 22, 40, 60, 79, 93 & 111]:

$$
\begin{array}{ccccc}
 & & \textbf{Characterisation} & & \\
 & \text{Kinds} & \longleftarrow & \text{Attributes} & \\
\textbf{Instantiation} & \Downarrow & & \Downarrow & \textbf{Instantiation} \\
 & \text{Substances} & \longleftarrow & \text{Modes} & \\
 & & \textbf{Characterisation} & &
\end{array}
$$

Besides instantiation and characterisation, there are two (respectively three) other ties that make up the structure of the Ontological Square, namely *subsumption* (which holds between types) as well as (occurrent and dispositional) *exemplification* (which holds between substances and attributes). It is worth mentioning that these formal ties are not entities *sui generis*, but supervene on their arguments in the sense of being necessitated by the latter. They are no addition to what there is, but are merely different sorts of “glue” that hold together the denizens of the four realms of being delimited by the Ontological Square.

1.2. The fundamental ties of the Ontological Square

1.2.1. Instantiation

Instantiation binds types to their tokens or instances: kinds are instantiated by substances and attributes by modes as particular features of substances. The bond between a type and its tokens or instances may superficially seem akin to that between a class and its members: we do often say that instances “belong to” or are “members” of a certain type. But contrary to classes, types are generally considered to be *intensional*: there may be distinct types sharing all their instances, e.g. the types “cordate” (animal having a heart) and “renate” (animal having kidneys) [6, pp. 25–26]. The extensional identity criterion would even fail in the case of *necessarily* co-extensive types such as “three-sided plane figure” and “tree-angled plane figure” [6, p. 26].

The distinction between types and tokens is generally conceived of as that between *universals* and *particulars* [6, pp. 1–2, 6–7]. According to D. C. Williams [7], the distinction between universals and particulars follows from considerations regarding the principle that (intrinsic or non-relational) indiscernibility implies identity. Indeed, while it is possible that there are two distinct particulars that are qualitatively, though not necessarily relationally undistinguishable (like the whiteness of a rose and the whiteness of a lotus), there cannot be several qualitatively indiscernible universals (like two *Whitenesses*).

Assuming that there are neither uninstantiated types nor untyped tokens, one can conceive of instantiation as a (non-mereological) partial identity or overlap between universals and particulars: if the world is likened to a spreadsheet consisting of universals as rows and particulars as columns, then a particular's instantiating a certain type corresponds to the intersection of the former's column with the latter's row [8, pp. 141–142] [9, pp. 46–48].

Ultimately, if all universals are immanent, i.e. "in" their instances, then the distinction between types and tokens amounts to a difference in the method of counting or thrawling through reality: a universal is tantamount to a plurality of tokens counted as one [7]. This does not conflict with the fact that types are intensional: indeed, their non-extensionality just implies that there may be more than one way of counting several instances as one. Hence, from the point of view of ontological parsimony, immanent universals come for free, since the commitment to (pluralities of) particulars already involves the commitment to immanent universals.

1.2.2. Characterisation

Characterisation links features to their bearers: modes characterise substances and attributes characterise kinds. While all modes are particular features of substances, some (but not all) attributes may be generic features of kinds, such as *Rationality* with respect to *Man*. In order to distinguish specific from generic characterisation, I will henceforth refer to the former as *inherence* and to the latter simply as *attribution*. Attribution does not simply reflect inherence: that a type is an attribute of some kind does not imply that all substances of this kind have a mode that is an instance of the attribute. For example, Sixleggedness is an attribute of the Insect, but there are some insects which have more or less than six legs due to some mutation or accident. That characterisation between types does not amount to a generalisation of characterisation between (their) instances lends support to the distinction between universals and particulars.

Modes are often referred to as "ways" of substances [10, chap. 4] [6, p. 116]. They do not exist in abstraction from their bearer(s) [11] and are non-transferable in the sense of being specific to their bearers or relata [6, pp. 117–118]. A mode inheres in the same substance(s) in each world it exists at all. A substance, by contrast, may not only be the bearer of more than one mode, but may also have or lack certain modes it actually has, i.e. those that are not essential to it.

The distinction between substances and modes is commonly motivated by recurring to the criterion of *(in)dependence*: while substances are ontologically independent, modes depend on substances. Now, if "dependence" is supposed to mean "existential dependence", this criterion fails, for there can be no substances without modes as their ways. It will not do to argue that modes depend specifically on their bearers, while substances only generically depend on their features, since a substance may specifically require the existence of some of its modes, e.g. the event of its coming into existence. In comparison, Strawson's conception of ontological priority in terms of *identifiability-(in)dependence* [12, p. 17] seems much more conspicuous. According to this conception, the items of a category *A* depend on the items of a category *B* iff the *As* can only be identified provided the *Bs* are already singled out.[1]

Substances are identifiability-independent for they are suitably dimensioned to ground a synchronic spatial framework of reference and also invariant enough to be re-identifiable and thus to support a diachronic framework of reference. Single modes, however, cannot provide such frameworks of reference and thus are identifiability-dependent on substances. Empirical psychology seems to corroborate this ontological priority of substances over modes. Indeed, experiments as to how members of different linguistic communities nonverbally represent events have shown that there is a language-independent preference for a certain ordering: first the substances as participants (i.e. the agent followed by the theme), then the event itself [13].

If inherence of monadic or singly-dependent modes amounts to parthood, then substances turn out to be sums or aggregates of monadic modes. This does not contradict the claim that substances have ontological priority over modes, since it could be reformulated as follows: while single modes cannot be identified on their own, certain pluralities of modes are thick enough to be nodes in a framework of reference. Now, assuming that a whole is nothing outside or above its parts [14, pp. 80–81], a substance may be regarded as the plurality of its non-relational modes counted as one, in the same way as a type is just a plurality of tokens counted as one. Like the type-token distinction, the dichotomy of substances vs. modes deflates to a difference in the method of counting or thrawling through reality. Thus substances come for free, since the commitment to (pluralities of) modes entails the commitment to substances.

1.2.3. Exemplification

The diagonal which glues particular substances and universal attributes together is the only formal tie expressed by the atomic logical form of conventional predicate calculus. According to Lowe [3, pp. 30–32], there are actually two distinct, but not mutually exclusive, sorts of exemplification, which can be accounted for in terms of the previously mentioned bonds:

1. *Dispositional* exemplification holds between a substance and an attribute iff the latter is an attribute of the kind which is instantiated by the former. E.g. since Sixleggedness is an attribute of the Insect, every particular insect dispositionally exemplifies Sixleggedness, even if it happens to have more or less than six legs.
2. *Occurrent* exemplification holds between a substance and an attribute iff the former has a mode that instantiates the latter. E.g. a particular insect that has lost two legs in an accident, occasionally exemplifies Fourleggedness, though as a member of the kind Insect it dispositionally exemplifies Sixleggedness.

Thus, a substance may exemplify an attribute, i.e. type of modes, without actually having an instance of this attribute as its mode, but merely in virtue of the attribute being a generic feature of the substance's kind. However, a substance may exemplify an attribute both dispositionally *and* occasionally, simply because it has an instance of the attribute *and* the latter happens to be an attribute of its kind: this is the case of an insect that actually has six legs.

[1] This is, of course, a generic notion of dependence; specific dependence could be defined as follows: an x depends on some y iff x cannot be identified unless y has already been singled out. However, for the purpose of distinguishing substances from modes, only the generic notion of identifiability-dependence is needed.

Note that attributes may be exemplified by many substances, while modes are bearer-specific; in other words, attributes are *repeatable*, while modes are not. Note that the repeatable-unrepeatable distinction is not tantamount to the universal-particular dichotomy, since there are types which are not attributes, namely kinds, and instances which are not modes, namely substances.

1.2.4. Subsumption

This hierarchical tie between types (not charted on the diagram above) is crucial in Aristotelian syllogistics and in object-oriented design. Under the assumption that types are not extensional, subsumption cannot be accounted for in terms of co-instantiation and hence has to be accepted as primitive. Nonetheless one may conceive of subsumption as a form of partial identity between types, i.e. as the subsuming universal's being contained in the subsumed one. Hence the attribute Blue is contained in the subsumed attribute Turquoise and the kind Insect is contained in the subsumed kind Ant. Note that the direction of containment between types is opposite to the direction of the inclusion between their extensions. Thus Blue is contained in Turquoise, but the extension of Turquoise is a subclass of the extension of Blue.

For the sake of generality, I will make but few assumptions as to the formal structure of subsumption. First, I will assume that it is an ordering relation between types of the same adicity (kinds being types of adicity 0). Furthermore, subsumption and attribution (i.e. characterisation on the level of types) interact with each other in the sense that characters of the subsumer are inherited by the subsumee. Hence if Sixleggedness is an attribute of the Insect and the Ant is a subkind of the Insect, then Sixleggedness is also an attribute of the Ant.

1.3. The idea of a Logic of the Ontological Square (LoOS)

For the sake of a minimal correspondence between language and reality, logical form should reflect ontological form, i.e. predication should mirror the ways in which items of the different ontological categories can stick together. Thus adopting the Ontological Square as a metaphysical scheme involves a revision of standard atomic logical form which expresses exemplification only. A logic based on the Ontological Square should provide four forms of predication instead of only one as in the case of classical predicate logic, namely one for instantiation, one for inherence, one for attribution and one for subsumption. Furthermore predication in such a logic is predication without predicates, i.e. without syntactically unsaturated terms the slots of which have to be filled by saturated individual terms. Instead it consists in the concatenation of names of different syntactical categories constrained by the adicity of one or all of these terms.

One may object that such a revision is unnecessary because the ties of the Ontological Square could be expressed by predicates in standard first-order logic. The problem, however, is that the ties of characterisation, namely inherence and attribution, are *multigrade*, i.e. they do not have a fixed arity: characterisation holds between a character and an arbitrary number of substrates or bearers. Now, the available strategies to tackle this conundrum in the context of classical first-order logic are all equally unintuitive. On the one hand, one could conceive of characterisation as holding between a feature and a tuple of bearers. This solution, however, would not do justice to our intuition that the feature (mode or attribute) is immediately tied to it(s) bearer(s). Furthermore, one would need to introduce a special ontological category, namely tuples, whose only function is to ensure an order between the bearers of the character: a step which is both artificial and ontologically wasteful.

On the other hand, one may adopt the view that there is an inherence or attribution tie for each arity – but this leads to a multiplication of inherence and attribution ties, while intuitively there is only one inherence or attribution tie. Leaving aside the introduction of so-called "vectors" [15, pp. 47–55], i.e. terms that denote ordered pluralities of arbitrary length, the simplest solution is to let the ties of the Ontological Square be mirrored by different ways of concatenating basic terms, i.e. by distinct sorts of atomic formulae. Since any position of such an atomic formula may be quantified in, the resulting logic can be regarded to be second-order.

In the remainder of this article, I show how the intuitions underlying the four-category ontology can be captured in a calculus of many-sorted second-order logic, called the *Logic of the Ontological Square (LoOS)*, which is in part inspired by Lowe's [16] proposal of a sortal logic. The choice of a second-order framework for the four-category ontology a well as the revision of atomic logical form distinguishes the present approach from the formalisation of the four-category ontology by Neuhaus, Grenon and Smith [17]. LoOS, and thus the Ontological Square, can be given a first-order semantics [18, pp. 74–75], with respect to which its soundness can be established.

2. The formal system LoOS

2.1. The Language of LoOS

2.1.1. Signature of LoOS

LoOS contains terms referring to or ranging over types and their instances:

Type terms

- for any $n \geq 0$, an infinite list of n-adic *type variables*:

  $$X^n, Y^n, Z^n, X'^n, Y'^n, Z'^n, X''^n, Y''^n, Z''^n, \ldots$$

- for any $n \geq 0$, a denumerable list of n-adic *type constants*:

  $$A^n, B^n, C^n, A'^n, B'^n, C'^n, A''^n, B''^n, C''^n, \ldots$$

Instance terms

- for any $n \geq 0$, an infinite list of n-adic *instance variables*:

  $$x^n, y^n, z^n, x'^n, y'^n, z'^n, x''^n, y''^n, z''^n, \ldots$$

- for any $n \geq 0$, a denumerable list of n-adic *instance constants*:

  $$a^n, b^n, c^n, a'^n, b'^n, c'^n, a''^n, b''^n, c''^n, \ldots$$

I will refer to 0-adic type terms as *kind* terms and to 0-adic instance terms as *substance* terms. Type terms of non-zero adicity are called *attribute* terms and instance terms of non-zero adicity *mode* terms. Monadic type terms are also labeled as *property* terms and type terms of adicity $n \geq 1$ as *relation* terms. In LoOS, adicity is not an indicator of the degree of unsaturatedness, but a constraint on numerical patterns of the different admissable ways of concatenation.

Note that introducing names for types of any adicity is compatible with an abundant view of universals according to which there are few or no restrictions on boolean recombinations of sparse types. However, the commitment to attributes of any non-zero adicity does not conflict with the stance that this abundance can ultimately be reduced to a set of sparse attributes the adicities of which have an upper limit.

I use uppercase *sans serif* letters (with or without subscripts and adicity superscripts) as schematic variables for type terms: $\mathsf{X}^{(n)}$, $\mathsf{Y}^{(n)}$, $\mathsf{Z}^{(n)}$ stand for (n-adic) type variables, $\mathsf{A}^{(n)}$, $\mathsf{B}^{(n)}$, $\mathsf{C}^{(n)}$ for (n-adic) type constants, and $\mathsf{T}^{(n)}$, $\mathsf{T'}^{(n)}$, $\mathsf{T''}^{(n)}$, etc. for (n-adic) type terms. Similarly lowercase *sans serif* letters (with or without subscripts and adicity superscripts) are used as schematic variables for instance terms: $\mathsf{x}^{(n)}$, $\mathsf{y}^{(n)}$, $\mathsf{z}^{(n)}$ stand for (n-adic) instance variables, $\mathsf{a}^{(n)}$, $\mathsf{b}^{(n)}$, $\mathsf{c}^{(n)}$ for (n-adic) instance constants, and $\mathsf{t}^{(n)}$, $\mathsf{t'}^{(n)}$, $\mathsf{t''}^{(n)}$, etc. for (n-adic) instance terms. $\upsilon^{(n)}$, $\upsilon'^{(n)}$, $\upsilon''^{(n)}$, etc. represent any (n-adic) variables, $\alpha^{(n)}$, $\alpha'^{(n)}$, $\alpha''^{(n)}$, etc. any (n-adic) constants, and $\theta^{(n)}$, $\theta'^{(n)}$, $\theta''^{(n)}$, etc. stand for any (n-adic) terms. Lists of terms $\theta_1, \ldots, \theta_n$ may be abbreviated as $\langle\theta\rangle_n$.

In addition, LoOS contains instance as well as type identity as primitives, since LoOS is semantically first-order [18, pp. 91–92]:

Instance identity: the two-place predicate "$=_i$"
Type identity: the two-place predicate "$=_t$"

2.1.2. *Well-formed formulae of LoOS*

The logical constants of LoOS are negation ($\neg$), implication ($\rightarrow$) and the universal quantifier ($\forall\upsilon$); brackets are used to delimit the scope of these operators. All other logical constants (conjunction $\wedge$, disjunction $\vee$, equivalence $\leftrightarrow$, and the existential quantifier $\exists\upsilon$) are defined. The lower-case greek letters ϕ, ψ and ξ are used as schematic variables for formulas.

1. for any n, for any instance term t^n and type term T^n, $\ulcorner \mathsf{t}^n\, \mathsf{T}^n \urcorner$, meaning: "$\mathsf{t}^n$ instantiates T^n", is a well-formed formula.
2. for any $n \geq 1$, for any mode term t^n and list of substance terms $\langle \mathsf{t}^0 \rangle_n$, $\ulcorner \mathsf{t}^n\, \langle \mathsf{t}^0 \rangle_n \urcorner$, meaning: "$\mathsf{t}^n$ inheres in $\langle \mathsf{t}^0 \rangle_n$", is a well-formed formula.
3. for any $n \geq 1$, for any attribute term T^n and list of kind terms $\langle \mathsf{T}^0 \rangle_n$, $\ulcorner \mathsf{T}^n\, \langle \mathsf{T}^0 \rangle_n \urcorner$, meaning: "$\mathsf{T}^n$ is an attribute of $\langle \mathsf{T}^0 \rangle_n$", is a well-formed formula.
4. for any n, for any type terms T^0 and T'^n, $\ulcorner \mathsf{T}^n\, \mathsf{T}'^n \urcorner$, meaning: "$\mathsf{T}^n$ is subsumed by T'^n", is a well-formed formula.
5. for any n, for any instance terms t^n and t'^n, $\ulcorner \mathsf{t}^n =_i \mathsf{t}'^n \urcorner$ is a well-formed formula.
6. for any n, for any type terms T^n and T'^n, $\ulcorner \mathsf{T}^n =_t \mathsf{T}'^n \urcorner$ is a well-formed formula.
7. If ϕ and ψ are well-formed formulae, then so are $\ulcorner \neg\phi \urcorner$ and $\ulcorner \phi \rightarrow \psi \urcorner$.
8. If ϕ is a well-formed formula and υ is any variable, then $\ulcorner \forall\upsilon(\phi) \urcorner$ is a well-formed formula.

Sequences of quantifiers of the same kind may be abbreviated as follows:

Definition 1. $\forall\langle\upsilon\rangle_n\, \phi(\langle\upsilon\rangle_n) \equiv_{df} \forall\upsilon_1 \ldots \forall\upsilon_n\, \phi(\upsilon_1, \ldots \upsilon_n)$

Definition 2. $\exists\langle\upsilon\rangle_n\, \phi(\langle\upsilon\rangle_n) \equiv_{df} \exists\upsilon_1 \ldots \exists\upsilon_n\, \phi(\upsilon_1, \ldots \upsilon_n)$

If υ is a variable occurring in a well-formed formula ϕ, then υ is *bound* by $\forall\upsilon$ within $\forall\upsilon(\phi)$, except in subformulae of the form $\forall\upsilon(\psi)$. Any variable υ that is not bound in a formula ϕ is *free in* ϕ. A term θ is *free in* ϕ iff θ is a constant or a variable that is free in ϕ. in A formula ϕ with no free variables is called a *closed formula* or a *sentence*. If ϕ is any formula, and θ_1 a term occurring free in ϕ, then $\phi^{\theta_2}_{\theta_1}$ is the result of substituting one or more occurrences of θ_1 in ϕ with occurrences of θ_2.

2.2. *The Deductive System of LoOS*

2.2.1. *Logical axioms and inference rules*

The logical axioms and inference rules of LoOS are *mutatis mutandis* those of restricted second-order logic (i.e. second-order logic without extensionality and comprehension axioms) with identity [18, pp. 65–66].

For any formulae ϕ, ψ or ξ, the logical axioms of LoOS are the following:

Axiom 1. $\phi \rightarrow (\psi \rightarrow \phi)$

Axiom 2. $(\phi \rightarrow (\psi \rightarrow \xi)) \rightarrow ((\phi \rightarrow \psi) \rightarrow (\phi \rightarrow \xi))$

Axiom 3. $(\neg\phi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \phi)$

Axiom 4. *(For any n, for any formula ϕ, instance variable x^n and instance term t^n occurring free in ϕ:)*

$$\forall \mathsf{x}^n\, \phi(\mathsf{x}^n) \rightarrow \phi(\mathsf{t}^n)$$

Axiom 5. *(For any n, for any formula ϕ, type variable X^n and type term T^n occurring free in ϕ:)*

$$\forall \mathsf{X}^n\, \phi(\mathsf{X}^n) \rightarrow \phi(\mathsf{T}^n)$$

The adicity restriction in Axioms 4 and 5 (that is superfluous in Rules 2 and 3, since the universal generalisation pertains to the same variable) is motivated in subsection 3.2.

Token and type identity are gouverned by the usual axioms:

Axiom 6. $\forall \mathsf{x}(\mathsf{x} =_i \mathsf{x})$

Axiom 7. $\forall \mathsf{x} \forall \mathsf{y}(\, \mathsf{x} =_i \mathsf{y} \rightarrow \; (\phi \leftrightarrow \phi^{\mathsf{y}}_{\mathsf{x}})\,)$

Axiom 8. $\forall \mathsf{X}(\mathsf{X} =_t \mathsf{X})$

Axiom 9. $\forall \mathsf{X} \forall \mathsf{Y}(\, \mathsf{X} =_t \mathsf{Y} \rightarrow \; (\phi \leftrightarrow \phi^{\mathsf{Y}}_{\mathsf{X}})\,)$

A formula ϕ is *derivable in LoOS* from a set of sentences Δ ($\Delta \vdash_{LoOS} \phi$) iff there is a finite sequence of formulas $\phi_1, \ldots, \phi_n$ such that $\phi = \phi_n$ and, for any $i \leq n$, ϕ_i is in Δ, and is either an axiom of LoOS or such as to follow from previous members of the sequence by one of the following inference rules.

Rule 1. *(For any formulae ϕ and ψ:) From ϕ, $\phi \rightarrow \psi$ infer ψ*

Rule 2. *(For any formulae ϕ and ψ and any instance variable x that does not occur free in ϕ or in any of the premisses in the derivation:)*

From $\phi \rightarrow \psi(x)$ *infer* $\phi \rightarrow \forall x\, \psi(x)$

Rule 3. *(For any formulae* ϕ *and* ψ *and any type variable* X *that does not occur free in* ϕ *or in any of the premisses in the derivation:)*

From $\phi \rightarrow \psi(X)$ *infer* $\phi \rightarrow \forall X\, \psi(X)$

Since the formal system of LoOS is a Hilbert-style calculus, one may establish the deduction theorem for LoOS:

Theorem 1. *If* $\Delta \cup \{\phi\} \vdash_{LoOS} \psi$, *then* $\Delta \vdash_{LoOS} \phi \rightarrow \psi$

Proof. The proof of this meta-theorem for LoOS is the same as in standard predicate logic [19, par. 36 & par 51, p. 299], with the exception of the duplicate axioms and inference rules for the universal quantifier. □

2.2.2. Axioms of predication

The following axioms are meant to capture some ideas stated in the informal presentation of the Ontological Square above. With respect to instantiation, I assume that each token instantiates at least one type and that each type is instantiated.

Axiom 10. $\forall x \exists X (x\, X)$

Axiom 11. $\forall X \exists x (x\, X)$

As to inherence, I stipulate that each mode is non-transferable, i.e. that it inheres in at least and at most one (list of) substance(s).

Axiom 12. *(For any* $n \geq 1$*:)* $\forall x^n\, \exists \langle y^0 \rangle_n\, (x^n\, \langle y^0 \rangle_n\,)$

Axiom 13. *(For any* $n \geq 1$*:)*
$\forall x^n\, \forall \langle y^0 \rangle_n\, \forall \langle z^0 \rangle_n\, (\, (\, x^n\, \langle y^0 \rangle_n \wedge x^n\, \langle z^0 \rangle_n\,) \rightarrow (y^0_1 =_i z^0_1 \wedge \ldots \wedge y^0_n =_i z^0_n)\,)$

Subsumption is antisymmetrical and transitive.

Axiom 14. *(For any* n*:)*
$\forall X^n\, \forall Y^n\, (\, (X^n\, Y^n \wedge Y^n\, X^n) \rightarrow X^n =_t Y^n\,)$

Axiom 15. *(For any* n*:)*
$\forall X^n\, \forall Y^n\, \forall Z^n\, ((\, X^n\, Y^n \wedge Y^n\, Z^n\,) \rightarrow X^n\, Z^n\,)$

The direction of subsumption being opposite to the inclusion between extensions, every instance of the subsumee is an instance of the subsumer:

Axiom 16. *(For any* n*:)*
$\forall X^n\, \forall Y^n\, (\, X^n\, Y^n \rightarrow \forall x^n\, (\, x^n\, X^n \rightarrow x^n\, Y^n\,)\,)$

Furthermore, attribution is inherited downwards the subsumption hierarchy: attributes of the subsumer are also attributes of the subsumee.

Axiom 17. *(For any* $n \geq 1$*:)*
$\forall X^n\, \forall \langle X^0 \rangle_n\, \forall \langle Y^0 \rangle_n\, (\, X^n\, \langle X^0 \rangle_n \rightarrow (\, (\, Y^0_1\, X^0_1 \wedge \ldots \wedge Y^0_n\, X^0_n\,) \rightarrow X^n\, \langle Y^0 \rangle_n\,)\,)$

As mentioned in the introduction, dispositional exemplification of an attribute by a (list of) substance(s) ("$\mathsf{Y}^n \langle \mathsf{x}^0 \rangle_n$") can be defined in terms of attribution and instantiation.

Definition 3. $Y^n \langle x^0 \rangle_n \equiv_{df} \exists \langle X^0 \rangle_n \, (\, Y^n \, \langle X^0 \rangle_n \wedge x^0_1 \, X^0_1 \wedge \ldots x^0_n \, X^0_n \,)$

Occurrent exemplification of an attribute by some substances ("$\langle \mathsf{x}^0 \rangle_n \mathsf{Y}^n$") is definable in terms of inherence and instantiation.

Definition 4. $\langle x^0 \rangle_n Y^n \equiv_{df} \exists y^n \, (\, y^n \, \langle x^0 \rangle_n \wedge y^n \, Y^n \,)$

3. Meta-theory of LoOS

3.1. First-order semantics for LoOS

As a basis for the semantics of LoOS I adopt a so-called first-order model [18, pp. 74–75]. A *first-order model* $\mathfrak{M}^1$ of LoOS is a structure $\langle \mathfrak{U}, \mathfrak{U}_\star, \langle \mathfrak{I}, \mathfrak{in}, \mathfrak{ih}, \mathfrak{at}, \mathfrak{sb} \rangle \rangle$ such that:

1. for any n, $\mathfrak{U}(n)$ is a non-empty set called *universe of n-adic instances*;
2. for any n, $\mathfrak{U}_\star(n)$ is a non-empty set called *universe of n-adic types*;
3. for any distinct n, m, $\mathfrak{U}(n)$ and $\mathfrak{U}(m)$ are disjoint;
4. for any distinct n, m, $\mathfrak{U}_\star(n)$ and $\mathfrak{U}_\star(m)$ are disjoint;
5. $\cup_n \mathfrak{U}(n)$ and $\cup_n \mathfrak{U}_\star(n)$ are disjoint;
6. for any n, $\mathfrak{in}(n) \subseteq \mathfrak{U}(n) \times \mathfrak{U}_\star(n)$ is the instantiation relation such that

   $\mathfrak{in}_1$ for any $\mathfrak{i} \in \mathfrak{U}(n)$, there is a $\mathfrak{t}$ of $\mathfrak{U}_\star(n)$ such that $\langle \mathfrak{i}, \mathfrak{t} \rangle \in \mathfrak{in}(n)$;
   $\mathfrak{in}_2$ for any $\mathfrak{t}$ of $\mathfrak{U}_\star(n)$, there is an $\mathfrak{i} \in \mathfrak{U}(n)$ such that $\langle \mathfrak{i}, \mathfrak{t} \rangle \in \mathfrak{in}(n)$;

7. for any $n \geq 1$, $\mathfrak{ih}(n) \subseteq \mathfrak{U}(n) \times \mathfrak{U}(0)^n$ is the inherence relation such that for any $\mathfrak{i} \in \mathfrak{U}(n)$, there is exactly one list $\mathfrak{i}'_1, \ldots \mathfrak{i}'_n$ of elements of $\mathfrak{U}(0)$ such that $\langle \mathfrak{i}, \mathfrak{i}'_1, \ldots \mathfrak{i}'_n \rangle \in \mathfrak{ih}(n)$ [2];
8. for any n, $\mathfrak{sb}(n)$ is the subsumption relation on $\mathfrak{U}_\star(n)$ such that:

   $\mathfrak{sb}_1$ $\mathfrak{sb}(n)$ is antisymmetrical and transitive;
   $\mathfrak{sb}_2$ for each $\mathfrak{t}$, $\mathfrak{t}'$ of $\mathfrak{U}_\star(n)$, if $\langle \mathfrak{t}, \mathfrak{t}' \rangle \in \mathfrak{sb}(n)$, then, for every $\mathfrak{i}$ in $\mathfrak{U}(n)$, if $\langle \mathfrak{i}, \mathfrak{t} \rangle \in \mathfrak{in}(n)$, then $\langle \mathfrak{i}, \mathfrak{t}' \rangle \in \mathfrak{in}(n)$.

9. for any $n \geq 1$, $\mathfrak{at}(n) \subseteq \mathfrak{U}_\star(n) \times \mathfrak{U}_\star(0)^n$ is the attribution relation such that for any $\mathfrak{t} \in \mathfrak{U}_\star(n)$, $\langle \mathfrak{t} \rangle_n \in \mathfrak{U}_\star(0)^n$ and $\langle \mathfrak{t}' \rangle_n \in \mathfrak{U}_\star(0)^n$, if $\langle \mathfrak{t}, \mathfrak{t}_1, \ldots, \mathfrak{t}_n \rangle \in \mathfrak{at}(n)$, then, if $\langle \mathfrak{t}'_1, \mathfrak{t}_1 \rangle \in \mathfrak{sb}(0)$ and $\ldots$ and $\langle \mathfrak{t}'_n, \mathfrak{t}_n \rangle \in \mathfrak{sb}(0)$, it is also the case that $\langle \mathfrak{t}, \mathfrak{t}'_1, \ldots, \mathfrak{t}'_n \rangle \in \mathfrak{at}(n)$;
10. $\mathfrak{I}$ is a function called *interpretation* on $\mathfrak{M}^1$ such that:

    $\mathfrak{I}1$ for each type constant A^n, $\mathfrak{I}(\mathsf{A}^n) \in \mathfrak{U}_\star(n)$;
    $\mathfrak{I}2$ for each instance constant a^n, $\mathfrak{I}(\mathsf{a}^n) \in \mathfrak{U}(n)$;

11. $\mathfrak{M}^1$ is assumed to be faithful to the axioms for token and type identity.

Remark: The first-order semantics of LoOS differs from that of standard second-order logic by providing four sorts of "predication"-relations as denotations of the implicit

[2] where $\mathfrak{U}(m)^0 = \mathfrak{U}(m)^1 = \mathfrak{U}(m)$, and , for each $n \geq 1$, $\mathfrak{U}(m)^{n+1} = \mathfrak{U}(m)^n \times \mathfrak{U}(m)$

copulae. The reader will have noticed that the conditions spelled out for these relations match the axioms of predication, i.e. Axioms 10 to 17.

A *assignment on* $\mathfrak{M}^1$ is a function $\mathfrak{A}$ from the set of variables into $\bigcup_n \mathfrak{U}(n) \cup \bigcup_n \mathfrak{U}_\star(n)$ which assigns to each n-adic instance variable an element of $\mathfrak{U}(n)$ and to each n-adic kind variable an element of $\mathfrak{U}_\star(n)$. For any assignments $\mathfrak{A}$, $\mathfrak{A}'$ and any variables v, I write $\mathfrak{A} \simeq_v \mathfrak{A}'$ for "$\mathfrak{A}$ agrees with $\mathfrak{A}'$ on every variable except possibly v". I call *denotation on* $\mathfrak{M}^1$ a function $\mathfrak{D}$ such that, for any term θ, $\mathfrak{D}(\theta) = \mathfrak{I}(\theta)$ iff θ is a constant, and $\mathfrak{D}(\theta) = \mathfrak{A}(\theta)$ iff θ is a variable. In other words, all denotations on $\mathfrak{M}^1$ agree on the constants of LoOS, but may not agree on the variables of LoOS.

The *satisfaction* relation between a first-order model $\mathfrak{M}^1$ for LoOS, an assignment $\mathfrak{A}$ on $\mathfrak{M}^1$ and a sentence ϕ of LoOS ($\mathfrak{M}^1, \mathfrak{A} \models \phi$) can be defined as follows:

$\models 1$ $\mathfrak{M}^1, \mathfrak{A} \models \neg\phi$ iff $\mathfrak{M}^1, \mathfrak{A} \not\models \phi$;
$\models 2$ $\mathfrak{M}^1, \mathfrak{A} \models \phi \rightarrow \psi$ iff $\mathfrak{M}^1, \mathfrak{A} \not\models \phi$ or $\mathfrak{M}^1, \mathfrak{A} \models \psi$;
$\models 3$ $\mathfrak{M}^1, \mathfrak{A} \models \forall \mathsf{x}\, \phi(\mathsf{x})$ iff for every assignment $\mathfrak{A}'$ such that $\mathfrak{A}' \simeq_{\mathsf{x}} \mathfrak{A}$, $\mathfrak{M}^1, \mathfrak{A}' \models \phi(\mathsf{x})$;
$\models 4$ $\mathfrak{M}^1, \mathfrak{A} \models \forall \mathsf{X}\, \phi(\mathsf{X})$ iff for every assignment $\mathfrak{A}'$ such that $\mathfrak{A}' \simeq_{\mathsf{X}} \mathfrak{A}$, $\mathfrak{M}^1, \mathfrak{A}' \models \phi(\mathsf{X})$;
$\models 5$ for any n, for any instance term t^n, for any type term T^n, $\mathfrak{M}^1, \mathfrak{A} \models \mathsf{t}^n\, \mathsf{T}^n$ iff $\langle \mathfrak{D}(\mathsf{t}^n), \mathfrak{D}(\mathsf{T}^n)\rangle \in \mathfrak{in}(n)$;
$\models 6$ for any $n \geq 1$, for every mode term t^n and any list of substance terms $\langle \mathsf{t}^0 \rangle_n$, $\mathfrak{M}^1, \mathfrak{A} \models \mathsf{t}^n\, \langle \mathsf{t}^0 \rangle_n$ iff $\langle \mathfrak{D}(\mathsf{t}^n), \mathfrak{D}(\mathsf{t}^0_1), \ldots, \mathfrak{D}(\mathsf{t}^0_n)\rangle \in \mathfrak{ih}(n)$;
$\models 7$ for any $n \geq 1$, for every attribute term T^n and any list of kind terms $\langle \mathsf{T}^0 \rangle_n$, $\mathfrak{M}^1, \mathfrak{A} \models \mathsf{T}^n\, \langle \mathsf{T}^0 \rangle_n$ iff $\langle \mathfrak{D}(\mathsf{T}^n), \mathfrak{D}(\mathsf{T}^0_1), \ldots, \mathfrak{D}(\mathsf{T}^0_n)\rangle \in \mathfrak{at}(n)$;
$\models 8$ for any n, for any type terms T^n, T'^n, $\mathfrak{M}^1, \mathfrak{A} \models \mathsf{T}^n\, \mathsf{T}'^n$ iff $\langle \mathfrak{D}(\mathsf{T}^n), \mathfrak{D}(\mathsf{T}'^n)\rangle \in \mathfrak{sb}(n)$;
$\models 9$ for any n, for any instance terms t^n, t'^n, $\mathfrak{M}^1, \mathfrak{A} \models \mathsf{t}^n =_i \mathsf{t}'^n$ iff $\mathfrak{D}(\mathsf{t}^n) = \mathfrak{D}(\mathsf{t}'^n)$;
$\models 10$ for any n, for any type terms T^n, T'^n, $\mathfrak{M}^1, \mathfrak{A} \models \mathsf{T}^n =_t \mathsf{T}'^n$ iff $\mathfrak{D}(\mathsf{T}^n) = \mathfrak{D}(\mathsf{T}'^n)$;

A set S of sentences is satisfied by a first-order model $\mathfrak{M}^1$ and an assignment $\mathfrak{A}$ on $\mathfrak{M}^1$, iff the latter satisfy every member of S. A sentence ϕ is *first-order valid* iff for every first-order model $\mathfrak{M}^1$ and assignment $\mathfrak{A}$ on $\mathfrak{M}^1$, $\mathfrak{M}^1, \mathfrak{A} \models \phi$. An inference rule is *first-order sound* iff every first-order model $\mathfrak{M}^1$ and assignment $\mathfrak{A}$ $\mathfrak{M}^1$ that satisfy its premisses also satisfy its conclusions.

3.2. Soundness

Theorem 2. *Every theorem of LoOS is first-order valid.*

Proof. Soundness is established by a simple inspection of the axioms and inference rules of LoOS: if all axioms of LoOS are first-order valid and all inference rules of LoOS are first-order sound, then all theorems of LoOS are first-order valid. That the logical axioms and inference rules of LoOS are first-order valid, respectively first-order sound, is obvious in virtue of clauses $\models 1$ to $\models 4$ (note that $\models 3$ and $\models 4$ are just special, i.e. sorted, cases of the usual definition of satisfaction of a universally quantified formula in a given model and assigment). A clarification is in order with respect to the adicity restriction in Axioms 4 and 5: since any n-adic type or instance variable v^n is assigned an element of the domain of n-adic types or instances, and since $\forall \upsilon^n\ \phi(\upsilon^n)$ is satisfied by a given model $\mathfrak{M}^1$ and a given assigment $\mathfrak{A}$ iff $\phi(\upsilon^n)$ is true in $\mathfrak{M}^1$ under all assigments for the variable υ^n, then, for any term θ, $\phi(\theta)$ is satisfied by $\mathfrak{M}^1$ and $\mathfrak{A}$, too, *provided* the term θ has a denotation in the same domain as υ^n, i.e. if θ is an n-adic (type or instance) term.

Clause $\models 5$ grounds the first-order validity of Axioms 10 and 11, given the definition of $\mathfrak{in}(n)$ (for any n). The first-order validity of Axioms 12 and 13 follows from $\models 6$, given the definition of $\mathfrak{ih}(n)$ (for any n). $\models 7$ ensures that Axiom 17 is first-order valid, given the definition of $\mathfrak{at}(n)$ (for any n). Axioms 14 to 16 regarding subsumption are first-order valid due to clause $\models 8$, given the definition of $\mathfrak{sb}(n)$ (for any n). $\models 9$ guarantees the first-order validity of Axioms 6 and 7, while Axioms 8 and 9 are first-order valid in virtue of clause $\models 10$ (since $\mathfrak{M}^1$ is assumed to be faithful to the axioms for identity). $\square$

4. Conclusions and future work

The aim of this paper has been to clarify the intuitions underlying the Ontological Square and to propose a logical framework designed to express these intuitions. As far as possible, I have tried to provide a deflationary account of the basic ontological distinctions and to motivate the choice of a non-standard approach to predication and atomic logical form. The changes with respect to standard second-order logic, however, are minimal and the semantics of the Logic of the Ontological Square is only a variant of conventional first-order semantics for second-order languages.

Future tasks include a completeness proof for LoOS as well as the integration of a minimal temporal ontology of events and change in this formal framework. But I hope to have convinced the reader of the utility of the Logic of the Ontological Square for the study of a set of metaphysical distinctions that are common to a wide range of paradigms in conceptual modelling and in knowledge representation.

Acknowledgements

Research leading up to the present article has been supported by a grant from the Boninchi Foundation (Geneva), as well as the Philibert Collart Prize from the Société Académique de Genève. I would like to thank Kit Fine, Jonathan Lowe, Fabrice Correia, Claudio Masolo, Chris Welty and Johannes Stern for discussion, as well as two anonymous reviewers for helpful suggestions.

References

[1] J. L. Ackrill. *Aristotle's* Categories *and* De Interpretatione. Oxford: Oxford University Press, 1963.
[2] I. Angelelli. *Studies on Gottlob Frege and Traditional Philosophy*. Dordrecht: Reidel, 1967.
[3] E. J. Lowe. *The Four Category Ontology. A Metaphysical Foundation for Natural Science*. Oxford: Oxford University Press, 2006.
[4] B. Smith. On Substances, Accidents and Universals: In Defence of a Constituent Ontology. *Philosophical Papers 26*, pp. 105–127, 1997.
[5] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. *The Wonderweb Library of Foundational Ontologies*. Wonderweb Deliverable D18. Trento: ISTC-CNR, 2003.
[6] D. M. Armstrong. *Universals. An Opinionated Introduction*. Boulder, London: Westview Press, 1989.

[7] D. C. Williams. Universals and existents. *Australasian Journal of Philosophy 64*, pp. 114, 1986.
[8] D. M. Armstrong. How do Particulars Stand to Universals ? In D. W. Zimmerman (ed.), *Oxford Studies in Metaphysics, vol. 1*. Oxford: Clarendon Press, pp. 139–154, 2004.
[9] D. M. Armstrong. *Truth and Truthmakers*. Cambridge: Cambridge University Press, 2004.
[10] D. A. J. Seargent. *Plurality and Continuity: An Essay in G. F. Stout's Theory of Universals*. Dordrecht: Martinus Nijhoff, 1985.
[11] C. B. Martin. Substance Substantiated. *Australasian Journal of Philosophy 58*, pp. 3–10, 1980.
[12] P. F. Strawson. *Individuals. An Essay in Descriptive Metaphysics.* London: Routledge, 1959.
[13] S. Goldin-Meadow et al.. The natural order of events: How speakers of different languages represent events nonverbally. *Proceedings of the National Academy of Sciences vol. 105, no. 27*, pp. 9163–9168, 2008.
[14] D. Lewis. *Parts of Classes*. Oxford: Blackwell, 1991.
[15] K. Hossack. *The Metaphysics of Knowledge*. Oxford: Oxford University Press, 2007.
[16] E. J. Lowe. *Kinds of Being. A Study of Individuation, Identity and the Logic of Sortal Terms*. Oxford: Blackwell, 1989.
[17] F. Neuhaus, P. Grenon, and B. Smith. A Formal Theory of Substances, Qualities and Universals. In A. Varzi and L. Vieu (eds.), *Formal Ontology in Information Systems. Proceedings of the Third International Conference (FOIS 2004)*. Amsterdam, Berlin, Oxford, Tokyo, Washington, DC: IOS Press, pp. 49–59, 2004.
[18] S. Shapiro. *Foundations without Foundationalism: A Case for Second-Order Logic*. Oxford: Oxford University Press, 1991.
[19] A. Church. *Introduction to Mathematical Logic*. Princeton, NJ: Princeton University Press, 1956.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-49

Formal Semantics and Ontologies

Towards an Ontological Account of Formal Semantics

Frank LOEBE [a,1] and Heinrich HERRE [b]

[a] *Department of Computer Science, University of Leipzig, Germany*
[b] *IMISE, University of Leipzig, Germany*

Abstract. Formal ontology relies on representation languages for expressing ontologies. This involves the formal semantics of these languages which is typically based on a limited set of abstract mathematical notions. In this paper, we discuss the interplay between formal semantics and the intended role of ontologies as semantic foundation. In this connection a circularity is identified if ontologies are to determine the conceptual equivalence of expressions. This is particularly relevant for ontologies which are to be provided in multiple formalisms. In order to overcome this situation, *ontological semantics* is generally defined as a novel kind of semantics which is purely and directly based on ontological entities. We sketch a specific application of this semantics to the syntax of first order logic. In order to beneficially rely on theoretical results and reasoning systems, an approximation of the proposed semantics in terms of the conventional approach is established. This results in a formalization method for first order logic and a translation-based variant of ontological semantics. Both variants involve an ontology for their application. In the context of developing a top-level ontology, we outline an ontology which serves as a meta-ontology in applying ontological semantics to the formalization of ontologies. Finally, resolved and remaining issues as well as related approaches are briefly discussed.

Keywords. formal semantics, ontology, ontological semantics, first order logic

1. Introduction

The development and application of ontologies frequently involves their provision in several distinct formalisms, adopting an understanding of "ontology" as a "conceptualization" rather than its "specification" in a particular language, cf. Gruber's definition [1, p. 199]. Especially top-level ontologies must be available in multiple formalisms in order to facilitate their application in distinct areas like conceptual modeling, information integration, and the Semantic Web. The issues in this paper arise in the context of a long-term research project of developing a top-level ontology, the General Formal Ontology (GFO)[2] [2,3]. The most relevant formalisms in our project are primarily logical languages like first order logic (FOL) and description logics (DL), but also languages employed in conceptual modeling, especially the Unified Modeling Language (UML) [4]. Providing formalizations of GFO in several formalisms leads to the problem of how

[1]Corresponding author. P.O. Box 100920, 04009 Leipzig, Germany. frank.loebe@informatik.uni-leipzig.de
[2]http://www.onto-med.de/ontologies/gfo

to justify that these formalizations capture the same contents, and can thus be used consistently. In a broader context, the same question is formulated in [5] for two arbitrary languages L_1 and L_2: "What exactly do we mean when we say that a set S_2 of L_2 sentences is a translation of a set S_1 of L_1 sentences?" This question is closely connected to the relationship between languages and their semantics, and we argue that this is still an open problem whose solution is intimately tied to ontologies.

In this paper, section 2 discusses interrelations of classical formal semantics and ontologies. We identify a specific circularity and argue that ontologies should play an important role for the foundation of *formal* semantics, in the context of ontology representation and meaning-preserving translations. Accordingly, section 3 introduces the notion of a formal, *ontological semantics*[3]. There we further outline an application of this semantics to the syntax of first order logic, and provide an approximation via classical FOL semantics in order to build on established work. The approximation results in a formalization method for FOL which itself requires an ontology to found syntactic compositions. Section 4 thus complements the approach by outlining an ontology which is proposed for applying ontological semantics to the formalization of ontologies. The motivating problems and the role of the approach in their regard are considered, accompanied by related work, in section 5, before we conclude and mention future directions.

2. Analysis of the Roles of Ontologies and Formal Semantics

Let us start with an ontology Ω and two representations of it, $R_1(\Omega) \subseteq L_1$ and $R_2(\Omega) \subseteq L_2$, in the languages L_1 and L_2 with distinct formal semantics. Since every communication about Ω must rely on representations, an important question arises: How to justify that $R_1(\Omega)$ and $R_2(\Omega)$ are representations of one and the same ontology? More generally, what does it mean to state that two expressions in distinct languages have the same meaning? Our first central claim in this respect is that the established types of formal semantics of languages are insufficient for answering these questions. Of course, this does not trivialize their value and adequacy for other tasks, e.g. theoretical analyses of mutual (formal) expressiveness, consistency, decidability, or complexity issues.

The claim of the inadequacy of formal semantics for meaning-preserving translations is based on a previously established meta-architecture for analyzing ontology constituents [7], which distinguishes the notions of *abstract core ontology* (ACO) and *abstract top ontology* (ATO). To avoid wrong intuitions about these and despite the terminological proximity, note that ACO and ATO do not immediately relate to the common ontology classification into top-level, core / generic domain, and domain ontologies [8, Sect. 1.4]. An ACO functions as an *ontology* for ontology constituents, i.e., it refers to the question of what ontological kind ontology constituents are (e.g., categories, relations, or attributes). This forms an (ontological) meta-level for languages, i.e., ontology constituents link to the ACO level via instantiation. ACOs relate closely to the abstract syntax categories of a language and correspond to an ontological understanding of knowledge representation ontologies in [8, Sect. 1.4]. For instance, for OWL's abstract syntax categories classID and individualvaluedPropertyID [9] one may postulate categories and (binary) relations, respectively, as appropriate ontological kinds in a suitable ACO. In

[3] Our work is not specifically related to and clearly differs from the equally termed approach in [6], which addresses natural language processing and semantics.

contrast, an abstract top ontology refers to the mathematical notions underlying the classical formal semantics assigned to a language, i.e., it captures the ontology of the formal semantics. In the case of OWL, this would be standard set theory based on the notion of sets and the membership relation. Accordingly, ontological constituents are *encoded* by means of instances of an ATO. For example, a unary FOL predicate $Lion$ (viewed syntactically) is interpreted by a set in the classical formal semantics (the abstract top view). This set encodes a category C within an animal ontology, i.e., C instantiates "category" with respect to an ACO.

Problems in ontology representation originate from (a) the lack of explicating the ACO view during formalization and (b) different choices for encodings in the formal semantics for the same syntax. As an example for (b), a FOL theory may encode category C by a functional constant $lion$ or a unary predicate $Lion$. Distinct encodings create formal differences originating from the same ontological entity by capturing different aspects of it. This in turn accumulates problems for language translations, even if a provably sound translation between the formal semantics of those languages is available. For instance, such translation exists for standard DLs and FOL [10, Sect. 4.2]. But translating a DL theory which encodes polyadic relations as DL concepts [11, use case 3] to FOL is hard if one expects for FOL that polyadic relations are expressed by polyadic predicates. That means, different encodings with respect to the ATO level may require "non-standard" translations between languages.

The justification for such "non-standard" translations lies outside of formal semantics alone. We believe that there is a kind of *conceptual* or *intensional* semantics which refers to the intensions of users of formal languages and is prior to formalization acts, cf. also [12]. *Ontologies* were "invented" in the context of knowledge-based systems research in order to tackle this problem, among others, cf. [13,14]. The basic idea is that different systems or languages *commit to* a common ontology Ω in order to share conceptual meaning, which should allow for a notion of meaning-preserving translations based on Ω. As an exemplary case of how this is frequently understood we formulate Def. 1, already taking into account that Ω can only be involved through a representation $R(\Omega)$ of it.

Definition 1 Let $R(\Omega) \subseteq L_\Omega$ be a representation of an ontology Ω in a logical formalism L_Ω, i.e., a theory. Let L_1 and L_2 be two arbitrary languages, and $\tau_i : L_i \rightarrow L_\Omega$ for $i \in \{1, 2\}$ be translations from L_i to L_Ω with respect to $R(\Omega)$. Two expressions $e_1 \in L_1$ and $e_2 \in L_2$ are said to be *conceptually equivalent* with respect to τ_1 and τ_2 iff their translations into L_Ω are $R(\Omega)$-equivalent:

– for terms $\tau_1(e_1)$ and $\tau_2(e_2)$: $R(\Omega) \models_{L_\Omega} \tau_1(e_1) = \tau_2(e_2)$
– for formulas $\tau_1(e_1)$ and $\tau_2(e_2)$: $R(\Omega) \models_{L_\Omega} \tau_1(e_1) \leftrightarrow \tau_2(e_2)$ □

The representation formalism L_Ω in this definition is a problematic parameter. Due to the above analysis we deny the common assumption that logical languages are "ontologically neutral" [15, p. 492] and could be used without an ontological bias. Instead there is a vicious circle in this approach. Ontologies are meant to overcome insufficiencies of formal semantics with respect to conceptual equivalence, which is itself based on the formal equivalence defined for L_Ω and the encoding of Ω into L_Ω – and thus on the formal semantics of L_Ω. This yields two problems that will be addressed subsequently.

Problem 1 How to assign a semantics to a language that is directly based on ontologies and avoids the just-mentioned circularity.

A solution would further clarify how to represent and interpret ontologies ontologically. For this purpose we introduce the notion of *ontological semantics* in the next section. The approach is applicable to arbitrary languages and is intimately tied to ontologies. In order to apply ontological semantics to the formalization of ontologies, it is necessary to provide suitable abstract core ontologies.

Problem 2 is to develop and specify suitable abstract core ontologies.

The plural form indicates that we expect multiple solutions for Problem 2. Sect. 4 outlines a proposal for a small yet powerful abstract core ontology. In general, the overall approach applied to ontologies can be understood as defining a semantics which is directly based on the abstract core level, or as one which combines the functions of an abstract core and an abstract top ontology [7].

3. Ontological Semantics

3.1. Ontological Structures and Ontological Semantics in General

A model theoretic semantics can abstractly be understood as a system $(L, M, \models)$ of a language L, a set of interpretation / model structures M, and a relation of satisfaction $\models \subseteq M \times L$, cf. [16, Ch. I.1, II.1]. We aim at a model theoretic approach for defining a formal semantics based on purely ontological entities. For this purpose, we establish a notion of *ontological structures* as an analogon to (mathematical) interpretation structures. These structures should avoid built-in ontological assumptions to the greatest possible extent. In order to achieve this and to draw an appropriate analogy to classical model theory, the set-theoretic background of model theory must first be explicated.

Consider a typical FOL-structure (restricting the signature to predicates and functional constants for simplicity) $\mathcal{A} = (A, R_1, \ldots, R_m, c_1, \ldots, c_n)$, where constants form elements of the logical universe A (a set), and predicates are interpreted as relations over A (as mathematical relations, i.e., as sets of tuples). This logical universe A does not cover all those entities which appear as interpretations of symbols in the formal semantics, in particular, it does not cover the interpretations of predicates. That means, relations over A are assumed silently based on standard set theories. The latter typically allow for constructing tuples and power sets over a given set, hence the structure $\mathcal{A}^{\mathcal{P}^{\omega}_{\text{fix}}} = \left(A^{\mathcal{P}^{\omega}_{\text{fix}}}, r_1, \ldots, r_m, c_1, \ldots, c_n\right)$ can be derived from $\mathcal{A}$, with $\mathcal{P}$ denoting the power set operator and $A^{\mathcal{P}^{\omega}_{\text{fix}}}$ being defined by:

$$A^{\mathcal{P}^{0}_{\text{fix}}} = A \tag{1}$$

$$A^{\mathcal{P}^{n}_{\text{fix}}} = \bigcup_{k<\omega} \left(\mathcal{P}\left(\left(A^{\mathcal{P}^{n-1}_{\text{fix}}}\right)^{k}\right)\right) \quad \text{for } n > 0 \tag{2}$$

$$A^{\mathcal{P}^{\omega}_{\text{fix}}} = \bigcup_{n<\omega} A^{\mathcal{P}^{n}_{\text{fix}}} \tag{3}$$

In $A^{\mathcal{P}^{\omega}_{\text{fix}}}$, all symbols of a logical language can be considered as constants, i.e., they are interpreted by elements of $A^{\mathcal{P}^{\omega}_{\text{fix}}}$. Some of them are in parallel subsets of (tuples over)

$A^{\mathcal{P}^{\omega}_{\text{fix}}}$. Hence, the elements of $A^{\mathcal{P}^{\omega}_{\text{fix}}}$ are interrelated according to the underlying set theory (i.e., set theory functions as an abstract top ontology here) – except for A-members. Only the elements c_i which interpret FOL constants are unconstrained by the set theory and may be related in arbitrary ways.

$\mathcal{A}^{\mathcal{P}^{\omega}_{\text{fix}}}$ appears as an adequate "template" for our intended ontological structures, in contrast to the classical structure $\mathcal{A}$. An ontological structure $\mathcal{O}$ is meant to provide constant-like interpretations for all symbols in terms of appropriate "members" of some "universe" O of $\mathcal{O}$. Technically, we say that those "members" are *associated with* O. No hidden assumptions are to be made on interrelations within O – if there are interrelations, they should be captured in an axiomatization using the symbols of the language. As *ontological* structures, neither O nor what is associated with O can generally be constrained to refer to mathematical notions. Altogether, this leads us to the following definition:

Definition 2 An ontological structure $\mathcal{O}$ can be described as $\mathcal{O} = (O, c_1, c_2, \ldots)$ where O is an arbitrary entity and the c_i are entities associated with O. □

Rendering O as an arbitrary entity may sound problematic. In other attempts to characterize O one may state that it is a "structure of intended semantics" (intended by the user of the language), a part of the world, or a state of affairs. In terms of situation theory [17], O corresponds closest to a "union" of *real* situations and events (but integrating individuals and categories). For a simplistic example, assume someone watching lion Leo in chasing some other animal. The observer may thus claim the existence of a corresponding part of the world O which would have associated with it the actual process / event, Leo, the categories of lion and chasing, Leo's participation in that process, etc. Referring to a single observer as well as to "reality" are problematic issues themselves, but cannot be discussed here. We just note that our view is more similar to that in [18, p. 13, Sect. 3.1, §2] which allows for a cognitive bias with respect to reality than to a purely objective and subject-independent view on reality. It must further account for entities in the widest sense, e.g. including hypothetical and fictitious entities, as well.

From the point of view of the c_i, O forms a kind of "aggregate" which comprises at least the c_i (and possibly further entities). The relation *associated with* must likewise be understood ontologically. In particular, we see this as a basic relation which generalizes e.g. part-of and inherence (which connects qualities with their bearers), and might include set membership. It is necessary that O offers counterparts for *all* basic syntactic entities of a language in the form of the c_i (see the treatment of FOL predicates below). Therefore, the c_i may be of arbitrary ontological kinds. All c_i and O coexist legitimately, without assuming reductions among them. For instance, we see no reductions among lion Leo, the category of lions, the chasing, and an ontological structure all of those entities are associated with. In addition, more entities may be associated with O than only the interpretations of the constants of a language. E.g. one can expect many, more or less detailed ontological structures comprising Leo. To emphasize the fact again, in general, O is not considered a set.

Definition 3 An *ontological semantics* for an arbitrary language L is a model theoretic semantics whose interpretation structures are ontological structures. □

This definition is clearly a very general characterization. For ontological structures as introduced in Def. 2 only constants can be interpreted immediately. In order to establish an ontological semantics for a declarative language L directly, more complex syntactic constructions must be assigned ontological interpretations in terms of those structures.

We indicate this direct approach for FOL in Sect. 3.2. Beforehand, note a difference in assigning an ontological semantics to a language L and common definitions of formal semantics. Given L in terms of a grammar G, G usually bottoms out with *identifiers*, i.e., symbols for which no further distinctions are made in the semantics. For example, in FOL, a non-terminal ‘predicate’ would have a range of admissible predicate-identifiers (terminals) assigned, but those do not influence the standard semantics of FOL. The latter is usually defined by non-terminal syntactic categories and a few fixed terminal symbols (or *keywords*), like ‘$\wedge$’ and ‘$\rightarrow$’. In contrast, ontological semantics / interpretations must ensure an appropriate interpretation for each single terminal in each particular use of a language. Since the number of non-terminal syntactic categories is limited, languages can be used very differently, which refers to their syntactic constructs and thus indirectly to the resulting formal semantic counterparts. It corresponds to our prior analysis that these distinct forms of using a language are only remotely dependent on classical formal semantics, and should be explicated by an ontological semantics.

3.2. Application to FOL Syntax

We sketch a definition of ontological semantics for FOL syntax, following classical definitions in a rather straightforward way in most cases, cf. [19, Sect. 2.2].[4] We assume appropriate valuations ν for variables in addition to an ontological structure $\mathcal{O}$ under consideration, where variables are assigned to entities associated with O. For a valuation ν, $\nu[\frac{x}{e}]$ refers to any valuation which agrees with ν on all assignments, yet only that of x is e in $\nu[\frac{x}{e}]$. $\mathcal{O} \models_\nu \phi$ means that $\mathcal{O}$ satisfies a formula ϕ for ν, $\mathcal{O} \models \phi$ that $\mathcal{O} \models_\nu \phi$ for every ν.

FOL logical constants do not have entities associated with O as semantic counterparts, corresponding to the case of set-theoretic interpretations. Rather, they manipulate or determine the combination of structures based on expressions and sub-expressions. They are defined for ontological structures in strict analogy to the standard definitions, e.g. for conjunction and negation:

$$\mathcal{O} \models \neg\alpha \ \text{ iff } \ \mathcal{O} \not\models \alpha \,. \tag{4}$$

$$\mathcal{O} \models \alpha \wedge \beta \ \text{ iff } \ \mathcal{O} \models \alpha \text{ and } \mathcal{O} \models \beta \,. \tag{5}$$

For quantification there are several options. Here, we adopt a variant that is equivalent with the classical definition and maintains the duality between existential and universal quantification.

$$\mathcal{O} \models_\nu \exists x \,.\, \phi \ \text{ iff } \ \text{there is an } e \text{ associated with } O \text{ s.t. } \mathcal{O} \models_{\nu[\frac{x}{e}]} \phi \,. \tag{6}$$

$$\mathcal{O} \models_\nu \forall x \,.\, \phi \ \text{ iff } \ \mathcal{O} \models_{\nu[\frac{x}{e}]} \phi \text{ for every } e \text{ associated with } O \,. \tag{7}$$

Notably and despite of adopting the same definitions, the nature of quantification changes considerably due to O being the domain of quantification (cf. the relations of A and $A^{\mathcal{P}^\omega_{\text{fix}}}$ above). Further definitions, e.g. of the validity of formulas regarding a structure, of logical validity, etc. also strictly follow their set-theoretic equivalents.

[4] Here we neglect epistemological issues as well as discussions on truthmakers, different degrees of conviction like beliefs, assumptions, truths, etc. Currently, we assume as a simplification only that all formulas share a common degree of conviction.

The major difference to the set-theoretic approach refers to predication. Classical semantics provides a uniform account in terms of set-membership: $P(x_1, \ldots, x_n)$ is true in interpretation I iff $(x_1^I, \ldots, x_n^I) \in P^I$. It is hard to provide a uniform ontological account of (syntactic) predication, because an arbitrary ontological interconnection among the arguments may be chosen for each individual predicate. For example, an atomic sentence $\mathsf{part\text{-}of}(x, y)$ may be read semantically as "x is a part of y" (intensionally, in contrast to $(x^I, y^I) \in \mathsf{part\text{-}of}^I$ which reduces "part of" to an extensional set of argument tuples). This would require a semantic condition like:

$$\mathsf{part\text{-}of}(x, y) \quad \text{iff} \quad \text{“}x \text{ is a part of } y\text{”}. \tag{8}$$

It is clearly undesirable to introduce every predicate in terms of informal phrases. Moreover, such definition would not profit from the requirement for an ontological semantics for a syntax with $\mathsf{part\text{-}of}(x, y)$ to contain an entity for the symbol part-of in its ontological structures. In this connection abstract core ontologies (ACOs) become relevant. An ACO should comprise a few basic entities and should be capable of classifying arbitrary entities (at a very abstract level). With sufficiently rich logical connectives, this allows for formally defining common predication patterns with respect to arbitrary entities (see Sect. 4 for sample patterns based on our proposed abstract core ontology). But before an actual formal proposal is to be made, let us consider to what extent classical interpretations can be "reused".

3.3. A Formalization Method based on Approximations of Ontological Models of FOL

The direct approach to defining an ontological semantics is not favorable due to the weak theoretical basis of such semantics. A better approach, especially for FOL, would be to build on classical theoretical results and to use established theorem provers for reasoning over ontological interpretations. Therefore, a major question concerns the relationship of ontological and set-theoretic interpretations, and whether the latter may be used for simulating or approximating ontological structures. For the present discussion we avoid interferences among the two types of semantics by restricting every ontological structure $\mathcal{O} = (O, c_1, c_2, \ldots)$ such that neither sets nor representational entities / syntactic elements like symbols are associated with O. Otherwise, one would have to take e.g. relationships between fore- and background membership relations into account.

Starting from an ontological structure $\mathcal{O}$ with its "universe" O, our approximation is initiated by an algebraic structure $\mathcal{A}(\mathcal{O}) = (U(O), c_1, c_2, \ldots)$ where $U(O) = \{x \mid x \text{ is associated with } O\}$ is a set of urelements (entities which are not sets). Consequently, every $c_i \in U(O)$ is the very same entity as considered ontologically (which connects to the relation between A and $A^{\mathcal{P}^{\omega}_{\text{fix}}}$ in Sect. 3.1). Next, we enrich these structures such that FOL formulas under a set-theoretic interpretation can be given an ontological interpretation, as well. Based on Sect. 3.2 the syntactic constructions which concern logical constants and quantification are directly transferable. It remains to accommodate the semantics of predication, now continuing the argumentation of Sect. 3.2. We aim at linking predication with intensions (via constants for the c_i) by explicit definitions:

$$\forall \bar{x} \,.\, P(\bar{x}) \leftrightarrow \phi(\bar{x}) \,. \tag{9}$$

Above we have indicated the potential diversity of interpreting (syntactic) predication ontologically, and we have argued that abstract core ontologies can be used to "bootstrap" such definitions. This leads us to the following method of formalizing ontologies.

Formalization Method. For an abstract core ontology Ω_{base}, we introduce a basic signature Σ_{base} for expressing relations of Ω_{base} by predicate symbols. Moreover, interconnections within Ω_{base} are specified axiomatically in a theory $Ax_{base} \subseteq L(\Sigma_{base})$. The major guideline of the method is to represent entities of *every* ontological kind, e.g. including categories and (ontological) relations, first as a functional constant in FOL. The introduction of new predicates (beyond Σ_{base}) must then be accompanied by a definition which involves previously introduced syntactic elements, most reasonably those functional constants which are understood to represent the intension of a predicate. Moreover, the new symbol(s) for representing an entity can be characterized axiomatically. □

FOL theories resulting from this method and their set-theoretic models can easily be related to an ontological semantics and ontological models. The main idea is that variables, functional constants and the Σ_{base} predicates are interpreted intensionally (or ontologically), whereas all other predicates are conceived extensionally, e.g., for $P \notin \Sigma_{base}$ the expression $P(x)$ is ontologically interpreted as $x \in P^{\in}$, which is adequate from a classical and an ontological point of view.[5] Moreover, an intensional specification of each predicate is available due to the required definitions and possibly additional axioms. Those definitions must ultimately rely on Σ_{base}-predicates, i.e., some intensionally interpreted relations from an abstract core ontology. Illustrations of the approach are presented in Sect. 4.

3.4. Ontological Usage Schemes

In terms of the approximation proposed for FOL, we can define a translational variant for ontological semantics.

Definition 4 Let L be an arbitrary language, and let Ω be an ontology for L with a FOL representation $R(\Omega) \subseteq L_\Omega$, $L(\Sigma_{base}) \subseteq L_\Omega$ according to the formalization method. An *ontological usage scheme* of L is a translation $\tau : L \to L_\Omega$.
For a set $S \subseteq L$ of L-expressions, its *ontological image* is the deductive closure of $R(\Omega) \cup \{\tau(s) \mid s \in S\}$. □

Ontological usage schemes have the intended advantage compared to the direct approach of defining an ontological semantics: one can rely on theoretical results established against the background of classical semantics, as well as on corresponding theorem proving algorithms and software. Nevertheless, the resulting classical models remain approximations of possible ontological models. In particular, given the exclusion of sets and symbols, there are far less ontological models of a theory T than there are set-theoretic approximations, because for the latter e.g. distinct, but isomorphic structures are also models of T. The existence of ontological models based on the existence of classical models must thus be justified individually. Moreover, the relation between set theory (as an ontology of sets) and the remaining ontological theory must be clarified. Altogether we think that a careful reuse of existing work clearly outweighs those approximation effects.

[5] There is one ontological concession, namely to grant sets an ontological status. This is applicable in GFO.

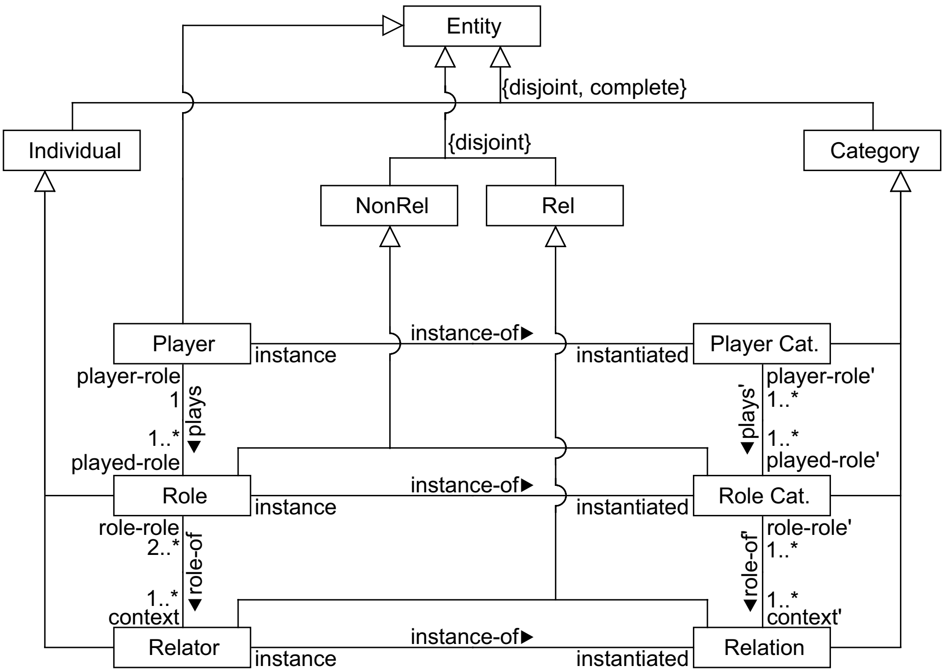


Figure 1. Overview of the ontology of categories and relations. The upper part shows two major distinctions of Entity, the first into categories and individuals based on instantiation, the second into relational (Rel) and non-relational (NonRel) entities. The lower part illustrates major categories relevant for relations, and their mediation between entities (note that Player and PlayerCat are extensionally equivalent with Entity and Category, respectively). Primed relations originate from their counterparts by lifting those to the level of categories, whereas primed roles are roles of primed relations.

4. Ontology of Categories and Relations

As a suitable abstract core ontology for formalizing top-level ontologies, we advocate an ontology of categories and (ontological) relations based on [20,21,7,2], among others. Fig. 1 outlines its major constituents in UML notation [4]. The most general notion for anything which exists is entity in GFO. Categories are those entities which can be instantiated (instance-of), in contrast to individuals. For example, a particular lion leo is an individual whereas $lion$ is a category[6]. Relations are granted an ontological status as categories of relators, specific entities which mediate between other entities. Relators are composed of (relational) roles[7], which appear like parts of a relator (role-of) and which “link” it to one of its arguments, cf. [20]. Roles are doubly dependent entities. Firstly, an entity plays a role (plays), which is a dependence on that player. Secondly, a role depends on other roles appearing in a relator, which must consist of at least two roles. Fig. 2 illustrates a part-of relator mediating between lion Leo and its head.

[6]We use the term “category” in accordance with GFO for anything which can be instantiated or predicated of something else. This is a much less specific use than the common philosophical reading of category as “highest kinds or genera” [22].

[7]This notion of roles differs from the notion of roles in description logics. The latter would typically correspond to relators connecting two arguments, hence being composed of two roles in the present sense.

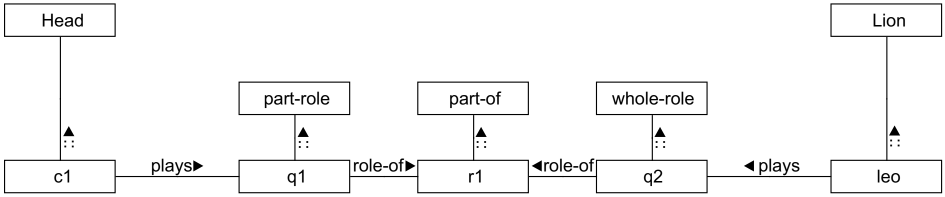


Figure 2. An example of relation analysis in GFO based on a phrase like "Lion Leo's head is a part of Leo". The relation `part-of` is a category of relators, instantiated (::) by `r1`. The relator `r1` mediates between Leo's head `c1` and `leo` via its two roles `q1` and `q2`, determined by the given role categories as roles of part and whole, respectively.

This ontology is adopted due to (a) the fundamental nature of categorization and (b) the possibility of relating two entities which appears as soon as they are distinguished from each other. The constituents of the approach further suffice to analyze themselves. Moreover, we see similarities with the most abstract levels in meta-modeling approaches which reinforces this position, cf. the root diagram of the Kernel package in the UML Superstructure Specification [23, fig. 7.3, p. 25].

The above formalization method can now be illustrated in connection with the ontology. Let $\Sigma_{base} = (::, \rightsquigarrow, \multimap)$ comprise only three binary predicates for our basic relations: $x :: y$ for "x instantiates y", $x \rightsquigarrow y$ for "x plays role y", and $x \multimap y$ meaning "x is a role of y". To distinguish symbols from their denotations, P^ι refers to the intended denotation of a symbol P. If P^ι has an extension, i.e., a set of instances or argument tuples, this is denoted by $P^\in$.

We introduce exemplary definition patterns which capture common types of predication, starting with the unary case. Unary predicates typically refer to non-relational categories. Following our method we introduce a functional constant ${}_cP$ and a unary predicate P. The following definition is added to bind P to its counterpart ${}_cP$.

$$\forall x \,.\, P(x) \leftrightarrow x :: {}_cP \,. \tag{10}$$

Classically, a FOL model interprets P and :: as sets over some universe which has an element interpreting ${}_cP$. The ontological interpretation of this formula (or of a corresponding classical model) is that ${}_cP$ captures an intension ${}_cP^\iota$ directly, and – as (10) states – P captures ${}_cP^\in$, the extension of ${}_cP^\iota$. Apart from this definition, appropriate axioms involving ${}_cP^\iota$ via P or ${}_cP$ should be stated.

For relations, there are several options of how n-ary predicates, $n \geq 2$, can be understood to abbreviate the linking of arguments via relators. A weak form (for a binary predicate R) is "there is a relator r which is an instance of the relation ${}_cR$ with respect to which x and y play roles in r":

$$\begin{aligned} \forall xy \,.\, R(x,y) \leftrightarrow \exists r q_1 q_2 (& \\ & q_1 \neq q_2 \wedge r :: {}_cR \wedge \\ & x \rightsquigarrow q_1 \wedge q_1 \multimap r \wedge \\ & y \rightsquigarrow q_2 \wedge q_2 \multimap r \quad) \,. \end{aligned} \tag{11}$$

This form entails symmetry of R, which may be counterintuitive for ${}_cR^\iota$. It can be strengthened to specify the instantiated role categories, cf. also Fig. 2. Moreover, assume that R is based on two intensionally distinct role categories ($Q_1 \neq Q_2$), each with exactly one role individual per R-relator, following the "closed world" intuition that a tuple (x, y) contains exactly x and y. This case concludes our sample patterns.

$$\begin{aligned}\forall xy \,.\, R(x,y) \leftrightarrow \exists r q_1 q_2 Q_1 Q_2 (& \\ & q_1 \neq q_2 \wedge Q_1 \neq Q_2 \wedge r :: {}_{c}R \wedge \\ & x \rightsquigarrow q_1 \wedge q_1 \multimap r \wedge q_1 :: Q_1 \wedge \\ & y \rightsquigarrow q_2 \wedge q_2 \multimap r \wedge q_2 :: Q_2 \wedge \\ & \forall q' (q' \multimap r \rightarrow (q' = q_1 \vee q' = q_2)\,)\ )\,.\end{aligned} \tag{12}$$

5. Discussion

5.1. Reconsideration of the Motivating Problems

The overall purpose of this paper is to propose a theoretical foundation for explaining "meaning-preserving" translations among languages, which maintain the declarative contents among the different expressions of those languages. This differs from simulations among the dynamics of the languages, e.g. encodings of reasoning problems of one logic into another. One may argue that logical formalisms can be used with intensional interpretations independently of or in addition to their set-theoretic model theory. Even if this is case, those intensional interpretations remain implicit and thus cannot be used e.g. for translations among languages.

The general approach to ontological semantics clearly adopts ontological entities as its foundation and thus avoids Problem 1, the circular interplay between ontologies and formal language semantics based on mathematical notions. The same applies indirectly to ontological usage schemes and ontological images of arbitrary expressions, i.e., translations of those expressions into FOL formalizations constructed according to the presented method. Classical models of ontological images are potential approximations which can help in determining ontological models. Moreover, ontological images comprise explicit ontological explanations for each predicate, which binds their extensional interpretation to complex expressions with intensional components (through the intensionally understood functional constants and basic relations). This suggests the need for refining Def. 1 of conceptually equivalent expressions in our initial analysis. For instance, two predicates are conceptually equivalent by Def. 1 if they originate from two intensionally distinct, but extensionally equivalent categories. Due to the formalization method, a suitable refinement of that definition can be based on the identity of functional constants and of compositions via basic relations. A related aspect is the dependence of Def. 1 on the chosen logic. For example, assume that an ontology is represented (i) in a monotonic logical language and (ii) a nonmonotonic language. In its present form, there will be immediate differences in the resulting notions of conceptual equivalence. We see a need for further investigations in this respect, which may involve different types of categories.

Allowedly, the formalization method is rather simple and could be adopted on an *ad hoc* basis. Readers who share our analysis and / or who take a purely proof-theoretic point of view may thus miss benefits of the approach, e.g. computational ones. Conceptually, we believe that the theory will prove useful in some respects. Deriving a novel definition for conceptual equivalence is one candidate for this. Another is the provision of justifications or rejections of certain formalizations. To name an example, if one were to formalize a category *entity* as the category which classifies everything (including sets), a predicate Entity could not be introduced meaningfully in line with formula (10) since this would contradict the well-foundedness of standard set theories.

The second problem of providing an abstract core ontology has been addressed in the previous section by outlining an ontology of categories and relations. This is a proposal of one potentially suitable ontology rather than its "unique solution", and different such proposals should be compared and evaluated. In general, ontological semantics leads to the fact that comparisons of two ontology representations $R(\Omega_1)$ and $R(\Omega_2)$ with an ontological semantics must determine one of the compared systems as a point of reference. Considering the use of a third ontology $R(\Omega_{ref})$ in this respect does not differ significantly, because then embeddings of the $R(\Omega_i)$ into $R(\Omega_{ref})$ are required – which is a case of the first kind.

Another important aspect of the general approach is its non-reductionism. In particular, we consider everything in an ontological model to be on a par with each other. The use of an abstract core ontology as a means to *analyze* entities and to initiate formalizations is not to be understood as a *reduction* to notions in the abstract core ontology, neither for its categories nor relations. Metaphorically, it is not sufficient to think of leo as an individual (at the abstract core level), nor as a lion (at a domain level), but leo is only fully recognized as leo, and is analyzable and related to other entities.

5.2. Related Work

There is an overwhelming amount of broadly related work, e.g. meta-modeling in conceptual modeling, works based on situation theory [17,24] and information flow theory [25,26] as well as approaches to intensional logics like Montague's, Tichý's, cf. [27], and George Bealer's [28]. From the perspectives of these fields, our approach originates from "practical" concerns in representing foundational ontologies like GFO, whereas establishing detailed connections to them is an ongoing effort. Situation theory in its origins is currently the most promising candidate regarding a tight linkage and particular aspects of its motivations. Nevertheless, basic differences remain there, as well, e.g., a built-in "ontology" of primitives (individuals, relations, space-time locations) plus situations and their construction from these primitives, and a set-theoretic metatheory.

Concerning knowledge representation, we focus on the approach of *Ontology-Based Semantics* by Ciocoiu and Nau [5]. It shares its motivation and goals with ours, and we agree with most of the analysis in its Sections 1 and 2, leading to the use of an ontology as a common semantic foundation. However, Problem 1 (circularity) is not identified in [5], but a classical FOL representation $R(\Omega) \subseteq \mathcal{L}_\Omega$ of the ontology is used. For defining ontology-based models of a language L, the authors use a two-step translation, (i) from L into a FOL language $\mathcal{L}$, and (ii) an interpretation in the sense of [19, Sect. 2.7] from $\mathcal{L}$ into $\mathcal{L}_\Omega$. We have reservations about both steps. For (i), this requires an encoding of ontological notions into set theory, which may differ for the same ontological notions contained in distinct languages. Some of these encodings cannot be unified in the second step, because those interpretations maintain the number of free variables in interpreted predicates, which prevents switching from a constant $lion$ to a predicate $Lion$, for instance. On the other hand, (ii) may allow for too strong encodings in other respects.

Finally, note that recent language proposals (e.g. Common Logic [29]; [30] in description logics) relate to our approach. They allow for syntactic expressions which seemingly require a classical higher order semantics, like in the theory $\{P(R), R(x,y)\}$. These languages have a non-standard set-theoretic semantics with parallels to our FOL approximations. It is promising to use the syntax of these languages with an ontological semantics, or to build approximations due to their semantics. However, the method of linking all predicates (more generally, composed syntactic expressions) to "intensional specifications" is not enforced elsewhere. It should be added since it is of major relevance for appropriate definitions of conceptual equivalence.

6. Conclusions

In this paper we have argued that it is insufficient to rely on conventional formal semantics when representing ontologies, due to their foundational role with respect to semantics. We proposed a new type of model theoretic semantics called *ontological semantics* together with an approximation for FOL syntax in order to utilize existing work, resulting in an additional, translation-based definition. Complementarily, we have outlined an ontology of categories and relations as one meta-ontological option for applying this semantics to the formalization of ontologies.

Essentially, our approach for formalizing ontologies should add a level of explanation to the encoding of ontological into abstract mathematical notions. We expect that this leads to "meaning-preserving translations" which may appear conceptually more adequate than conventional formal reductions (which are undoubtedly very valuable in other respects). This should influence working with ontologies, in particular ontology matching and integration. Moreover, the resulting theories offer specific properties which may be exploited, e.g. for modularly structured ontologies.

Future work comprises further studies of the proposed structures and their approximations. Furthermore, the general approach requires to explore its application based on different logics and the use of different abstract core ontologies. Instead of unique final solutions, for all parameters we expect a plurality of options which should compete with respect to practical applications.

Acknowledgments

We are grateful to the anonymous reviewers for helpful and inspiring criticism as well as pointers to further relevant literature. Moreover, we thank our colleagues Holger Andreas, Gerhard Brewka, Alexander Heußner, Robert Hoehndorf, and Alexander Nittka for lively discussions and valuable feedback on the topic.

References

[1] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[2] Heinrich Herre, Barbara Heller, Patryk Burek, Robert Hoehndorf, Frank Loebe, and Hannes Michalek. General Formal Ontology (GFO) [Version 1.0]. Onto-Med Report 8, IMISE, University of Leipzig, Germany, 2006.

[3] Barbara Heller and Heinrich Herre. Ontological categories in GOL. *Axiomathes*, 14(1):57–76, 2004.

[4] James Rumbaugh, Ivar Jacobson, and Grady Booch. *The Unified Modeling Language Reference Manual*. Addison Wesley, 2nd edition, 2005.

[5] Mihai Ciocoiu and Dana S. Nau. Ontology-based semantics. In *Principles of Knowledge Representation and Reasoning: Proceedings of KR-2000*, pages 539–546. Morgan Kaufmann, 2000.

[6] Sergei Nirenburg and Victor Raskin. *Ontological Semantics*. MIT Press, 2004.

[7] Heinrich Herre and Frank Loebe. A meta-ontological architecture for foundational ontologies. In *On the Move to Meaningful Internet Systems 2005, (Part II): Proceedings of ODBASE 2005*, volume 3761 of *LNCS*, pages 1398–1415. Springer, 2005.

[8] Asunción Gómez-Pérez, Mariano Fernández-López, and Oscar Corcho. *Ontological Engineering*. Springer, 2004.

[9] Peter Patel-Schneider, Patrick J. Hayes, and Ian Horrocks. OWL Web Ontology Language semantics and abstract syntax. W3C Recommendation, W3C, 2004.

[10] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, 2003.

[11] Natasha Noy and Alan Rector. Defining N-ary relations on the Semantic Web. W3C working group note, W3C, 2006. http://www.w3.org/TR/2006/NOTE-swbp-n-aryRelations-20060412/.

[12] Marc Denecker. What's in a model? Epistemological analysis of logic programming. In *Principles of Knowledge Representation and Reasoning: Proceedings of KR-2004*, pages 106–113. AAAI Press, 2004.

[13] Jon Doyle, editor. Special issue on implemented knowledge representation and reasoning systems. *SIGART Bulletin* 2(3), 1991.

[14] Robert Neches, Richard Fikes, Tim Finin, Thomas Gruber, Ramesh Patil, Ted Senator, and William R. Swartout. Enabling technology for knowledge sharing. *AI Magazine*, 12(3):36–56, 1991.

[15] John F. Sowa. *Knowledge Representation*. Brooks/Cole, 2000.

[16] Jon Barwise and Solomon Feferman, editors. *Model-Theoretic Logics*. Springer, 1985.

[17] Jon Barwise and John Perry. *Situations and Attitudes*. MIT Press, 1983.

[18] Claudio Masolo, Stefano Borgo, Aldo Gangemi, Nicola Guarino, and Alessandro Oltramari. WonderWeb Deliverable D18: Ontology library (final). Technical report, Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy, 2003.

[19] Herbert B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, 1972.

[20] Frank Loebe. Abstract vs. social roles – Towards a general theoretical account of roles. *Applied Ontology*, 2(2):127–158, 2007.

[21] Frank Loebe. An analysis of roles: Towards ontology-based modelling. Onto-Med Report 6, IMISE, University of Leipzig, Germany, 2003.

[22] Amie Thomasson. Categories. In *The Stanford Encyclopedia of Philosophy*. CSLI, Stanford University, Fall 2004 edition, 2004. http://plato.stanford.edu/archives/fall2004/entries/categories/.

[23] OMG. Unified Modeling Language: Superstructure. Specification v2.1.2, OMG, 2007.

[24] Keith Devlin. Situation theory and situation semantics. In *Logic and the Modalities in the Twentieth Century*, volume 7 of *Handbook of the History of Logic*, pages 601–664. Elsevier, 2006.

[25] Robert E. Kent. Semantic integration in the Information Flow Framework. In *Semantic Interoperability and Integration*, number 04391 in Dagstuhl Seminar Proceedings. IBFI, Schloss Dagstuhl, Germany, 2005.

[26] W. Marco Schorlemmer, Yannis Kalfoglou, and Manuel Atencia. A formal foundation for ontology-alignment interaction models. *International Journal on Semantic Web and Information Systems*, 3(2):50–68, 2007.

[27] Barbara Partee and Herman L. W. Hendriks. Montague grammar. In *Handbook of Logic and Language*, chapter 1, pages 5–92. MIT Press, 1997.

[28] George Bealer. *Quality and Concept*. Clarendon Press, 1982.

[29] ISO. Information technology — Common Logic (CL): a framework for a family of logic-based languages. International Standard 24707, ISO/IEC, 2007.

[30] Boris Motik. On the properties of metamodeling in OWL. *Journal of Logic and Computation*, 17(4):617–637, 2007.

2. Ontological Engineering

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-65

Cognitive Context and Arguments from Ontologies for Learning

Christiana PANAYIOTOU [a,1], Brandon BENNETT [a]
[a] *School of Computing, University of Leeds, LS2 ,Leeds,UK*

Abstract. The deployment of learning resources on the web by different experts has resulted in the accessibility of multiple viewpoints about the same topics. In this work we assume that learning resources are underpinned by ontologies. Different formalizations of domains may result from different contexts, different use of terminology, incomplete knowledge or conflicting knowledge. We define the notion of *cognitive learning context* which describes the cognitive context of an agent who refers to multiple and possibly inconsistent ontologies to determine the truth of a proposition. In particular we describe the cognitive states of *ambiguity* and *inconsistency* resulting from incomplete and conflicting ontologies respectively. Conflicts between ontologies can be identified through the derivation of conflicting arguments about a particular point of view. Arguments can be used to detect inconsistencies between ontologies. They can also be used in a dialogue between a human learner and a software tutor in order to enable the learner to justify her views and detect inconsistencies between her beliefs and the tutor's own. Two types of arguments are discussed, namely: arguments inferred directly from taxonomic relations between concepts, and arguments about the necessary and jointly sufficient features that define concepts.

Keywords. ontologies, reasoning, formal comparison between ontologies

Introduction

Learning resources are becoming increasingly available on the web. As a result a learner may have access to multiple resources about a single topic. We assume that each learning resource is underpinned by an ontology. Ontologies of the same domain may be represented at various degrees of abstraction and granularity. They may also represent knowledge at different degrees of completeness. Reasons can be traced to different points of view and experience of the experts that derive them. The learner may not be able to determine whether discrepancies in ontologies arise due to incompleteness of knowledge, due to disagreement between ontologies, or due to differences in the perspectives giving rise to different viewpoints. Our long term objective is to develop a computational framework of an agent capable of handling viewpoint discrepancies in the ontologies of learning resources and to enable a learner to engage in a dialogue with the software tutor to clarify differences of her own viewpoints with the viewpoints of learning resources.

[1]Corresponding Authors: Christiana Panayiotou, Dr. Brandon Bennett, School of Computing, University of Leeds, LS2 Leeds, UK; E-mail: cpaa, brandon@comp.leeds.ac.uk

This paper focuses on the formalization of three important aspects of this framework, described below.

Firstly we formalize two cognitive states, namely the cognitive states of ambiguity and inconsistency that enable us to plan the interaction between a human learner and the software agent. In order to address the problem of cognitive ambiguity and confusion of learners, we allow resources with conflicting or different information to be part of the same cognitive context. We assume that the context is related to the goal of the learning activity (referred to as the *focus* of the learning activity) rather than on the compatibility of the resources referred to by the context. As a consequence, the context may involve multiple domains, if multiple domain points of view are relevant to the learning topic. For example, the topic may involve the points of view of multiple domains like psychology, social science and anthropology in order to form a particular position.

Secondly, we propose a proof-theoretic approach to the automatic derivation of arguments from ontologies. To resolve cognitive confusion arising from inconsistencies in ontologies, we suggest the use of reasoning via argumentation. A theorem prover can be used to check consistency of arguments of one ontology with ontologies of other resources when arguments are translated into an appropriate form. It can also be used in human computer interaction to enable the learner and the tutor to clarify their positions about a topic via arguments. The software agent can also verify the validity (soundness) of a learner's argument from its form. We formalize two different types of arguments that are useful in learning. These are syllogistic arguments derived from hierarchical relations in ontologies and arguments about necessary and jointly sufficient features of concepts.

Thirdly, we suggest a set of utterances that enable a learner and a tutor to exchange arguments in a human computer interaction and check the validity of the learner's arguments. In order to facilitate human-computer interaction, utterances between agents are represented internally as dialogue moves. Each move may include an ontological statement of a particular resource and may cause a change on the beliefs (ontologies) of the participants of the dialogue. Also as the focus of the interaction may change during the dialogue, the set of ontologies associated with the cognitive learning context may change as well. In order to capture this dynamic behavior of the system we make the learning context of the participants and the belief stores of the participants of the dialogue situation depended and we formalize changes via the use of situation calculus. Our third objective in this paper is to formalize a set of moves that enable the exchange of arguments inferred from particular ontologies.

The rest of this paper is outlined as follows. Section 1.2 reviews related work on the definition of context and on paraconsistent logics. The notions of cognitive learning context, cognitive ambiguity and cognitive inconsistency are discussed in this section. In section 3 we discuss syllogistic arguments and arguments related to the necessary and jointly sufficient properties of concepts. Section 4 shows an example of an interaction of a learner with the software agent in order to discuss differences in ontologies of underlying resources. Finally section 5 outlines the main issues discussed in this paper and briefly describes future research plans.

1. Related Work

1.1. Mental Spaces

Notable approaches to modeling cognitive context can be found in the linguistics literature. We note the works of Fauconnier [1] and Dinsmore[2]. Fauconnier [1] advocated the idea of *mental spaces*, which are described as constructs build up in any discourse according to guidelines provided by linguistic expressions. Objects in mental spaces are treated as real objects independently of their status in the actual world. Mental spaces can be built up from many sources and domains. Examples, of sources are: the immediate experience of the agent, what other people say to us, or what other people think, etc. They are created by *Space Builders* which are particular words triggering the creation of a new space. For example, one such space builder is the word *maybe* used to build the *possibility* mental space. A *base mental space* is the mental space in which the discourse takes place. Other important notions introduced in the literature of mental spaces are: the notion of ambiguity arising from multiple connecting paths between partitioned configurations that yield multiple understandings and the requirement of compatibility between mental spaces. Fauconnier's work aimed to address the problems of referential opacity, the projection of presuppositions, the semantic processing of counterfactuals, etc [3].

Dinsmore [2] complemented Fauconnier's theory by focusing on the external structure of mental spaces and attributed semantics to them so that they can be used in representing and reasoning about knowledge. He introduced the notion of knowledge partitioning as the process of distributing knowledge along different spaces according to context. The context of a space is a propositional function, e.g. the context of Mary's space is a function that takes a proposition p and maps it onto the proposition that Mary believes p. Dinsmore, showed that inheritance of information from one space in another is determined by the semantic properties of the respective contexts. Inheritance contexts constitute a form of secondary contexts, the latter being used to provide a mapping from the content of one space to the contents of another [3].

Fauconnier's contribution to modeling mental spaces that correspond to linguistic forms and words is important for representing the context of utterances and for referencing objects in different contexts. However, the focus of his investigation was the mental configurations resulting from english sentences and the construction of meaning during discourse. Instead, we focus on the epistemic state of agents who have access to incomplete resources. Several of the notions used in the representation of knowledge partitions by Dinsmore point to the artificial intelligence perspective of context representation and contextual reasoning [4,5,6,7]. Although they provide a significant insight to the problem of context representation and reasoning, their models do not capture the notions of incompleteness and inconsistency between different resources.

1.2. **Local reasoning with multiple epistemic alternatives**

The Local Model Semantics [8] provide a foundation for reasoning with contexts which is based on two main principles: the principle of *locality* and the principle of *compatibility*. The first states that reasoning requires only a part of what is potentially available [8]. The principle of compatibility states that there is compatibility among the kinds of reasoning performed in different contexts [8] and associates different contexts with some meaningful relation of subsets of their local models. Our notion of cognitive context is different from the above as it may include incompatible resources that are related to the reasoning task of the learning activity. However, the principle of locality and the assumption that the available information may be incomplete affect the way learners interpret information and are used to model the cognitive state of the learner.

Several logics addressed the problem of inconsistency in logic theories and knowledge bases. To name but a few, paraconsistent logics, many-valued logics and modal logics have been developed to tackle inconsistency. Among those, notable uses of paraconsistent and possible world semantics to model mental models and epistemic states are the works of [9] and [10]. Fagin and Halpern [9] consider each agent as a *society of minds* rather than a single mind. Inspired by the work of Fagin and Halpern [9], Lokhorst [11] developed a (two-valued) local reasoning model of split-patients as a structure:

$$M = \langle W, w_0, \Psi, S, R, V \rangle \tag{1}$$

where W denotes a set of possible worlds, w_0 the actual world, Ψ a set of minds (each mind behaves independently of the other), S a function $S : W \rightarrow \wp(\Psi)$ (S maps a world to the set of minds in which this world is possible) and R is a function from Ψ into $W \times W$. The above model had some utility in creating our cognitive learning context for the following reason. Suppose that we represent each mind in Ψ above, as a (local) ontology with its own signature. Then S would associate each (separate) ontology with the set of worlds with which this ontology is compatible. This would be useful if the learner was unable to compare information from different ontologies. Hence lack of comparison would mean lack of confusion caused from differences between ontologies. But this differs from the problem we are trying to solve. Therefore the above model cannot be applied as it is to model the cognitive learning context of a learner.

The paraconsistent logic LEI is based on the idea of multiple observers having diverging views about a certain state of affairs. It extends classical logic with the formula $p?$ where $p?$ is satisfied whenever p holds in all plausible worlds. Unlike the traditional modal logics approach to modeling necessity and possibility, the LEI employs two satisfaction relations: the credulous and the skeptical approach. Martins et al. [10] provided a multiple world semantics to the above idea where each plausible world corresponds to a particular view of the world. The above approach is useful in comparing beliefs derived by the credulous vs. skeptical entailment relation which is different from the focus of this paper. In this paper we assume that each agent combines two levels of reasoning: a local reasoning level which considers each ontology locally and the meta-epistemic level, at which the agent compares inferences drawn locally in each ontology and determines compatibility with other ontologies.

2. Cognitive Context, Incompleteness and Inconsistency

We illustrate the notion of incompleteness and inconsistency in resources via the use of an example. Then we introduce our proposed definitions for the concepts of cognitive context, ambiguity due to incompleteness and inconsistency using the possible world semantics.

2.1. Example

A learner L comes across a professional training programming course on visual basic. This resource states that *visual Basic is an object-oriented language*. The learner believes that an object-oriented language needs to satisfy the property of encapsulation but she does not know whether visual basic has this property. In addition the online notes of her class instructor show visual basic as an example of a non 'object oriented language' because it does not have the property of inheritance.

The learner in this example makes use of three resources - her own background knowledge about object oriented languages, the instructor's online notes and the professional programming course site. All of them are part of the cognitive context of the learner. Two of these resources, namely the instructor's notes and the professional programming course are inconsistent. Although the learner's background knowledge is not directly inconsistent with the online resources, it is not reinforced by them either. Since both online resources are assumed to be expert resources the learner does not know how to interpret lack of evidence supporting her own knowledge. Do the experts possess partial knowledge about significant concepts of the domain? Should relevant information from both resources be integrated or should one be dropped for another? Since there is no definite answer for all situations and more than one interpretations of the situation are possible, we interpret the epistemic state resulting from this situation as ambiguous. In the following paragraphs of this section we formalize the notions referred to in the above example. In this paper we represent each learning resource via its underlying ontology. So a definition of ontology is relevant.

2.2. Ontology

In this project we use *OWL-DL* as an ontology representation language because it is a decidable fragment of description logic and expressive enough to satisfy our need for the representation of concepts, roles and hierarchies that give rise to the type of arguments formalized in this work. An Ontology in this paper is described as a structure $\langle T, A\rangle$ where T denotes a DL TBox (i.e. a set of terminological) axioms and A denotes a DL ABox (i.e. a set of grounded assertions). Each ontology has its own signature consisting of a disjoint set of relation names, concept names and constant names of individuals. We denote the signature of an OWL ontology O_i by $Sig(O_i) \equiv R \cup C \cup N$, where R denotes the relation names, C the concept names and N the set of individual names. The interpretation I_i of the $Sig(O_i)$ is the structure $\langle D_i, \cdot^{I_i}\rangle$ where D_i is the domain of the ontology and $\cdot^{I_i}$ is the interpretation function such that: $C^{I_i} \subseteq D_i$, $R^{I_i} \subseteq {D_i}^n$ (in OWL is $D_i \times D_i$).

2.3. Cognitive Learning Context

The model of the local reasoning learning context of a learner L is defined as a structure

$$\Upsilon_{sit} \equiv \langle O, W, \delta, \eta, s\rangle \tag{2}$$

where $O = \{O_1^{'}, \dots O_n^{'}\}$ and $O_i^{'} \equiv \langle T_i^{'}, A_i^{'}\rangle$ represents the part of each ontology $O_i \equiv \langle T_i, A_i\rangle$ referenced that is relevant to the *focus*, η, of the learning activity, i.e. $T_i^{'} \subseteq T_i$ and $A_i^{'} \subseteq A_i$. Each ontology O_i has a standard interpretation $I_i = \langle \Delta_i, \cdot^{I_i}\rangle$. Let $T =$

$T_1 \cup \ldots \cup T_n$ and $A = A_1 \cup \ldots \cup A_n$. Let I_i^* be an extension (interpretation) of I_i on $T \cup A$. We define W to be the set of interpretations of $T \cup A$, i.e. $W = \{I_i^*\}_{i=1\ldots n}$ and δ to be an accessibility relation associating each $O_i^{'} \in O$ in each situation to a set of possible epistemic alternatives: $\delta : O \rightarrow \wp(W)$. η is a proposition.

Note that there may not be any interpretation satisfying all ontologies. If we assume that ontologies are locally consistent then there is at least one interpretation satisfying each ontology in O. For example, if $A \sqsubseteq B \in T_i$ and $A \sqsubseteq C \in T_j$ but $A \sqsubseteq C \notin T_i$ then there exist two subsets of possible worlds in W, W_1 and W_2 say, such that W_1 supports both $A \sqsubseteq B$ and $A \sqsubseteq C$ and W_2 supports $A \sqsubseteq B$ but not $A \sqsubseteq C$. Also, for each conflicting set of formulae $A \sqsubseteq B \in T_i$ and $A \sqsubseteq \neg B \in T_j$ for $i \neq j$, there is at least one possible world $w \in W$ which assigns true to one formula and false to the other. Using the above definition of the cognitive state of a learner, we are now able to discuss the cognitive states of ambiguity and inconsistency.

2.4. Cognitive Ambiguity due to Incompleteness

Intuitively, a learner reaches a cognitive state of ambiguity whenever she has access to more than one plausible epistemic alternatives and the learner is unable to choose one. The Oxford English Dictionary defines ambiguity as: *wavering of opinion, hesitation, doubt, uncertainty, as to one's course, or, capable of being understood in two or more ways, or, doubtful, questionable, indistinct, obscure, not clearly defined and lastly, admitting more than one interpretation or explanation; of double meaning or several possible meanings* (in [12]). The notion of ambiguity in our case refers to the interpretation of incompleteness of information contained in learning resources by the learner. We assume that a learner becomes aware of the incompleteness of a learning resource when she compares it with her background knowledge or with another resource.

2.4.1. Definition of Cognitive Ambiguity

Assume a resource R_1 and $\delta(R_1) = W_{R_1} \subseteq W$ i.e. R_1 is compatible with a subset of possible worlds W_{R_1} of W. Then, assume that the agent has access to another resource R_2 which is compatible with$W_{R_2} \subseteq W$. If there exist $w_1, w_2 \in W$ where $w_1 \in W_{R_1}$ and $w_2 \in W_{R_2}$ such that w_1 supports η and w_2 supports $\neg\eta$ then we say that agent A is ambiguous with respect to η and we denote this as: $U_A(\eta)$.

2.4.2. Vocabulary Assumption

The type of ambiguity we address here is the ambiguity that results from incompleteness of knowledge rather than the lexical vocabulary used by each resource. The set of resources relevant to the subject of the learning activity may change in each situation according to the focus of the learning activity. To be able to determine incompleteness and inconsistency between ontologies we need to make some assumptions regarding the vocabularies of the ontologies that form part of the cognitive context. Assume a unified signature Σ which consists of the union of all the signatures $Sig(O_i^{'})$ (defined as above). To simplify matters, we assume that any two identical non-logical symbols of two resources R_1 and R_2 are considered the same unless there is evidence to the contrary. Further, where we have explicit default mappings between terms we may apply default inference rules to draw conclusions between multiple ontologies as follows:

$$\frac{[R_1 : C(x)] : [R_2 : C(x)] \leftrightarrow [R_1 : C(x)]}{[R_2 : C(x)]} \tag{3}$$

Default rule 3 states that if there is no inference inconsistent to $[R_2 : C(x)] \leftrightarrow [R_1 : C(x)]$ in R_2 then $R_2 : C(x)$ can be asserted in R_2. A similar default inference rule is used for relations between concepts and names of individuals.

$$\frac{[R_1 : R(x,y)] : [R_2 : R(x,y)] \leftrightarrow [R_1 : R(x,y)]}{[R_2 : R(x,y)]} \tag{4}$$

The biconditional used in the inference rules aims to maintain consistency with mappings of terms between different vocabularies. If the conclusion of the default rule that refers to an assertion about a resource is not inconsistent with the assertions of the resource and is not already in the ontology of the resource, then the resource is incomplete.

As an example, of a case where direct equivalences of assumptions can be used to assert new facts about different resources consider two people (P_1 and P_2 say) viewing a scene from opposite sites then $P_1 : right(P1, x) \leftrightarrow P_2 : left(P2, x)$ for some object x. Further assume that the constrain $P_i : right(P1, x) \rightarrow \neg P_j : left(P2, x)$ where $i \neq j$ and $i, j \in \{1, 2\}$ holds for each person. Then obviously, it is inconsistent to assume that $P_1 : right(P_1, x) \leftrightarrow P_2 : right(P_2, x)$. Note that the intended meaning of the notions of $P_i : right(P_i, X)$ and $P_i : left(P_i, X)$ for each $i \in \{1, 2\}$ is independent of the situation of P_i. However the actual assignment of terms is dependent on their situation.

2.5. Cognitive Inconsistency (Confusion)

Intuitively, we assume that cognitive inconsistency arises when in the actual world of the learner, information about a topic is conflicting. It is different from cognitive ambiguity in that cognitive ambiguity appears as a consequence of possible epistemic alternatives (not necessarily inconsistent) due to lack of knowledge. We model this by the derivation of refuting arguments relating to the *focus* of the learning activity.

2.5.1. Definition of Inconsistency

Assume a resource R_1 and $\delta(R_1) = W_{R_1} \subseteq W$. Then, assume that either $\delta(R_2) = W_{R_2} \subseteq W$ for some resource R_2 (or that $\delta(BK) = W_{R_{BK}} \subseteq W$ where BK is the background knowledge of the agent). If for any two $w_1, w_2 \in W$ such that $w_1 \in W_{R_1}$ and $w_2 \in W_{R_2}$ (or $w_2 \in W_{R_{BK}}$) we have that w_1 supports η and w_2 supports $\neg\eta$ then we say that agent A is inconsistent with respect to η and we denote this as: $Inc_A(\eta)$.

The use of argumentation to identify and justify claims that may be conflicting each other is not only important for the recognition of the cognitive state of the learner but also for the recognition of differences or inconsistencies in ontologies automatically. In the next section we discuss the formalization of two types of arguments that can be inferred from ontologies, namely syllogistic and arguments about necessary and jointly sufficient features associated to the definition of concepts.

3. Syllogistic Arguments and Ontological Taxonomic Relations.

An Ontology may include one or more hierarchies of concepts that can be used to infer categorical statements.

3.1. Concept hierarchy

A *concept hierarchy* is a structure $\mathcal{H} = \langle C_{\mathcal{H}}, R_{\mathcal{H}} \rangle$ where $C_{\mathcal{H}}$ is a set of concepts, st. $C_{\mathcal{H}} \subseteq C$ of the ontology O, and $R_{\mathcal{H}} = \{Disjoint, SubclassOf, Union, Intersects\}$ and every concept in $C_{\mathcal{H}}$ is associated with another concept via a relation in $R_{\mathcal{H}}$. OWL-DL provides for all of relations in $R_{\mathcal{H}}$ and therefore a hierarchy can be represented in it. We are interested in those interpretations of a hierarchy that satisfy all the taxonomic relations within the hierarchy. A model, $\mathcal{M}_{\mathcal{H}}$ of $\mathcal{H}$ is an interpretation I of $\mathcal{H}$ where all the taxonomic relations in $R_{\mathcal{H}}$ are satisfied. Obviously, $\mathcal{M}_{\mathcal{H}}$ is a sub-model of $\mathcal{M}$ and therefore any entailment of $\mathcal{M}_{\mathcal{H}}$ is an entailment of $\mathcal{M}$.

3.2. Categorical statements

Generalized statements the form: *Every X is a Y* or *Every X has the property of Y* can be inferred from taxonomic hierarchies and can be combined to form *syllogistic arguments*. These statements are referred to as *categorical statements*. A syllogism [13] is a particular type of argument that has two premises and a single conclusion and all statements in it are categorical propositions.

3.2.1. Individuals

In ontologies, a distinction is made between individuals and classes. In the consequent we argue that the set equations that can be used to represent ontological primitives can be translated to propositional logic formulae that can be used to test validity of arguments. To simplify computation and to prove whether an individual belongs to a class (or a refutation that an individual belongs to a class) we represent individuals as singular sets consisting of that individual only. In this way we treat individuals as classes during inference. An ontology may include one or more hierarchies of concepts that can be used to infer syllogisms.

3.2.2. Syllogisms

Syllogisms form a particular type of arguments that are constructed from generalized statements (categorical statements). There are four basic categorical statements which can be combined to produce 64 patterns of Syllogistic Arguments. These are shown below together with the corresponding ontological primitives:

Categorical Statement	**Ontological Primitive**
Every S is a P	SubclassOf(S, P)
No S is a P	SubclassOf(S, ComplementOf(P))
Some S is a P	Intersects(S, P)
Some S is not P	Intersects(S, ComplementOf(P))

However, only 27 of them are valid syllogisms. This suggests the need to check the validity of syllogisms constructed from ontologies and exchanged during interaction with the learner.

3.3. Necessary and Sufficiency Conditions Arguments.

The classical view of the representation of concepts states that the features representing a concept are *singly necessary* and *jointly sufficient* to define a concept. In line with the above view we propose the following definitions for the *necessary* and *jointly sufficient* features representing a concept.

3.3.1. Necessary Features for the Representation of a Concept

Intuitively, a feature ϕ is *singly necessary* for the definition of C if and only if existence of C implies existence of ϕ. Assume a feature ϕ. We define a set Φ consisting of all individuals of the domain which have property ϕ (e.g. via the onProperty restriction in OWL-DL). Then, ϕ is a necessary property for the representation of concept C if and only if $C^I \subseteq \Phi$. An example of a refutal to the assumption that ϕ is a necessary feature for C is the derivation of an individual that belongs to C and to a class disjoint with Φ.

3.3.2. Jointly Sufficient Features for the Representation of a Concept

Let $\{\Phi_1, \ldots, \Phi_n\}$ represent the set of concepts corresponding to features $\phi_1, ..., \phi_n$ respectively. Then $\phi_1, ..., \phi_n$ are jointly sufficient for the representation of concept C if and only if $\{\Phi_1\cap, \ldots, \cap\Phi_n\} \subseteq C^I$. An example of a refutal (i.e. an attacking argument) to the above assumption would be the existence of an individual that has these properties but does not belong to C. Conflicting arguments about these notions can be used to differentiate concept definitions between different ontologies.

3.4. Bennett's theory

Bennett [14] proved that set equations can be translated to propositional logic formulae that can be tested for their validity with a Gentzen theorem prover. Although his theory was intended primarily for reasoning with mereological relations it is applicable in our case for reasoning with the type of arguments described above. This is because the mereological relations being represented using this theory closely resemble the set-theoretic semantics attributed to the ontological primitives describing associations between concepts in ontologies. Bennett [14] proves that the mereological equations with set theoretic semantics can be translated to equivalent universal equations which can in turn be converted to propositional logic formulae that can be validated with a simple Gentzen theorem prover. Based on Bennett's *classical entailment correspondence theorem* we were able via a small adaptation to derive a *taxonomic entailment correspondence theorem* which is very similar to the theorem described above but concerns hierarchical relations. This is stated below:

3.4.1. Taxonomic entailment correspondence theorem

$$M_H \models \phi \; if \; and \; only \; if \; M_{C+} \models \tau = \mathcal{U} \tag{5}$$

where $\mathcal{U}$ is the universe of discourse. Unintended models of the theory are excluded by the use of (entailment) constraints. It therefore follows that satisfaction of these constraints forms a refutal against the association of concepts being modeled. To avoid technical details which are beyond the scope of this paper, it suffices to say that since each categorical statement in a syllogistic argument can be translated to propositional form, then the validity of the syllogistic argument can be tested against a propositional theorem prover.

3.5. Conflicts between arguments

Intuitively, a set of arguments consists of a minimal set of premises (here categorical statements) used in the derivation of a claim. In this paper we focus on strict arguments that are inferred via the classical entailment relation. Two arguments conflict with each other (attack) if either (i) the claim of one argument is inconsistent with the claim of the other argument (i.e. *rebutal* [15]) or (ii) the claim of one argument is inconsistent with one of the other premises of the other argument (i.e. *undercutting* [15]) or (iii) one argument's premises are inconsistent with the other argument's premises. Since a syllogism is defined entirely in terms of categorical expressions then two syllogistic arguments conflict each other if any expression in one argument is inconsistent with an expression in the other argument.

4. Human Computer Interaction Using Arguments

In this section we extend the example given in section 1.2 to include an interaction between a human learner and a software agent to clarify differences in ontologies.

4.1. Extended Example

Scenario as above but extended to show the situation where a learner initiates a dialogue with the agent to discuss the differences in conceptualizations (in this paper we make the simplifying assumption that ontologies are used to represent conceptualizations). The software agent uses the ontology of the resource to provide tutoring support to the learners. The dialogue proceeds as follows:

(1) **L:** Let us Discuss the statement in Resource: URI_1: *Visual Basic is an Object Oriented Language*?
(2) **A:** Does the resource provide a justification why *Visual Basic is an Object Oriented Language*?
(3) **L:** The resource URI_1 states that *visual basic is an object oriented language because visual basic contains Objects*.

In line (1) above the learner is confused about the truth of a statement. The focus of the dialogue is 'visual basic is an object oriented language'. The agent responds to the request of the learner to discuss this statement by trying to find out whether the resource provides a justification (i.e. whether the statement is a claim of an argument) or whether the statement is an assertion. To find out whether the learner agrees with the viewpoint of the resource, the software agent A explores the viewpoint of the learner. The interaction continues with:

(4) **A:** So do you agree that *visual basic includes objects*?
(5) **L:** Yes..I agree that *visual basic includes objects*
(6) **A:** Do you agree that a *programming language is object oriented language* because it *contains objects*?
(7) **L:** I disagree with this.
(8) **A:** OK, do you agree that *contains objects is a necessary property of object oriented language*?
(9) **L:** Yes.

During lines $(4) - (9)$ the agent collects information about the learner. The learner may disagree with the argument given by the resource in the following ways: (i) either disagrees that 'visual basic contains objects' or (ii) disagrees that the property 'contains objects' is necessary to infer that visual basic is object oriented language, or (iii) disagrees that 'contains objects' is sufficient to infer that visual basic is object an oriented language. In lines $(4)-(5)$ the agent determines whether the learner believes that visual basic has property 'contains objects'. In lines $(6)-(7)$ the agent determines whether the learner believes that 'contains objects' is a necessary and sufficient property of 'object oriented language' (i.e. whether it defines the concept object oriented language). In lines $(8)-(9)$ the agent determines whether the learner believes that property contains objects is necessary for the concept object oriented language. After this interaction the agent knows that although the learner believes that 'contains objects' is a necessary property of 'object oriented language', it is not sufficient to define the concept of 'object oriented language'.

(10) **A:** What other property is a necessary property of object oriented language?
(11) **L:** Encapsulation is a necessary property of Object oriented language.
(12) **A:** So, you think that *IF a programming language contains objects AND has the property of encapsulation then it is an object oriented language?*
(13) **L:** Yes, that's it!
(14) **A:** Therefore, you think that *IF visual basic has the property of encapsulation THEN visual basic is an object oriented Language*
(15) **L:** Yes.

During the lines $(10)-(15)$ the agent is able to obtain the learner's definition of object oriented language and is able to explain her confusion. In this case, confusion arises due to inconsistency between the knowledge of the resource and the learner. Such confusion can arise due to directly conflicting arguments.

4.2. Dialogue Moves

Interaction between the learner and the tutor is enabled via a set of statement openers and templates which provide the structure of communication and constrain the scope of interaction. The parts that the user is required to complete represent ontological statements which correspond to OWL-DL statements. Utterances are parsed internally as *moves*. A dialogue move is a tuple of the form:

$$DM(id, Speaker, Move, \phi) \tag{6}$$

where id is the identifier of the move, *Speaker* is the utterer, *Move* is a performative verb representing the locution being uttered, and ϕ is an ontology statement.

4.3. Situation Calculus Approach to Formalizing Moves

A situation calculus [16,17,18] approach to modeling moves and changes in the commitment stores of the participants of the dialog, is followed. A similar approach is advocated by Brewka in [19]. Each move in our framework is formalized in terms of its effect on the beliefs of the participants and advances the existing situation to the next situation. Below we illustrate the formalization of a set of effect rules from moves performed by the learner. It is important to note that both the learner and the tutor are allowed to disagree and challenge each other's opinion. A complete list of the effect rules is beyond the scope of this paper.

4.3.1. The learner initiates the discussion

$commit(learner, \{*\phi\}, do(DM(id, learner, iDiscuss, \phi), s_0))$ where $*\phi = \neg Bel\phi \wedge \neg Bel\neg(\phi)$.
i.e. the learner commits to not knowing whether ϕ after it initiates the discussion.

$commit(learner, \{*\phi, R_i : \phi\}, do(DM(id, learner, iDiscuss, R_i : \phi), s_0))$
i.e. the learner commits that $R_i : \phi$ after it initiates the discussion for $R_i : \phi$.

4.3.2. The learner clarifies a statement asked to clarify by the tutor

$commit(learner, R_i : \psi, do(DM(id, learner, iClarify, R_i : \phi \textbf{ because } \psi), do(a, s)) \leftarrow a = do(DM(id, tutor, qClarify, \phi) \wedge commit(learner, R_i : \phi, s))$.
i.e. the learner is committed that $R_i : \psi$ is the justification (in our case sufficient condition) provided for $R_i : \phi$.

4.3.3. The learner justifies a statement challenged or questioned to clarify by the tutor

$commit(learner, \phi \textbf{ because } \psi, do(DM(id, learner, iJustify, \textbf{because } \psi), do(a, s))) \leftarrow commit(learner, \phi, s) \wedge (a = DM(id, tutor, qClarify, \phi) \vee (a = DM(id, tutor, qChallenge, \phi)))$.
i.e. the learner provides a justification (sufficiency condition) for believing ϕ.

4.3.4. Either of the agents disagrees a statement

$commit(S, \neg\phi, do(DM(id, S, iDisagree, \phi), do(a, s))) \leftarrow commit(\dot{S}, \phi, s) \vee a = DM(id, \dot{S}, qInquire_1, \phi)$.
i.e. agent S disagrees that ϕ if the other participant, $\dot{S}$ have already committed to ϕ A

full list of moves with their corresponding natural language expression is provided in the table below.

5. Conclusion and Future Work

In this paper we introduced the notion of cognitive learning context that refers to multiple and possibly inconsistent ontologies. Differences in ontologies can be identified via arguments that can be inferred from relevant subsets of terminological axioms and assertions of ontologies referred to by the cognitive context. We show that syllogistic arguments follow naturally from ontological primitives and we represent arguments about the necessary and jointly sufficient properties of concepts. We also illustrated via the use of an example a dialogue where the learner interacts with the software tutor in order to clarify differences in ontologies via the use of justifications provided in support of claims made either by the learner or the learning resources accessible to the learner. Issues like alignment of vocabularies of different ontologies are addressed via default inference rules. In the near future we plan to elaborate on the formalization of arguments and define precisely the associations between arguments and their relevance to different situations. Additionally we plan to work on dialogue management taking into consideration the cognitive state of the learner in each situation.

Speech Act	Natural Language Expression
With **inform (super)type:**	
The discuss move	
$DM(id, l, iDiscuss^*, ist(URL, \phi))$	Let us discuss statement ϕ in URL.
The clarify move	
$DM(id, l, iClarify, ist(URL, \{\psi, \psi \Rightarrow \phi\}))$	The resource with $URI = URL$ states that ϕ holds because ψ holds
The justify moves	
$DM(id, l, iJustify, \phi)$	Because ϕ.
$DM(id, t, iJustify\phi)$	(ψ holds) Because ϕ.
$DM(id, l, iJustify, \psi)$	Because ψ and $\psi \Rightarrow \phi$.
The agree moves	
$DM(id, l, iAgree, \phi)$	Yes, I agree that ϕ.
$DM(id, t, iAgree, \phi)$	Yes, I agree that ϕ.
The disagree moves	
$DM(id, l, iDisagree, \phi)$	I disagree that ϕ.
$DM(id, t, iDisagree, \phi)$	I disagree that ϕ.
$DM(id, l, iDisagree, \_)$	I disagree with the previous statement.
$DM(id, t, iDisagree, \_)$	I disagree with this statement.
$DM(id, l, iDisagree, \psi)$	I disagree because ψ.
$DM(id, l, iDisagree, \psi \Rightarrow \phi)$	I disagree because ψ implies ϕ.
The claim moves	
$DM(id, l, iClaim, \phi)$	I think that ϕ.
$DM(id, l, iClaim, \psi \Rightarrow \phi)$	I think that if ψ then ϕ.
The concede moves	
$DM(id, S, iConcede, \phi)$	Yes, I think that ϕ.
With **Question (super)type:**	
The clarify move	
$DM(id, l, qClarify, ?\psi : ist(URI, \psi \Rightarrow \phi))$	What is the explanation given in resource with $Resource_{uri} = URI$ for ϕ?
The inquire moves	
$DM(id, t, qInquire_1, \phi)$	Do you think that ϕ?
$DM(id, t, qInquire_1, \psi \Rightarrow \phi)$	Do you think that if ψ then ϕ ?
$< id, t, qInquire, \phi >$	What is ϕ?
The challenge moves	
$DM(id, t, qChallenge, \phi)$	Why do you think that ϕ?
$DM(id, t, qChallenge, \psi \Rightarrow \phi)$	Why do you think that ψ implies ϕ? (Here we assume that ψ is given as a reason, the rule of which is not clear)

Table 1. **l** stands for the **learner**, **t** stands for the tutor, ϕ is a statement in the domain language, URL is the uri of the external resource and $ist(URL, \phi)$ means that ϕ is true in resource with $URI = URL$.

References

[1] Fauconnier, G.: Mental Spaces. A Bradford Book, The MIT Press Cambridge, Massachutts, London, England (1985)

[2] Dinsmore, J.: Mental spaces from a functional perspective. Cognitive Science (1987) 1–21

[3] Giunchiglia, F., paolo Bouquet: Introduction to contextual reasoning, an artificial intelligence perspective. Technical report, D.I.S.A-University of Trento (2000)
[4] Guha, R.: Varieties of context. CONTEXT 2003, pages 164–167 (2003)
[5] Bouquet, P., Giunchiglia, F., van Harmelen, F.: Contextualizing ontologies. Dit-04-013, Department of Information and Communication Technology, University of Trento (2004)
[6] Bouquet, P., serafini, L., Zanobini, S.: Peer-to-peer semantic coordination. Technical report, Istituto Trentino di Cultura (2004)
[7] McCarthy, J.: Notes on formalizing contexts. In Kehler, T., Rosenschein, S., eds.: Proceedings of the Fifth National Conference on Artificial Intelligence, Los Altos, California, Morgan Kaufmann (1986) 555–560
[8] Ghidini, C., Giunchiglia, F.: Local models semantics, or contextual reasoning = locality + compatibility. Artificial Intelligence **127** (2001) 221–259
[9] Fagin, R., Halpern, J.: Awareness and limited reasoning. Artificial Intelligence **34** (1988) 39–76
[10] Martins, A, T., Pequeno, M., Pequeno, T.: A multiple worlds semantics for a paraconsistent nonmonotonic logic. In Carnielli, W., Coniglio, M., eds.: lecture notes in pure and applied mathematics, paraconsistency: the logical way to the inconsistent. Volume 228. (2002) 187–209
[11] Lokhorst, G.J.C.: Counting the minds of split-brain patients. Logique and Analyse (1996) 315–324
[12] Dalmau, M.: Ambiguity as a conceptual framework for design. (Technical report)
[13] Walton, D.: Foundamentals of Critical Argumentation. Cambridge University Press (2006)
[14] Bennett, B.: Logical Representations for Automated Reasoning about Spatial Relationships. PhD thesis, School of Computer Studies, The University of Leeds (1997)
[15] Dung, P.: On the acceptability of arguments and its foundamental role in nonmonotonicreasoning,logic programming and n-person games. Artificial Intelligence **77** (1995) 321–357
[16] Levesque, H., Pirri, F., Reiter, R.: Foundations for the situation calculus. European Transcations on AI **(reference paper)** (1998)
[17] Reiter, R.: The frame problem in the situation calculus: A simple solution (sometimes)anda completeness result for goal regression. In: Artificial Intelligence and Mathematical Theory of Computation: Papers inHonorof John McCarhty. Academic Press (1991)
[18] Reiter, R.: Knowledge in action: Logical foundations for describing and implementingdynamicalsystems. book draft (1999)
[19] Brewka, G.: Dynamic argument systems: A formal model of argumentation processes based on situation calculus. Journal of Logic and Computation **11**(2) (2001) 257–282

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-79

KEMM: A Knowledge Engineering Methodology in the Medical Domain

Pinar WENNERBERG [a] Sonja ZILLNER [a] Manuel MÖLLER [b]
Paul BUITELAAR [c] Michael SINTEK [b]

[a] *Siemens AG, Corporate Technology, Munich, Germany*
[b] *DFKI GmbH, Knowledge Management Dept., Kaiserslautern, Germany*
[c] *DFKI GmbH, Language Technology Lab, Saarbrücken, Germany*

Abstract. Medical research and clinical practice deal with complex and heterogeneous data. This requires a systematic approach for semantic integration of information to support clinicians in their daily tasks.

As the clinicians speak and think in a very different language than that of the computer scientists, existing knowledge engineering approaches based on classical expert interviews fall short. Moreover, as human health is a very sensitive subject, the reuse of standardized hence reliable ontologies as medical knowledge resources becomes a key requirement.

In this paper, we first discuss the specific medical knowledge engineering requirements, we identified along a semantic medical image and text retrieval use case. Then we report on ongoing work towards establishing a corresponding methodology based on ontology reuse that is derived from the requirements. The methodology, which will be discussed in detail, relies on a novel technique for semi-automatically generating a set of potential user queries to support the knowledge elicitation process.

Keywords. knowledge engineering methodology, biomedical ontologies, reasoning, image retrieval

Introduction

Clinical care and research deal with large volumes of complex information that originates from different sources, with different structures and different semantics. By establishing an explicit formal specification of the concepts and their interrelations for a particular domain, such as medicine, ontologies (following the definition of the term "ontology" by [1]) facilitate integration and reuse of valuable knowledge across applications.

To incorporate the external medical knowledge in the ontologies and hence to semantically enhance clinical data, one has to identify the query strategies that the clinicians are interested in. As medical knowledge concerns a very sensitive context, *i. e.*, the human health, reusing standardized thus reliable medical ontologies instead of developing them from scratch is an important requirement. Consequently, it becomes necessary to decide for the appropriate, application related ontologies and their fragments (or modules).

The focus of this paper is to discuss the challenges, requirements and best practice solutions for ontology-based knowledge engineering in the medical domain. In particular, work towards establishing the KEMM (Knowledge Engineering Methodology in the Medical Domain) methodology consisting of several knowledge engineering sub-processes is introduced, whereby the focal point here is the linguistic-driven ontology engineering. The KEMM methodology is derived from the experiences gained during the realization of a clinical use case within the context of the THESEUS MEDICO project, which [1] aims for advanced search and analysis technologies that exploit image semantics. For a more detailed description the reader is referred to [2].

The main research contributions of this paper are twofold. Firstly, the domain specific challenges and requirements for medical knowledge engineering are identified and discussed. Secondly, an initial knowledge engineering methodology is introduced that is being designed to address these requirements.

The envisioned methodology has two main objectives. Firstly, it will support the communication between the knowledge engineer and the clinical expert so that the expert's knowledge can be acquired most efficiently. Secondly, the knowledge acquired in this way will be transferred to the target software application effectively.

The KEMM methodology defines six tasks to achieve these objectives: *Query Pattern Derivation*, *Ontology Identification*, *Ontology Modularization and Pruning*, *Ontology Customization*, *Ontology Alignment*, *Reasoning-Based Ontology Enhancement* and *Ontology Testing and Deployment*. We will discuss all six steps in detail, and will demonstrate how our methodology is applied in a MEDICO specific use case, where the goal is to find all related information (images, text) about patients suffering from head and neck lymphoma.

The remainder of this paper is organized as follows. Section 1 discusses the challenges of the medical knowledge engineering process, whereas Sect. 2 presents the requirements derived to address these challenges. Section 3 introduces the actual KEMM methodology, describing all the tasks that comprise the methodology in detail. In Section 4 we compare our work with related approaches and Section 5 concludes the paper with open issues and further research.

1. Challenges of Knowledge Engineering in Medical Applications

In this paper we refer to knowledge engineering as a collection of systematic activities conducted to build knowledge-based systems as defined in [3]. Additionally, we require that the knowledge-based application incorporates ontologies, which provide the domain model, *i. e.*, they supply the background knowledge necessary to build the application.

One challenge in knowledge engineering is the so-called *"knowledge acquisition bottleneck"* meaning that the knowledge is hard to acquire as it is in the heads of the domain experts (*i. e.*, tacit knowledge) or it is distributed in different

[1]This research has been supported in part by the THESEUS Program in the MEDICO project, which is funded by the German Federal Ministry of Economics and Technology under the grant number 01MQ07016. The responsibility for this publication lies with the authors.

resources in different formats (*e. g.*, natural language, visual media, structured, semi-structured sources). Another challenge is the discovery of actual knowledge from the overdose of data.

We distinguish between two different types of knowledge engineering challenges; the first one concerns the communication process between the knowledge engineer and the medical expert, whereas the second is specifically about the engineering of medical ontologies.

The communication related challenges were identified during the intensive interviews with medical experts in the context of the MEDICO project. These interviews revealed the fact that each party has quite a different perception of the medical domain. From the computer scientist's perspective, the knowledge, in the way it is presented by the medical expert (and who are also always in lack of time) is often fuzzy if not contradictory or inconsistent.

The medical ontology engineering specific challenges arise due to multiple reasons. One reason is that knowledge in the medical ontologies is hidden, *i. e.*, it is opaque to the ontology engineer as he is not familiar with the terminology. Another one concerns the sheer size of the medical ontologies such as the Unified Medical Language System (UMLS) [4] containing over one million concepts or the Foundational Model of Anatomy (FMA) [5] with 75,000 concepts and over 2.1 million relationship instances about human anatomy. Thus, accessing the contents of complex ontologies of these sizes and hence assessing their appropriateness for the target application becomes an issue. Finally, modeling medical information is another major challenge as it requires accurate and comprehensive domain knowledge, which can only be acquired through intensive domain expert support[2].

To address these challenges, some knowledge engineering methodologies [6,7] suggest a process, where the so-called *"competency questions"* shall support the knowledge transfer from the expert to the engineer and shall help structure the engineering process. These questions, for which the future application shall be capable of providing answers, are derived from the business use cases. In the medical domain, however, the knowledge is too specific, too technical and sensitive for a knowledge engineer to be able to define such competency questions even when assisted by the clinician.

Thus, other aiding tools and methodologies become necessary for at least two purposes; (1) for supporting the communication between the expert and the engineer to acquire the expert knowledge most efficiently and (2) for engineering the knowledge acquired so that it can be incorporated into the target application most effectively.

2. Requirements Analysis

To achieve the objectives set for MEDICO in a systematic way we identified the following requirements:

[2] *i. e.*, only for the FMA Ontology more than 30 person years were needed so far according to the FMA FAQ, `http://sig.biostr.washington.edu/projects/fm/FAQs.html`.

Query Pattern Derivation For improved semantic medical applications, we need to identify the right level of information coverage and detail the clinicians are interested in. This type of information is typically contained in the queries the clinicians would want to ask to a clinical search engine. Yet, it is difficult to acquire it by classical interview techniques as the clinicians rely on very specific information that is difficult to talk about in a general manner. Thus, we require to establish means for semi-automatic query pattern derivation, *i. e.*, an approach for generating a set of hypothetical user queries that are subsequently evaluated by the clinicians.

Ontology Identification As human health is a sensitive matter, the quality and the quantity of the medical knowledge to be used in the target application has to be ensured by reusing the work of acknowledged authors and standardization committees that comes in form of medical ontologies. However, these ontologies are typically very comprehensive and cross-linked, which cannot be easily read, navigated or understood. Therefore, we require some automatic support for identifying the appropriate ontologies that fulfill our requirements set by the query patterns that were derived and expert validated in the previous step.

Ontology Modularization and Pruning For an effective reuse of the large medical ontologies we require (modular) ontology subsets that can be easily navigated by humans and reasoned by machines. These modules need to cover all concepts and relationships for describing the particular scenario, thus the derived set of query patterns will also determine the criteria for pruning and modularizing the ontologies that were identified as relevant.

Ontology Customization Quite often the modules extracted from the ontologies have either redundant or missing knowledge, which therefore will be post-processed, *i. e.*, customized to meet the requirements w.r.t. the application.

Ontology Alignment Each customized ontology module represents a piece of knowledge that is necessary to realize the entire application. These knowledge pieces are not arbitrary but they need to be interrelated within the context of the application. Therefore, the separate ontology modules will be integrated to deliver the whole picture.

Reasoning-Based Ontology Enhancement To discover new knowledge in form of relations and concepts, reasoning processes will need to be incorporated.

Testing and Deployment The results of each step will be tested for validity in an iterative process before the engineered model is deployed.

3. The KEMM Methodology

The methodology being developed is designed to address the two main challenges, *i. e.*, communication and medical ontology engineering. Consequently, it defines an initial task called *"query pattern derivation"* that supports the communication process between the clinical experts and the computer scientists during the interviews. This task also frees the expert from having to imagine arbitrary scenarios that can be potential use cases. As described in the next section, it is based on semi-automatically deriving a set of query patterns that represent potential medical expert queries.

The succeeding tasks of the envisioned methodology are also driven by our query pattern derivation approach. Again, based on these patterns the ontologies to be reused are identified, pruned, and modularized; the relevant modules are customized and finally integrated. Once an integrated model is established in this way, reasoning processes can be applied to infer new knowledge and thus enhance the model. Parallel to this, the system is iteratively tested to guarantee consistency, clarity, coherence and validity before it is deployed. Figure 1 gives an overview of the methodology being developed that comprises the six tasks mentioned above.

3.1. Query Pattern Derivation

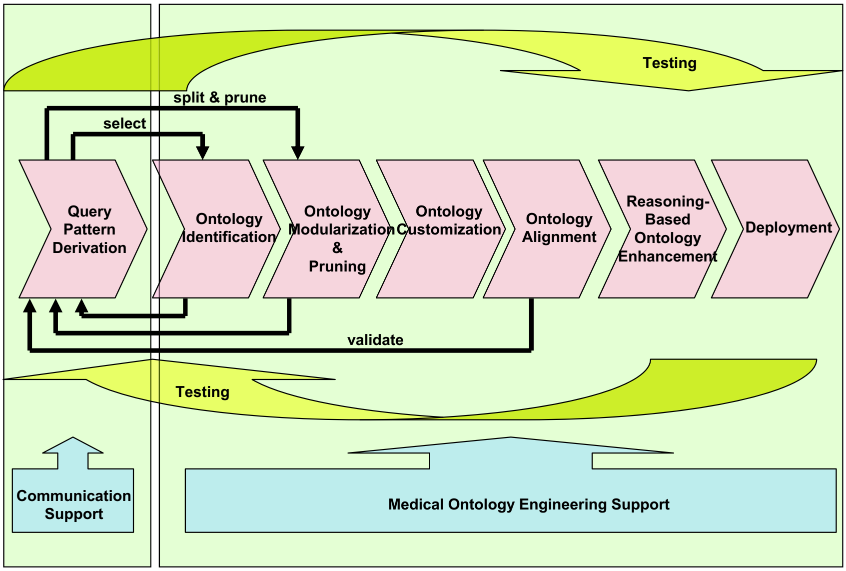


Figure 1. KEMM Workflow

To improve semantic medical text and image search, it is essential to first find out what kind of information the clinician, in our case a lymphoma expert, wants to know. Interview-based approaches such as [8] and [9] to the identification of possible user queries did not provide satisfactory results within the medical domain because of the reasons discussed earlier.

Thus, we concluded that pre-interview preparations were necessary to achieve effective results from the actual interviews on the way to discover the expectations of the clinicians from a semantic medical search engine.

We identified a set of interrelated domain terms, concepts and relations, that we considered as potentially relevant to the clinicians, which therefore can be hypothesized as query patterns, *i. e.*, typical queries that the clinicians would want to ask. To identify the set of potential query patterns, we referred to the information readily available in ontologies, in patient health records, on the Web *etc.*The initial set of query patterns established in this way, provided the basis for a (more successful) communication with the clinicians at the same level of granularity. Additionally, it released the medical experts from the obligation of having to imagine arbitrary query use cases, thereby guiding them to concentrate on more machine-processable and technologically realizable scenarios.

Query pattern derivation is based on a combination of various techniques from natural language processing and text mining as described in detail in [10], [11]. A possible query pattern for the head and neck lymphoma is shown below, where the anatomical structure is instantiated by the head and neck, the radiology image and its modality are instantiated by the head and neck CT scan and the disease, symptom or observation is instantiated by the lymphoma (or lymph node).

[ANATOMICAL STRUCTURE] *located_ in* [ANATOMICAL STRUCTURE]
AND
[[RADIOLOGY IMAGE] modality] *is_ about* [ANATOMICAL STRUCTURE]
AND
[[RADIOLOGY IMAGE] modality] *shows_ symptom* [DISEASE,SYMPTOM,OBSERVATION]

3.2. Ontology Identification

Because of the reasons discussed earlier, in MEDICO we follow the rationale of ontology (module) reuse rather than developing them from scratch. Consequently, the available medical ontologies need to be identified to incorporate the knowledge from those of high quality which have been developed over the years as a result of joint efforts of knowledge engineers and health care experts. Reusing existing medical ontologies also requires evaluating their appropriateness w.r.t. the target application, before finally the selecting them (or not).

The identification of the domain and task relevant, high quality ontologies is not a trivial task. On the contrary, it is an active research field on its own with various approaches [12,13,14] offering different solutions such as ontology search engines (*e. g.*, Swoogle[3], Watson[4]), ontology libraries (*e. g.*, DAML ontology library[5]) or both (*e. g.*, OntoSelect[6]). However, most influential (w.r.t to Google rankings) work in medical ontology development has been carried out by the joint efforts of computer scientists and biologists working within the Open Biomedical Ontologies (OBO) consortium [7], whose activities are reported in [15]. Within this framework the main focus has been on strategies for identifying and creating high quality ontologies within the biomedical domain by adhering to a set of ontology design and development principles [8].

Within the MEDICO context, the interviews with the clinicians and radiologists showed that medical imaging and patient data need to be considered along

[3]`http://swoogle.umbc.edu/`
[4]`http://watson.kmi.open.ac.uk/editor_plugins.html`
[5]`http://www.daml.org/ontologies/`
[6]`http://olp.dfki.de/ontoselect/`
[7]`http://www.obofoundry.org/`
[8]`http://www.obofoundry.org/crit.shtml`

three different perspectives; (i) the anatomical spatial perspective that addresses body parts and their locations, (ii) the radiology-specific perspective, which describes the relationships between various image modalities and anatomical regions as shown on medical images and finally (iii), the disease perspective that concerns the distinction between the normal and the abnormal imaging features.

We set these three perspectives as our search dimensions in identifying relevant ontologies and thesauri. Consequently, we manually searched in BioPortals,[9] or used ontology search engines to determine a first set of potentially relevant ontologies. Upon agreement with the clinicians, from this set we decided for the FMA in anatomy, RadLex in radiology and for ICD-9 CM to represent the disease dimension.

The main criterion we set for the ontology selection process in the MEDICO scenario was the capability of the ontologies to reflect the three joint perspectives (or dimensions) that are the in focus of the radiology images and hence in that of the semantic search. Consequently, we excluded ontologies as (*e. g.*, UMLS, MeSH, GALEN *etc.*) since these were far too generic for our purposes. Further criteria we considered were:

Representation Language The Web Ontology Language OWL[10]was a pre-defined requirement for the THESEUS-MEDICO use case, therefore we considered only those ontologies that were available in OWL format or were easily transferable.

Comprehensiveness Even though the amount of available knowledge in the medical ontologies is a challenge, we concentrated on those ontologies that are most comprehensive (*e. g.*, FMA) to avoid missing relevant important information.

Popularity Being non-experts, the popularity of a given ontology was a helpful guide for determining its appropriateness for our use case. Popularity, in our case, is defined in terms of how well-known it is, the amount of documentation available and the number of projects using the ontology.

Semantic formalism The logical formalism was also a pre-defined requirement of the THESEUS-MEDICO use case, therefore we considered only those ontologies that were formalized in Description Logics.

While we have already described the overall MEDICO ontology hierarchy (see Fig. 2) in another publication [16], Fig. 3 details the relationships between medical thesauri and ontologies in MEDICO. On the bottom of the diagram we list the different thesauri and on the top the corresponding ontologies. Thesauri are dictionaries of words but they additionally contain synonyms and antonyms. Ontologies contain abstract concepts, which are language-independent and they represent formal specifications of the represented entities. *Terms* in the thesauri are connected to concepts in the ontologies via simple `rdfs:label` or preferably via LingInfo relations (that allow more complex linguistic information to be attached to concepts, see [17]), which thus conceptualize these entities by using ontology mapping techniques as reported in [18].

[9] `http://www.bioontology.org`
[10] `www.w3.org/TR/owl-features/`

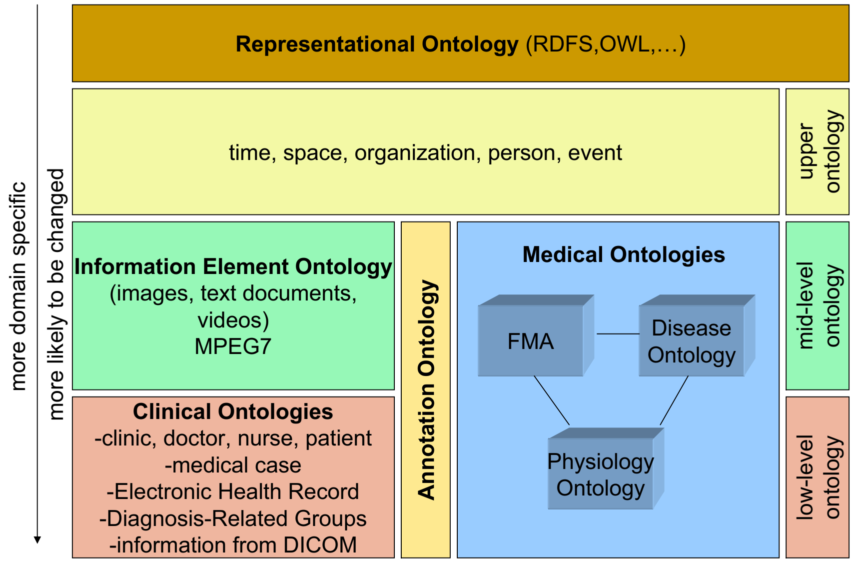


Figure 2. MEDICO Ontology Hierarchy

From left to right we differentiate between different aspects—throughout this document we will also refer to them as *dimensions*—of the conceptualization of the medical domain.

The Information Element Ontology contains the information elements that we want to annotate (images, text documents, videos, ...). [17] have shown how MPEG7[11] can be used as a generalized formalism for segmentation of arbitrary document formats and annotation of segments. We will apply this generalized segmentation to treat parts of images and documents in the same way as complete documents.

The *Image Parameters* ontology contains abstract descriptions of imaging procedures (*e. g.*, the peculiarities of CT scans, of ultrasound examinations), about the image quality *etc.*

The *Annotation Ontology* includes concepts, which are used to annotate information elements with concepts from the Medical Ontologies that were detected during object recognition. Using an ontology (instead of just a simple relation) allows us to express that, *e. g.*, an image *partially* deals with a specific concept from the anatomical ontology, because only parts of it are on the picture. At least in some cases we want to annotate the relations with a probability. The Annotation Ontology allows us to express such qualifications as properties of attributes.

On the right side of the *Annotation Ontology* are the different medical ontologies and thesauri, which cover different dimensions of image annotation. Here we differentiate between *Anatomy*, *Disease* and other aspects like medical treatment and applied substances. RadLex is a thesaurus, which covers different aspects: anatomical, pathological as well as aspects about imaging parameters, quality and even applied treatments. Thus, the according parts of RadLex appear in each

[11]MPEG Homepage: `http://www.chiariglione.org/mpeg/`

of these dimensions. For the disease dimension we additionally leverage on ICD9 CM. Together with the lymphoma part of NCI Thesaurus they are conceptualized to the *Disease Ontology*.

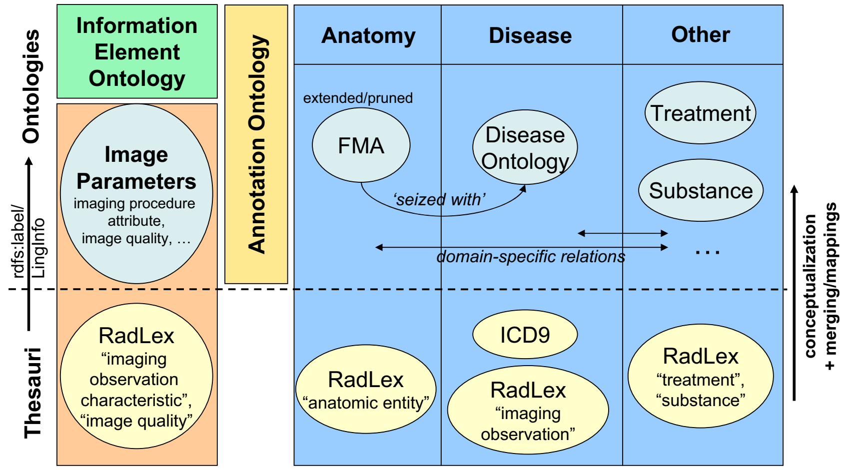


Figure 3. Relationships Between Medical Thesauri and Ontologies in MEDICO

3.3. Ontology Modularization and Pruning

Ontology modularization can be addressed automatically or user-driven, but in both cases the segmentation of the ontology is a difficult task. For instance, ontology modularization approaches guaranteeing logical consistency, such as [19], occasionally result in too large fragments and a slow algorithm. On the other hand, graph-based approaches such as [20] perform well in terms of speed and of fragment size but do not guarantee the logical completeness. User-driven, semi-automatically created fragments established in the way as described in [21], have the advantage of covering the necessary level of detail, but they require additional user interaction and might miss relevant relational knowledge.

Within the KEMM methodology, ontology pruning can be realized based on the query patterns, as these patterns reflect the user relevant level of detail and coverage that the ontology is expected to contain. This implies that those parts of the selected ontologies that are not relevant to the initial set of query patterns need not be considered. Thus, the query patterns derived as the initial task of our methodology act as a filter in a way for determining not only the relevant ontologies but also their relevant parts which allows them to be pruned to (hopefully) useful sizes. In case query patterns are clustered (*e. g.*, based on query situations), pruning the ontologies based on these clusters will derive ontology modules (one module per cluster). These pruned (and modularized) ontologies are then represented to the clinical experts to confirm relevance and validity.

We followed this approach when determining the lymphoma use case relevant parts of the NCI thesaurus, the FMA ontology and the RadLex annotation schema. Consequently, we semi-automatically extracted the lymphoma related parts of these resources based on simple string-similarity. Assuming that anything containing the base word form *lymph* would in some way be related to lymphoma, we recorded all terms/concepts/entities that matched lymph*. Thus, we obtained a return set with *lymphoma, lymph node, Hodgkin's lymphoma etc.*, which was then statistically profiled yielding a ranked term list as described in [10]and was discussed with the clinical expert.

3.4. Ontology Customization

Often the ontologies that were modularized/pruned in this way did not qualify for an immediate deployment because of several reasons. Sometimes they did not cover the information required so that additional attributes, relations or concepts were needed. In contrast, there were also cases with redundancies or some knowledge was irrelevant. Finally, in some cases the initial ontology syntax (*i. e.*, the language) did not fit.

During our work, we came across to all three cases. As we were interested in modeling the three dimensions, *i. e.*, the joint perspectives of anatomy, disease and radiology it was important for us to establish the correspondences between the three semantic resources. As cross-references did not exist between them, we semi-automatically created relationships, where the concepts (or terms for that matter) from these resources can be referenced to each other within the lymphoma context. For example, we defined a relationship called *has_ nci_ code* that relates the concepts in the lymphoma module to the entities in the NCI thesaurus. Similarly, the associated RadLex terms and the FMA concepts are related to each other along *the relates_ to* relationship. Finally, as the the lymphoma extract from the NCI thesaurus was in flat text format, it was semi-automatically converted to the OWL syntax.

3.5. Ontology Alignment

We use the term ontology alignment to imply ontology mapping and integration. Once again, since our goal was to obtain the joint view of anatomy, disease and radiology as observed on the images, separate knowledge pieces needed to be integrated to deliver the whole picture. This ontology alignment process, however, happens in our case at the ontology module level. In other words, the KEMM methodology targets integrating the customized ontology fragments or modules rather than the actual ontologies themselves because of the reasons discussed earlier.

Ontology alignment is studied under various subtopics such as *ontology merging, ontology mapping*, and *ontology integration* and is tackled by different linguistic, logic or graph based approaches [22].

When applying the KEMM methodology within the MEDICO context we concentrated on the linguistic approach and consequently performed ontology mapping based on string-similarity. Hence, cross-references were made between the similar concepts/terms/entities of the corresponding ontology modules by using the relationships defined in the previous ontology customization task.

3.6. Reasoning-Based Ontology Enhancement

The MEDICO use case is characterized by reuse and integration of distributed ontological knowledge that may introduce inconsistencies. Inconsistencies can be avoided or detected early more easily by concentrating the reasoning process on the smaller ontology modules instead of the large ontologies themselves.

With the KEMM methodology we concentrate on two specific reasoning services. Thus, in our lymphoma use case one objective is to be able to deduce the relevant image modalities (MR, CT scan *etc.*) given the symptoms of head and neck lymphoma.

Via deductive reasoning we also target the discovery of valid relationships—spatial as well as pathological and physiological— between anatomical structures.

3.7. Testing and Deployment

To avoid the propagation of inconsistencies and modeling mistakes, each and every task shall be tested for validity, completeness and coherence. The query patterns additionally need to be verified iteratively by the clinical experts.

During deployment one MEDICO specific reasoning scenario is the identification of possible diseases or symptoms given an anatomical structure and an image modality. Furthermore we can facilitate inductive reasoning to discover relations based on existing facts *e. g.*, between disease symptoms and body regions.

4. Related Work

There has been a rapid increase in ontology building, especially within the context of the Semantic Web activities initiated by various groups from industry and academia. It has been suggested that these separate activities should be executed in a systematic manner [23,24], as a result of which several ontology engineering methodologies have been proposed [7,25,6].

Some of these methodologies are the outcome of experiences collected during the ontology development process such as the Enterprise Methodology [7] or the TOVE (Toronto Virtual Enterprise) Methodology [25]. The TOVE methodology is characterized by its definition of the so-called "competency questions" that determine the scope of the ontology to be modeled and for which the future ontology shall be capable of providing answers.

Other communities concentrated on developing stand-alone ontology engineering methodologies that shall be applicable across domains and tasks. Examples of these are METHONTOLOGY [6], ON-TO-KNOWLEDGE [9] and COMMON-KADS [8]. METHONTOLOGY proposes to align ontology development with software development activities and consequently defines an ontology development life cycle process (similar to software development) that consists of several phases.

ON-TO-KNOWLEDGE, on the other hand, is a process oriented methodology that concentrates on ontology based knowledge management and maintenance in distributed enterprises.

The common view of the stand-alone methodologies is that they consider ontology development processes as equivalent to business processes and consequently they introduce higher level activities such as requirements analysis, development, evaluation, maintenance and project management. The methodology described in this paper also includes some of these activities; however, it has other components such as the query pattern derivation and ontology modularization in order to meet the specific requirements of the medical domain. More concretely, the special focus of our methodology is not on the business processes, but more on the specific medical processes such as the clinicians' analysis and diagnosis perspectives.

The TOVE alike methodologies also share a common point with our methodology in that KEMM is also based on the experiences collected along the MEDICO project. However, it differs in the sense that it facilitates ontology reuse by emphasizing the ontology identification, modularization and customization processes.

5. Conclusions and Future Work

In this paper we reported on ongoing work about KEMM, a knowledge engineering methodology for the medical domain, which is derived from the experiences collected along the MEDICO project. KEMM addresses the two domain specific challenges, *i. e.*, communication and medical ontology engineering by defining six separate tasks, whereby the initial task of *Query Pattern Derivation* influences the other tasks as shown in the KEMM workflow.

Next, we will concentrate on deploying the reasoning mechanism over the joint disease, radiology and anatomy perspectives to discover additional knowledge, which might not have caught our attention or is simply unknown to us. In the long run, we target defining new use cases to transfer the KEMM methodology to domains other than medicine, such as engineering or law.

References

[1] T. R. Gruber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.

[2] Manuel Möller, Michael Sintek, Paul Buitelaar, Saikat Mukherjee, Xiang Sean Zhou, and Jörg Freund. Medical image understanding through the integration of cross-modal object recognition with formal domain knowledge. *Proceedings of Healthinf 2008*, 2008.

[3] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach.* Pearson Education, 2003.

[4] Oliver Bodenreider. The Unified Medical Language System (UMLS): Integrating biomedical terminology. *Nucleic Acids Research*, 32 (Database Issue):D267–D270, 2004.

[5] Cornelius Rosse and José L. V. Mejino Jr. A reference ontology for biomedical informatics: the foundational model of anatomy. *Journal of Biomedical Informatics*, 36(6):478–500, 2003.

[6] Mariano Fernández-López, Asunción Gómez-Pérez, J. Pazos, and Alejandro Pazos. Building a chemical ontology using methontology and the ontology design environment. *IEEE Intelligent Systems and Their Applications*, 14(1):37–46, 1999.

[7] Mike Uschold. Building ontologies: Towards a unified methodology. In *16th Annual Conf. of the British Computer Society Specialist Group on Expert Systems*, Cambridge, UK, 1996.

[8] Guus Schreiber, Hans Akkermans, Anjo Anjewierden, Robert Dehoog, Nigel Shadbolt, Walter Vandevelde, and Bob Wielinga. *Knowledge Engineering and Management: The CommonKADS Methodology.* The MIT Press, 1999.

[9] Y. Sure and R. Studer. To-knowledge methodology - final version, 2002.

[10] P. Buitelaar, P. O. Wennerberg, and S. Zillner. Statistical term profiling for query pattern mining. In *Proceedings of the ACL BioNLP 2008*, June 2008.

[11] P. O. Wennerberg, P. Buitelaar, and S. Zillner. Towards a human anatomy data set for query pattern mining based on wikipedia and domain semantic resources. In *Proc. of a workshop on Building and Evaluating Resources for Biomedical Text Mining at LREC 2008*, May 2008.

[12] P. Buitelaar and T. Eigner. Evaluating ontology search, November 2007.

[13] Y. Ding and D. Fensel. Ontology library systems: The key to successful ontology reuse, 2001.

[14] Marta Sabou, Vanessa Lopez, and Enrico Motta. Ontology selection for the real semantic web: How to cover the queen's birthday dinner? In Steffen Staab and Vojtech Svátek, editors, *EKAW*, volume 4248 of *Lecture Notes in Computer Science*, pages 96–111. Springer, 2006.

[15] Barry Smith, Michael Ashburner, Cornelius Rosse, Jonathan Bard, William Bug, Werner Ceusters, Louis J. Goldberg, Karen Eilbeck, Amelia Ireland, Christopher J. Mungall, Neocles Leontis, Philippe Rocca-Serra, Alan Ruttenberg, Susanna-Assunta Sansone, Richard H. Scheuermann, Nigam Shah, Patricia L. Whetzel, and Suzanna Lewis. The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotech*, 25(11):1251–1255, November 2007.

[16] Manuel Möller and Michael Sintek. A generic framework for semantic medical image retrieval. *Proceedings of the Knowledge Acquisition from Multimedia Content (KAMC) Workshop, held at 2nd international conference on Semantics And digital Media Technologies (SAMT)*, November 2007.

[17] Massimo Romanelli, Paul Buitelaar, and Michael Sintek. Modeling linguistic facets of multimedia content for semantic annotation. In Bianca Falcidieno, Michela Spagnuolo, Yannis S. Avrithis, Ioannis Kompatsiaris, and Paul Buitelaar, editors, *SAMT*, volume 4816 of *Lecture Notes in Computer Science*, pages 240–251. Springer, 2007.

[18] Yannis Kalfoglou and Marco Schorlemmer. Ontology mapping: The state of the art. In Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab, and M. Uschold, editors, *Semantic Interoperability and Integration*, number 04391 in Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany, 2005. <http://drops.dagstuhl.de/opus/volltexte/2005/40> [date of citation: 2005-01-01].

[19] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov, and Ulrike Sattler. Just the right amount: extracting modules from ontologies. In *WWW'07: Proceedings of the 16th international conference on World Wide Web*, pages 717–726, New York, NY, USA, 2007. ACM.

[20] E. Jimenez-Ruiz, Rafael Berlanga, V. Nebot, and Ismael Sanz. Ontopath: A language for retrieving ontology fragments. In Robert Meersman and Zahir Tari, editors, *OTM Conferences (1)*, volume 4803 of *Lecture Notes in Computer Science*, pages 897–914. Springer, 2007.

[21] *Semantic Visualization of Patient Information*, June 2008.

[22] J. Davies. *Semantic Web Technologies: Trends and Research in Ontology-based Systems.* John Wiley & Sons, July 2006.

[23] J. BOUAUD, B. BACHIMONT, J. CHARLET, and P. ZWEIGENBAUM. Methodological principles for structuring an ontology, 1995.

[24] D. Jones, T. Bench-Capon, and P. Visser. Methodologies for ontology development.

[25] M. Gruninger and M. Fox. The design and evaluation of ontologies for enterprise engineering, 1994.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-92

Similarity as a Quality Indicator in Ontology Engineering

Krzysztof JANOWICZ [1], Patrick MAUÉ, Marc WILKES, Sven SCHADE, Franca SCHERER, Matthias BRAUN, Sören DUPKE, and Werner KUHN
Institute for Geoinformatics, University of Muenster, Germany
janowicz|patrick.maue|marc.wilkes|sven.schade|franca.scherer|m.braun|soeren.dupke|kuhn@uni-muenster.de

Abstract. In the last years, several methodologies for ontology engineering have been proposed. Most of these methodologies guide the engineer from a first paper draft to an implemented – mostly description logics-based – ontology. A quality assessment of how accurately the resulting ontology fits the initial conceptualization and intended application has not been proposed so far. In this paper, we investigate the role of semantic similarity as a quality indicator. Based on similarity rankings, our approach allows for a qualitative estimation whether the domain experts' initial conceptualization is reflected by the developed ontology and whether it fits the users' application area. Our approach does not propose yet another ontology engineering methodology but can be integrated into existing ones. A plug-in to the Protégé ontology editor implementing our approach is introduced and applied to a scenario from hydrology. The benefits and restrictions of similarity as a quality indicator are pointed out.

Keywords. ontology engineering, semantic similarity, quality assurance, requirements engineering, knowledge management

1. Introduction

Knowledge engineering deals with the acquisition, representation, and maintenance of knowledge-based systems. These systems offer retrieval and reasoning capabilities to support users in finding, interpreting, and reusing knowledge. The engineering of ontologies is a characteristic application of knowledge engineering, with ontologies as tools to represent the acquired knowledge. Various formal languages can be used to implement ontologies, i.e., to develop a computational representation for knowledge acquired from domain experts. Description Logics (DL), mostly used to implement ontologies on the Web, are a family of such languages with a special focus on reasoning services.

Answering the question how adequate the developed ontology captures the experts' initial conceptualizations (i.e., the intended meaning at a specific point in time) as well as the users' intended application area is a major issue in ontology engineering. Several methodologies offer support for knowledge acquisition and implementation, while tools for quality assessment suitable for both the domain experts and ontology users without

[1] Corresponding Author

a strong background in information science are missing. This paper proposes semantic similarity measurement as a potential quality indicator. Similarity measurement – originated in psychology – gained attention as a cognitive approach to information retrieval [1]. Inter-concept similarity rankings obtained using the SIM-DL similarity server [2] have been compared with human similarity rankings. Both correlate positively and significantly, if the natural language descriptions underlying the DL concepts were shown to the participants beforehand [3]. We therefore claim that a correlation between similarity rankings obtained from experts and the computed ontology ranking indicates whether the ontology captures the experts' initial conceptualization (given that the developed ontology was implements using the experts' input).

The paper is structured as follows. It starts with an introduction into relevant aspects of knowledge engineering (section 2) and semantic similarity measurement (section 3). Next, section 4 discusses the role of similarity as a quality indicator within the ontology engineering process. The proposed approach is applied to a hydrology use case involving existing ontologies (section 5). The benefits and restrictions of our methodology are elucidated. Finally, in section 6, conclusions and directions of further work are given.

2. Quality Assurance in Ontology Engineering

Ontologies are typically used for data annotation and integration, or to ensure interoperability between software components. In Ontology Driven Architectures [4], ontologies are included at different stages of the software engineering process. A systematic approach for the development of such ontologies is required to ensure quality. Various methodologies have been developed to accomplish a controlled and traceable engineering process. Overviews of these methodologies are given in [5]. One of the most frequently applied methodologies is *Methontology* [5].

According to *Methontology*, the ontology development process can be divided into five phases: *Specification* includes the identification of intended use, scope, and the required expressiveness of the underlying representation language. In the next phase (*conceptualization*), the knowledge of the domain of interest is structured. During *formalization*, the conceptual model, i.e., the result from the conceptualization phase, is transformed into a formal model. The ontology is implemented in the next phase (*implementation*). Finally, *maintenance* involves regular updates to correct or enhance the ontologies. This paper focuses on two activities involved in this process: *knowledge acquisition* and *evaluation*. Both are discussed in detail in the following subsections.

As illustrated in figure 1, three types of actors are involved in the development of ontologies. The steps 1-4 and the involved actions are described in section 4.

1. *Ontology users* define the application-specific needs for the ontology and evaluate whether the engineers' implementation matches their requirements.
2. *Domain experts* contribute to and agree on the knowledge which should be implemented in the ontology.
3. *Ontology engineers* analyze whether existing ontologies satisfy the experts' needs or implement the experts' conceptualization as a new ontology.

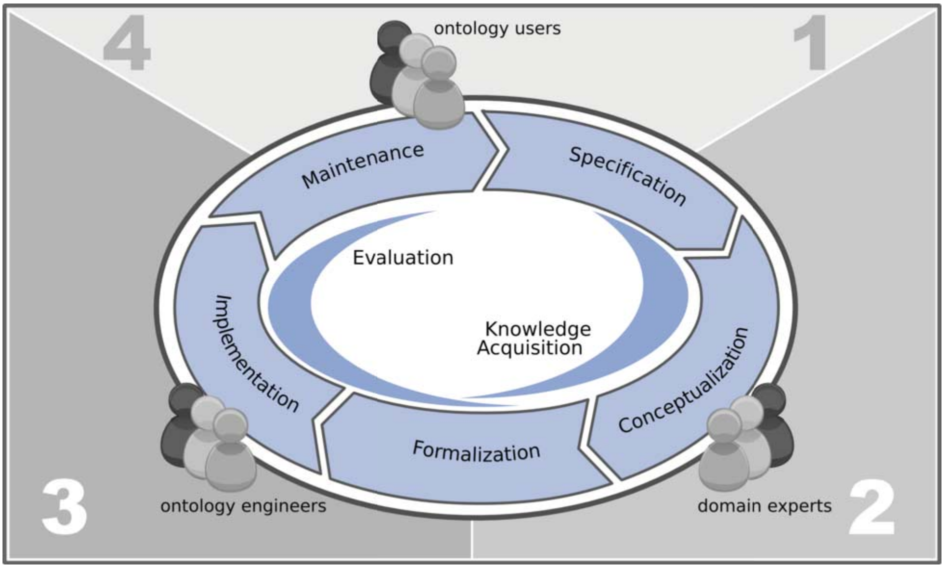


Figure 1. Phases and activities of Methontology and their relation to the actors (modified from [5]).

2.1. Knowledge Acquisition

Knowledge acquisition already starts in the specification phase, and is essential during the conceptualization phase [6]. Similar to software engineering, ontology engineering involves the identification of requirements, e.g., by specifying usage scenarios with the client. The ontology engineer is then responsible for the subsequent implementation.

Two methods can be joined to initiate knowledge acquisition. The 20-question technique [6] is a game-like approach where two persons perform a semi-structured interview. An ontology engineer (as interviewer) has a particular concept in mind, and a domain expert has to guess the concept by asking up to 20 questions. These questions have to be answered with *Yes* or *No*. All questions and answers are written down in protocols. This approach has proven to reveal concepts and relations that are central to the experts' domain [6]. Several groups need to perform these interviews to ensure suitable results. Applying the 20-question technique multiple times per group results in a rich set of protocols used as input for subsequent steps. All concepts which can directly be extracted from these protocols are used as starting point for a card sorting technique [7] . The domain experts structure a set of cards, where each card represents a concept. Without any further input they are free to order these cards. In addition, they are allowed to remove and to add cards. Building and naming clusters among the cards sketch the domain view. Using the 20-question technique in conjunction with card sorting results in a set of concepts, which can be used to generate small- to medium-sized ontologies.

Additionally, the repertory grid technique [8] can be embedded into the knowledge acquisition. In this interview technique a person compares concepts and reasons, based on their properties, why some concepts are similar while others are different. This reasoning gives information about the way a person constructs concepts. Therefore, it offers an individual domain view of the conceptualization and answers the question why the concepts are constructed in a certain way.

2.2. Evaluation

Before ontologies can be released and deployed in applications, the ontology engineers have to ensure that they meet the pre-defined quality standards. An evaluation is performed in order to validate a certain ontology according to the application-specific criteria [9], which can be further divided into technological, structural, conceptual, functional, and user-oriented aspects [10].

Functional parameters, which are related to the intended use of an ontology, are addressed by the proposed similarity-based approach. They indicate if the formalized knowledge suits the intended purpose, and if the used formalization matches the desired application. Accordingly, this facet of an ontology's quality is called *fitness for purpose* within this paper. Other parameters to assess fitness for purpose also include consistency, spelling of terms, and meeting of competency questions based on usage scenarios [11, 12].

3. Semantic Similarity Measurement

Similarity originated in psychology to investigate how entities are grouped into categories, and why some categories (and their members) are comparable while others are not [13, 14]. Similarity gained attention within the last years in computer science and especially in research on artificial intelligence [1]. In contrast to a purely structural comparison, *semantic* similarity measures the proximity of meanings. While semantic similarity can be measured on the level of individuals, concepts, or ontologies, we focus on inter-concept similarity within this paper. In dependence of the (computational) characteristics of the representation language, concepts are specified as unstructured bags of features [15], dimensions in a multi-dimensional space [16], or set-restrictions specified using various kinds of description logics [2, 17, 18, 19]. Besides applications in information retrieval, similarity measures have also been used for ontology mapping and alignment [20, 21]. As the computational concepts are models of concepts in human minds, similarity depends on what is said (in terms of representation) about these concepts.

While the proposed ontology evaluation approach is independent from a particular similarity theory, we focus on the SIM-DL [2] theory here. It has been implemented as a description logics interface (DIG) compliant semantic similarity server. In addition, a plug-in to the Protégé ontology editor has been developed to support engineers during similarity reasoning. The current release[2] supports subsumption and similarity reasoning up to the description logic $\mathcal{ALCHQ}$, as well as the computation of the *most specific concept* and *least common subsumer* up to $\mathcal{ALE}$. A human participants test (carried out using SIM-DL and the *FTO* hydrology test ontology also used within this paper) has proven that the SIM-DL similarity rankings are positively and significantly correlated with human similarity judgments [3].

SIM-DL, which can be seen as an extension of the measure proposed by Borgida et al. [19], is a non-symmetric and context-aware similarity measure for information retrieval. It compares a *search concept* C_s with a set of *target concepts* $\{C_{t_1}, ..., C_{t_m}\}$ from an ontology (or several ontologies using a shared top-level ontology). The concepts can be specified using various kinds of expressive DL. The target concepts can either

[2]The release can be downloaded at http://sim-dl.sourceforge.net/. SIM-DL is free and open source software.

be selected by hand, or derived from the *context of discourse* $\mathcal{C}_d$ [22] which is defined as the set of concepts which are subsumed by the context concept C_c ($\mathcal{C}_d = \{C_t | C_t \sqsubseteq C_c\}$). Hence, each (named) concept $C_t \in \mathcal{C}_d$ is a target concept for which the similarity $sim(C_s, C_t)$ is computed. Besides cutting out the set of compared concepts, $\mathcal{C}_d$ also influences the resulting similarities (see [2, 22] for details).

SIM-DL compares two DL concepts in canonical form by measuring the degree of overlap between their definitions. A high level of overlap indicates a high similarity and vice versa. Hence, also disjoint concepts can be similar. DL concepts are specified by applying language constructors, such as intersection or existential quantification, to primitive concepts and roles. Consequently, similarity is defined as a polymorphic, binary, and real-valued function $C_s \times C_t \rightarrow$ R[0,1] providing implementations for all language constructs offered by the used logic. The overall similarity between concepts is the normalized sum of the similarities calculated for all parts (i.e., subconcepts and superconcepts, respectively) of the concept definitions. A similarity value of 1 indicates that the compared concepts cannot be differentiated, whereas 0 shows that they are not similar at all. As SIM-DL is a non-symmetric measure, the similarity $sim(C_s, C_t)$ is not necessarily equal to $sim(C_t, C_s)$. Therefore, the comparison of two concepts does not only depend on their descriptors, but also on the direction in which both are compared.

A single similarity value (e.g., 0.67) for $sim(C_s, C_t)$ does not answer the question whether there are more or less similar target concepts in the examined ontology. It is not sufficient to know that possible similarity values range from 0 to 1 as long as their distribution is unclear. Consequently, SIM-DL delivers similarity rankings SR. The result of a similarity query is an ordered list with descending similarity values $sim(C_s, C_{t_i})$. The SIM-DL similarity server and plug-in also offer additional result representations which are more accessible for domain experts and users. These include font-size scaling (as known from tag-clouds) or the categorization of target concepts with respect to their similarity to C_s [22].

4. Similarity as Quality Measure in Ontology Engineering

This section introduces semantic similarity as a potential quality indicator. Similarity measurement does not cover all aspects of quality assurance, but rather suggests whether an ontology reflects the domain experts' initial conceptualization and the users' intended application. Consequently, semantic similarity is a candidate for assessing fitness for purpose in ontology engineering. This section describes how the ontology engineering process benefits from the proposed similarity-based approach, and how and where the three types of actors are involved. Figure 2 shows the role of similarity at certain steps of this process.

4.1. Ontology Users: Request

The ontology users request ontologies for a particular domain or application. The ontology engineering life cycle starts (step 1) and ends (step 4) with the user. In both cases the users' task is to evaluate if the available ontology fits the specific purpose, e.g., if it can be deployed in the users' application. In step 1, the ontology users have identified the need for an ontology, and therefore initiate the ontology engineering process by forwarding the request to domain experts.

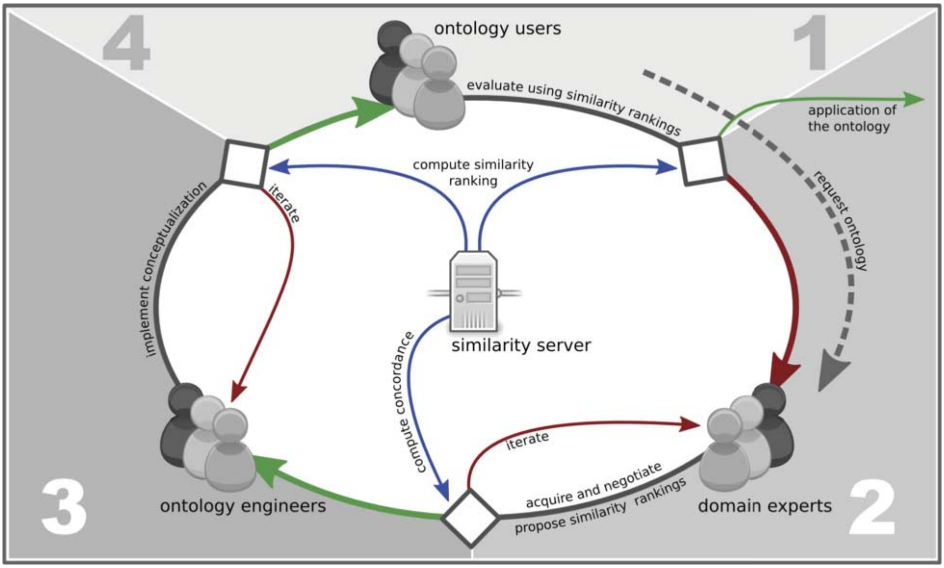


Figure 2. The role of similarity within the ontology engineering process.

4.2. Domain Experts: Knowledge Acquisition and Negotiation

Knowledge acquisition usually depends on domain experts as knowledge sources. Once they receive the users' request (step 1), the domain experts' task is to identify the requirements together with the users. The identification of scope and core concepts [5] is part of the requirements engineering process. We suggest to extend this task with the identification of the search concept C_s and a set of target concepts as well as the creation of similarity rankings SR_{de} between those concepts. These rankings will then be used as an indicator for the quality of an ontology in terms of fitness for purpose.

Current results from the SWING project [23] show that the combination of the 20-question technique and the card sorting method (see section 2.1) provide a way to identify search and target concepts. Five experts from the geology and quarrying domain participated in a knowledge acquisition process. One goal of the meeting was drafting an ontology about the transportation of aggregates. Each of the domain experts is interviewed by an ontology engineer. All concepts appearing in the protocols are used for the card sorting technique. By structuring the cards the domain experts jointly built clusters. One cluster, named "vehicle", contained the concepts Car, $Truck$, $Train$, $Bicycle$, $Pipeline$, $Boat$, and $Plane$. Another cluster ("transportation network") was built by the concepts $Motorway$, $Railroad$, $WaterCourse$, $River$, $Canal$, $Highway$, $Road$, and $Street$. The similarity-based approach to ontology evaluation can now be applied per cluster, i.e., all concepts within a cluster are potential search and target concepts. The concepts appearing most frequently in the 20-question protocols are likely to be most central for the domain. Those are chosen as search concepts, all remaining concepts of a cluster become target concepts. For the "vehicle" cluster this means $Truck$ is selected as C_s and all other concepts of the cluster make up the set of target concepts. $Road$ is selected as the search concept in the "transportation network" cluster.

The approach to identify the core concepts for the similarity rankings, and in particular C_s, is not necessarily crucial for the similarity computation. But we assume that the search concept as well as the target concepts are carefully selected and match the scope of the required ontology. As similarity rankings can be calculated on concepts from several clusters, matching the scope of even large ontologies can be fulfilled. All domain experts propose their individual similarity ranking SR_{de} with regard to the ontology's application area (step 2 in figure 2) using the identified search concept and target concepts. Next, the concordance as measure of the level of agreement between the domain experts' similarity rankings is calculated. A high (and significant) value indicates a common understanding of the core concepts by the domain experts. If the concordance is statistically insignificant (with respect to a pre-defined significance level) for the application, the domain experts' understanding of the core concepts needs to be revised (iteration at step 2). The discussion needs to clarify the definitions of each concept regarding its important characteristics. Afterwards, each domain expert performs a new similarity ranking and the concordance of these new rankings is calculated. Step 2 is repeated until a significant concordance between the similarity rankings is reached.

4.3. Ontology Engineers: Implementing the Experts' Conceptualization

Once there is a significant concordance between the similarity rankings of the domain experts, the information necessary to implement the experts' conceptualization is passed to the ontology engineers (this includes the protocols from the techniques introduced in section 2.1). In addition, an averaged similarity ranking SR_{de} is computed out of the experts' individual similarity rankings. This ranking becomes part of the requirements for the ontology. After the ontology has been developed, the ranking acts as a reference to determine whether the new ontology reflects the domain experts' initial conceptualization. Thus, the averaged ranking is used to evaluate fitness for purpose. The engineers compute a similarity ranking SR_{oe} using the SIM-DL similarity server and Protégé plug-in (see section 3 and figure 4) for the same search and target concepts as used by the domain experts. A significant and positive correlation between the domain experts' and SIM-DL's rankings indicates that the developed ontology reflects the experts' initial conceptualization. In this case, the ontology can be passed to the ontology users for further evaluation (again, using the proposed similarity ranking approach as depicted in step 4 of figure 2). If the similarity rankings do not correlate (or the correlation does not reach the pre-defined strength and significance level), an iteration in the ontology engineering process becomes necessary, i.e., step 3 is repeated until the ontology reflects the domain experts' conceptualization. If, after several iterations, no significant correlation is achieved, it might be necessary to return to the specification phase (step 2) to ensure that all relevant information from this phase is available to the engineers.

Instead of developing a new ontology, the engineers can also decide to investigate an existing ontology beforehand. In this case, the SIM-DL similarity ranking is computed using this ontology and compared to the averaged expert ranking. This requires that the external ontology uses the same concept names, else the engineers have to decide whether other names used in the external ontology can be treated as synonyms for the concepts selected by the experts. Finding synonyms may also benefit from similarity measurement, which is not discussed here but left for further work.

4.4. Ontology Users: Application

After passing all steps of the engineering process, the developed ontology is ready to be deployed. Following figure 1, the ontology users are also involved in the maintenance of the ontology. Up to now, the computed similarity ranking SR_{oe} and the averaged similarity ranking SR_{de} provided by the domain experts are available. But even the best correlation between these two rankings does not necessarily mean that the ontology match the users' view. With the last missing similarity ranking SR_{ou}, we compute the correlation between the rankings SR_{oe} from the engineered ontology and those from the users (step 4). SR_{ou} is also an averaged similarity ranking collected from the ontology users during the maintenance phase, e.g., using questionnaires or user feedback techniques built into the software. The knowledge and therefore also the conceptualization of a particular domain can evolve over time, which means this step has to be performed regularly.

If a significant correlation between SR_{ou} and SR_{oe} exists and does not change over time, it can be assumed that the ontology represents the users' view with respect to the application. A low correlation between SR_{ou} and SR_{oe} might imply that the ontology does, in its current state, not satisfy the users' needs. Re-initiating the ontology engineering life cycle, including the users' similarity rankings, is advisable.

5. Application

This section applies the steps described in section 4 to a set of concepts from four different ontologies to demonstrate our approach. The similarity between these concepts is measured and the resulting ranking is compared to a similarity ranking defined by the authors of this paper acting as domain experts and users, respectively. The concepts and ontologies were chosen to elucidate selected aspects of similarity as a quality indicator. An evaluation involving external domain experts and ontology engineers is left for further work. The used ontologies are excerpts of the hydrology ontology from Ordnance Survey *OS Hydrology*[3], a (OWL-Lite) version of the Alexandria Digital Gazetteer Feature Type Thesaurus *ADL FTT*[4], the *AKTiveSA* ontology[5], and the Feature Type Ontology *FTO Hydrology*[6] developed by the authors for the human participants test described by Janowicz et al. [3]. Figure 3 gives a brief overview over the hydrological concepts within these ontologies; interested readers are referred to the online OWL versions.

In the following we assume that users of a specific hydrology application such as a decision support system for an agency request an ontology. Domain experts analyze the users' requirements and identify core concepts for the new hydrology ontology using the 20-question and card sorting technique. The resulting core concepts are *Canal*, as search concept, and *River*, *Lake*, *IrrigationCanal*, *Ocean*, *Reservoir*, and *OffshorePlatform* as target concepts.

After deciding on the core concepts, and negotiation how these concepts should be specified, each domain expert defines a similarity ranking to express her initial conceptualization. All rankings are performed independently and afterwards compared for con-

[3]http://www.ordnancesurvey.co.uk/oswebsite/ontology/
[4]http://ifgi.uni-muenster.de/ janowicz/downloads/FTT-v01.owl
[5]http://www.edefence.org/ ps/aktivesa/OntoWeb/index.htm
[6]http://sim-dl.sourceforge.net/downloads/

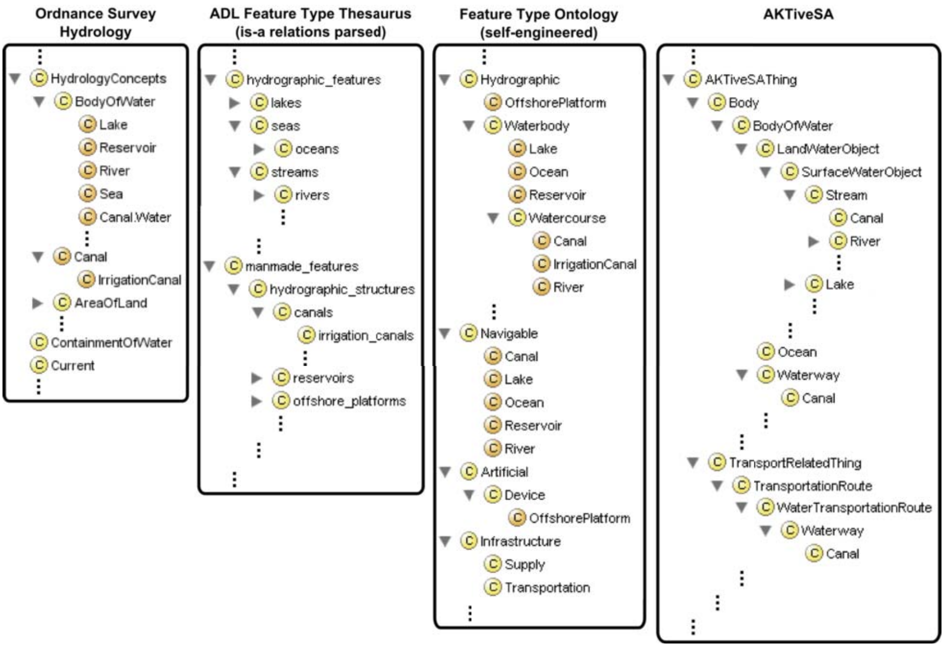


Figure 3. Overview of the four ontologies used for similarity measurement.

cordance using Kendall's coefficient of concordance W as measure of the level of agreement between the domain experts. In case of the authors' rankings this yields $W = 0.77$, which is a statistically significant result (using a significance level of 0.05).

The averaged similarity ranking by the domain experts is passed on to both the users and the ontology engineers. The users might refine the requirements if the domain experts' rankings do not match the users' expectations. The ontology engineers use these rankings for later verification of the implemented ontologies. The computed similarity rankings are then compared with those produced by the domain experts.

To measure similarity and compare the resulting rankings for correlation, the SIM-DL similarity server is used in conjunction with an extended version of the Protégé similarity plug-in. As depicted in figure 4 the extension offers a tab for estimating the similarity between the search and the target concepts using sliders. The resulting ranking and the similarity values are compared to the results obtained from the SIM-DL server.

The Protégé extension shown in figure 4 not only allows for specifying a ranking of concepts, but also for expressing a quantitative distance between these concepts. However, different people (i.e., domain experts) use different (cognitive) similarity scales and distributions [3]. Hence, the interpretation of the absolute similarity values and distances between them is difficult. Consequently, this paper focuses on similarity rankings.

The *FTO Hydrology* ontology is supposed to be the ontology developed by the ontology engineers based on the experts' conceptualization. Figure 4 shows the resulting chart and correlation based on the averaged similarity ranking of the experts and the results computed by SIM-DL for the *FTO Hydrology* ontology. As shown in table 1, there is a positive ($r_s = 0.94$) and significant ($p < 0.05$) correlation between both rankings. These

Table 1. Similarity rankings for the used ontologies with respect to *Canal* as search concept.

Similarity ranking to Canal	River	Irr. Canal	Reservoir	Lake	Ocean	Off. Platform	Correlation◇
Experts' Ranking	1	2	4	3	5	6	—
ADL FTT Ranking	3	1	2	3	3	2	0.06
OS Hydrology Ranking	3	1	4	2	4⋆	—	0.67
FTO Hydrology Ranking	1	2	3	4	5	6	0.94
AKTiveSA Ranking	1	—	2	2	3	—	0.95

◇: Spearman's rank correlation r_s measured to the experts' averaged ranking.
⋆: The concept *Sea* is used as no concept named *Ocean* is available in the ontology.

results indicate that the *FTO Hydrology* ontology reflects the experts' conceptualization. The ontology is then passed to the users for further evaluation.

The users evaluate the received ontology using their similarity rankings in order to investigate if the ontology can be deployed into the final hydrology application. Otherwise, the users can initiate a new iteration cycle starting again with the domain experts.

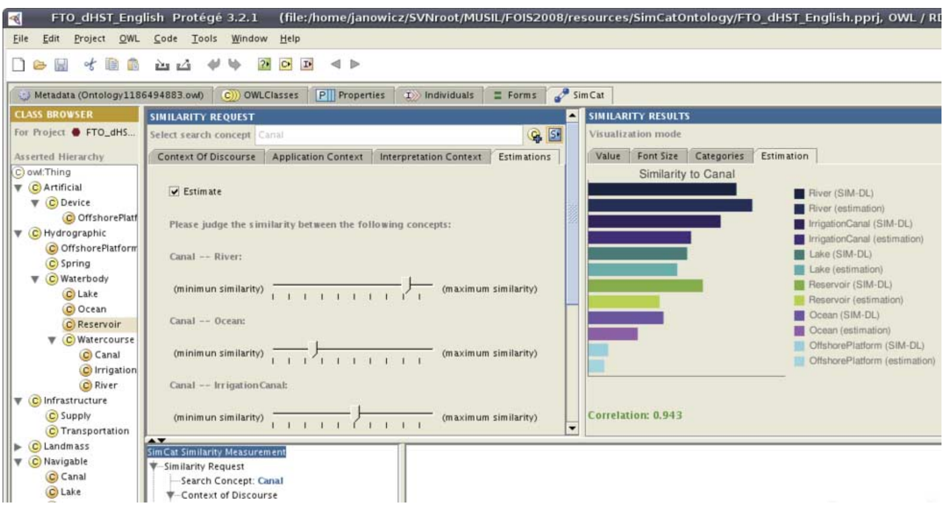


Figure 4. The extended SIM-DL Protégé plug-in with the new estimation tab (compare to [2, 22]).

It is reasonable to assume that ontology engineers first check for existing external ontologies before developing a new one. They compare the SR_{de} ranking of the experts with those from the external ontologies (in our case the *ADL FTT*, *OS Hydrology*, and *AKTiveSA* ontologies) to evaluate their fitness for purpose.

Unlike the self-engineered *FTO Hydrology* ontology, table 1 shows that the ranking obtained from the *ADL FTT* ontology does not correlate with the experts' ranking. For instance, the *ADL FTT* concept *offshore platforms* is ranked in second place, and hence above *rivers*. This can be explained with the single-inheritance structure used within this ontology, i.e., a concept cannot be a direct subconcept of two different concepts. As a consequence, the top-level distinction between *hydrographic features* and *manmade features*, and the definition of the concept *hydrographic structures* as a subclass of *manmade features*, implies that all concepts classified as *hydrographic structures* are considered manmade, but not *hydrographic features* (see figure 3). As the *ADL FTT* ontology was derived from automatically parsing the thesaurus, the subsumption relationship is the only one which could be used to measure conceptual overlap (and hence similarity). Consequently, the similarity between concepts which are not beneath a common superconcept (such as *canals* and *rivers*) is low. In contrast, sharing the same superconcept increases similarity as for *canals* and *offshore platforms*. Both are *hydrographic structures*[7] and *manmade features*. Such view does not reflect the experts' initial conceptualization, and therefore the ontology cannot be used for the hydrology application (or requires substantial modification).

A test run for the second external ontology, an excerpt from *OS Hydrology*, shows a positive ($r_s = 0.67$) but insignificant correlation to the experts' ranking. This is due to several reasons: first, the concepts *OffshorePlatform* and *Ocean* are not part of this ontology which decreases the number of ranked concepts decisively. Second, the implemented concepts do not meet the experts' conceptualization. As described in section 4.3 the *OS Hydrology* concept *Sea* is chosen as potential alternative for *Ocean* within this example. The surprising result that *Lake* is more similar to *Canal* than *River* can be explained as follows. First, while *River*, *Lake*, *Sea*, and *Reservoir* are subconcepts of *BodyOfWater*, *Canal* and *IrrigationCanal* are not (see figure 3). However, there is a subconcept of *BodyOfWater* called *Canal.Water* that comprises some of the intended characteristics missing in *Canal* (e.g., being navigable). Second, in contrast to *Canal* and *Lake*, the definition of *River* does not contain a value restriction for being connected to other bodies of water.

The *AKTiveSA* ontology represents the case where a high correlation ($r_s = 0.95$) indicates that the concepts reflect the experts' conceptualization. However, not all concepts are defined in the ontology, and hence the correlation is statistically insignificant. No candidate concepts for *OffshorePlatform* and *IrrigationCanal* were found. In this case, the engineers can decide to extend the ontology with the missing concepts and recalculate the correlation.

Summing up, the application of similarity as quality indicator points to the following benefits and shortcomings. Similarity helps to assess if developed ontologies reflect the intended conceptualizations of experts and users. Simplicity is a desired prerequisite for an evaluation method in order to be adopted by non-technical experts. As similarity is grounded in cognition, the cognitive effort imposed on actors to produce similarity rankings is low. This is especially important for non-technical domain experts and end-users lacking formal background on description logics. Therefore, similarity rankings provide the engineer with the possibility to integrate the users and experts during the implementation phase. SIM-DL compares concepts for overlapping definitions. This does not guarantee that these definitions are relevant for the particular application. For an external ontology this may cause a correlating similarity ranking, although the definitions focus on other applications (such as recreation instead of navigation). Therefore, SIM-DL allows to set the context of discourse (see section 4.3) to enforce particular concept definitions. Finally, similarity does not answer the question how concepts differ. To improve the expressivity of similarity as quality indicator, it should therefore be combined with *difference* operations as proposed by Teege [24].

[7]Which is surprising as the thesaurus defines hydrographic structure as "constructed bodies of water".

6. Conclusions and Further Work

Ontology engineering and similarity reasoning have only been remote cousins so far. We have shown in this paper that semantic similarity rankings founded in formal ontology can support the ontology engineering process. In particular, they serve as measures for how accurately an ontology matches the conceptualizations held by ontology engineers and users. Our approach is orthogonal to ontology engineering methods and can be incorporated into any of them. The contributed plug-in to the Protégé ontology editor serves this purpose and has been successfully tested in a scenario with hydrological information. While we focused on the simplified hydrology example here, a more sophisticated scenario from quarry mining involving external domain experts and users is under development in the SWING project (see section 4.2).

Our main contribution is toward the problem of quality assurance for information system ontologies. The simple idea to compare similarity rankings of concept specifications in natural language (produced by domain experts or users) with those of concept specifications in DL (produced by ontology engineers) represents an effective way of assessing how closely the stated constraints on meaning match the intended meaning.

Our method is rooted in formal ontology, as the semantic similarity rankings are based on a similarity theory that accounts for concept specifications instead of a purely syntactical measure. The similarity theory and its application have been developed with theoretical foundations in psychological literature on similarity and the logics to express them. All similarity measures crucially depend on the representation chosen for the compared concepts. A solid grounding in formal ontology can therefore be expected to improve the match between human and computational similarity rankings. This has been shown to be the case by Janowicz [3]. In this paper, we have used a non-symmetric similarity measure. SIM-DL also supports symmetric similarity; further work should investigate which approach fits better for quality assessment.

Beyond the formal foundations, the iterative engineering model involving three actors (domain expert, knowledge engineer, user) represents a way toward more realistic knowledge acquisition and management scenarios. The social nature of these processes, particularly the fact that specifications of conceptualizations are negotiated among the participants, is ideally supported by a concise and transparent quality measure (which is also easy to use) such as the match between similarity rankings.

From a formal ontology point of view, a benefit of our approach is that it can reveal incomplete concept definitions. For instance, in *AKTiveSA*, canals differ from other bodies of water by also being transportation routes. Length is a characteristic of transportation routes, but not automatically of rivers, since it does not apply to all bodies of water. The lacking length of rivers has a negative impact on the similarity value of *Canal* to *River*. It indicates to the ontology engineer that, from a certain perspective, the ontology is incomplete or inhomogeneous.

7. Acknowledgements

The comments from three anonymous reviewers provided useful suggestions to improve the content and clarity of the paper. This work is funded by the SimCat project granted by the German Research Foundation (DFG Ra1062/2-1) and the SWING project which is funded by the European Commission (FP6-26514). We also acknowledge the contribution of five domain experts from the French geological survey, BRGM (Bureau de Recherches Géologiques et Minières).

References

[1] E. L. Rissland. AI and similarity. *IEEE Intelligent Systems*, 21(3):39–49, 2006.

[2] K. Janowicz, C. Keßler, M. Schwarz, M. Wilkes, I. Panov, M. Espeter, and B. Baeumer. Algorithm, Implementation and Application of the SIM-DL Similarity Server. In *Second International Conference on GeoSpatial Semantics (GeoS 2007)*, number 4853 in Lecture Notes in Computer Science, pages 128–145. Springer, 2007.

[3] K. Janowicz, C. Keßler, I. Panov, M. Wilkes, M. Espeter, and M. Schwarz. A Study on the Cognitive Plausibility of SIM-DL Similarity Rankings for Geographic Feature Types. In *11th AGILE International Conference on Geographic Information Science (AGILE 2008)*, Lecture Notes in Computer Science, Girona, Spain, 5-8. May 2008; forthcoming. Springer.

[4] H. Happel and S. Seedorf. Applications of ontologies in software engineering. In *International Workshop on Semantic Web Enabled Software Engineering (SWESE'06)*, Athens, USA, November 2006.

[5] A. Gómez-Pérez, O. Corcho, and M. Fernández-López. *Ontological Engineering: with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web.* Springer, July 2004.

[6] S. Schade, P. Maué, J. Langlois, and E. Klien. Ontology engineering with domain experts - a field report. In *European Geosciences Union - General Assembly*, 2008.

[7] N. J. Cooke. Varieties of knowledge elicitation techniques. *Int. J. Hum.-Comput. Stud.*, 41(6):801–849, 1994.

[8] D. Jankowicz. *Easy Guide to Repertory Grids*. Wiley, London., 2004.

[9] A. Gangemi, C. Catenacci, M. Ciaramita, and J. Lehmann. A theoretical framework for ontology evaluation and validation. In *Semantic Web Applications and Perspectives (SWAP) – 2nd Italian Semantic Web Workshop*, 2005.

[10] S. Schade, E. Klien, P. Maué, D. Fitzner, and W. Kuhn. Ontologies in the SWING-Project: Report on Modelling Approach and Guideline - Deliverable 3.2, 2008.

[11] M. Grüninger and M. S. Fox. Methodology for the design and evaluation of ontologies. In *Proceedings of IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.

[12] M. Uschold and M. Grüninger. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 10(2):92–155, 1996.

[13] R. L. Goldstone and J. Son. Similarity. In K. Holyoak and R. Morrison, editors, *Cambridge Handbook of Thinking and Reasoning*. Cambridge University Press, 2005.

[14] D. Medin, R. Goldstone, and D. Gentner. Respects for similarity. *Psychological Review*, 100(2):254–278, 1993.

[15] A. Rodríguez and M. Egenhofer. Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science*, 18(3):229–256, 2004.

[16] M. Raubal. Formalizing conceptual spaces. In A. Varzi and L. Vieu, editors, *Formal Ontology in Information Systems, Proceedings of the Third International Conference (FOIS 2004)*, volume 114 of *Frontiers in Artificial Intelligence and Applications*, pages 153–164. IOS Press, Amsterdam, NL, 2004.

[17] C. d'Amato, N. Fanizzi, and F. Esposito. A semantic similarity measure for expressive description logics. In *CILC 2005, Convegno Italiano di Logica Computazionale*, Rome, Italy, 2005.

[18] R. Araújo and H. S. Pinto. Semilarity: Towards a model-driven approach to similarity. In *International Workshop on Description Logics*, volume 20, pages 155–162. Bolzano University Press, June 2007.

[19] A. Borgida, T. Walsh, and H. Hirsh. Towards measuring similarity in description logics. In *Proceedings of the 2005 International Workshop on Description Logics (DL2005)*, volume 147 of *CEUR Workshop Proceedings*. CEUR, Edinburgh, Scotland, UK, 2005.

[20] A. Maedche and S. Staab. Measuring similarity between ontologies. In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, number 2473 in Lecture Notes in Computer Science, pages 251 – 263. Springer, 2002.

[21] W. Sunna and I. F. Cruz. Structure-based methods to enhance geospatial ontology alignment. In *GeoSpatial Semantics, Second International Conference, GeoS 2007*, number 4853 in Lecture Notes in Computer Science, pages 82–97. Springer, 2007.

[22] K. Janowicz. Kinds of contexts and their impact on semantic similarity measurement. In *5th IEEE Workshop on Context Modeling and Reasoning (CoMoRea) at the 6th IEEE International Conference on Pervasive Computing and Communication (PerCom'08)*, Hong Kong, March 2008. IEEE Computer Society.

[23] D. Roman and E. Klien. *The Geospatial Web. In the Advanced Information and Knowledge Processing Series*, chapter SWING - A Semantic Framework for Geospatial Services, pages 229–234. Springer, 2007.

[24] G. Teege. Making the difference: A subtraction operation for description logics. In J. Doyle, E. Sandewall, and P. Torasso, editors, *4th International Conference on Principles of Knowledge Representation and Reasoning (KR'94)*, pages 540–550, Bonn, Germany, 1994. Morgan Kaufmann.

This page intentionally left blank

3. Objects

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-109

A New Classification of Collectives

Zena WOOD, Antony GALTON
School of Engineering, Computing and Mathematics, University of Exeter, UK

Abstract. We are surrounded by collective phenomena, with examples existing on many levels of granularity. Despite our frequent experiences of such phenomena they seem to have been largely ignored within the field of ontology. In this paper, existing ontologies are examined to determine the extent to which they can adequately represent collective phenomena, and are found wanting in a number of important respects. Our goal is to find good ways of representing collective phenomena in a way which does justice to the often subtle relationship that exists between the view of a collective as a whole and its constitution as a plurality of individual participants. An important prerequisite for this is to determine the range of variation that exists within the broad class of collectives. A number of example collective phenomena have been studied to extract appropriate classification criteria. The results from this study are used to produce a new typology of collectives which is intended to establish a basis for the adequate treatment of collectives within an ontology. The paper concludes with a set of further research questions that have been raised during the development of the classification.

Keywords. Collective, collection, social group, dynamic phenomena

1. Introduction

We are surrounded by collective phenomena, with examples existing at many different levels of granularity. At a molecular level water can be seen as consisting of a collection of molecules; at a somewhat higher level, a living tissue is a collection of cells. At a higher level still there exist phenomena such as traffic jams, crowds and herds. We are likely to form part of a collective at least once every day of our lives whether it is that traffic jam as we make our way to work or the queue of people at the bank. There is increasing use of information systems to model the behaviour of various kinds of collective such as crowds [1] and carnivals [2]. Despite our frequent experiences of collective phenomena their importance seems to have been largely unrecognised within the field of ontology.

Previous research into collective phenomena such as crowds [1] and biological collectives [3,4] appears to have focussed on the level of individuals. At this level it is a challenge to extract information about the behaviour of the collective as a whole; yet it is the latter which makes such phenomena most meaningful to us. Consider an example such as a crowd. Questions such as 'What was the overall movement of the crowd between t_1 and t_2?' are from many points of view much more meaningful than the movements of the various individual members of the crowd between t_1 and t_2. The same can be said of phenomena such as a queue, a traffic jam, a procession and a marathon.

There is an ongoing debate within the philosophical literature, which looks at whether the behaviour of a collective is more than just the aggregated behaviour of its individual members [5]. We do not wish to add to the literature on this debate and instead, like [6,7], recognise that collectives form a class distinct from that of their individual constituents. We assign attributes to the collective that could not be assigned to their individual members. For example, a football team can win a match but the individuals cannot. Within our natural language we tend to refer to, discuss and reason about collective phenomena as single units and do not really focus on the fact that they are made up of a number of individual sub-units. Why we do this is outside the scope of this paper. However, we do believe that any framework which is used to model collective phenomena should allow the phenomena to be modelled as we think of them (i.e. as one single unit).

Some researchers have begun to recognise and try to represent collective phenomena as single units but it appears most of this research focuses only on a particular type of collective phenomena such as intentional collectives [7]. There seems to be no research into building a framework that could adequately model a number of different types of collectives including those that are biological, social and even epistemic. The ultimate goal of the research reported in this paper is not only to find good ways of representing a wide variety of collective phenomena but also to find a representation that does justice to the often subtle relationship that exists between the view of a collective as a whole and its constitution as a plurality of individual participants. An important prerequisite for this is to determine the range of variation that exists within the broad class of collectives. To this end, part of our investigation will need to be directed towards the classification of these phenomena.

The purpose of this paper is to highlight the need for a new ontology that can adequately model a wide range of collective phenomena and to propose a new classification of criteria that can be used as a basis for such an ontology. Existing ontologies, including those proposed to represent a subset of collective phenomena [7], are examined to determine the extent to which they can adequately represent collective phenomena (Section 2), and are all found wanting in a number of important respects (Section 2.4). In order to develop a new ontology that could adequately represent collective phenomena, we must understand what we need to model and therefore the different collectives that exist and how they relate to each other. A number of example collective phenomena have been studied to extract appropriate classification criteria and these criteria are discussed, resulting in a new typology of collectives which is intended to establish a basis for the adequate treatment of collectives within an ontology (Section 3). We illustrate the typology by applying it to a diverse range of example collectives (Section 3.5). The paper concludes with a set of further research questions that have been raised during the development of the classification and will need to be answered in order to develop our new ontology (Section 4).

2. Ontologies

The use of ontologies is widespread in a number of fields including GIS, Artificial Intelligence and Computational Linguistics [8]. Since ontologies are seen as the Semantic Web's 'basic infrastructure' [9] it is becoming increasingly important to ensure that ontologies can reflect the social reality that we live in. As discussed above, our lives are pervaded by collective phenomena but this does not appear to be reflected in existing ontologies. This section examines two existing ontologies (DOLCE and BFO) and an ontology developed to model a specific type of collective phenomena [7]. Although each ontology is found to possess some tools and relations that allow them to model features of collective phenomena, they are found wanting in many areas.

2.1. DOLCE

Part of the WonderWeb Foundational Ontologies Library, DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) has been developed to represent a very general domain. Limited to particulars, DOLCE tries to capture the 'ontological categories' that underly human common sense thereby giving it a cognitive bias [9]. However, even though in many cases collectives only exist when we deem them to, there is little support for them in DOLCE [7].

A number of relations defined in DOLCE could be used to help represent collective phenomena. 'Parthood' and 'temporary parthood' could be used to indicate when an individual is a member of a collective. If an individual is only a member for a short time, for example an individual in a queue, the 'temporary parthood' relation could be used to indicate that the relation exists only for a certain time period.

Within natural language, we say that a member participates in a collective. However, [9] notes that owing to the 'strict definition' between perdurants and endurants within DOLCE, 'participation' has to be something more than just parthood. Would both relations be needed in order to adequately model collective phenomena? If not, which relation would be more suitable: parthood or participation?

A collective cannot exist without its members which would suggest the use of the relation 'dependence'. However, what about those collectives where the members can change? Any dependence relation would need to be relativised to time, for example by inclusion of a temporal argument in the formal expression of the relation.

As with all ontologies, the basic categories that are defined must be sufficient to accurately model the domain of discourse. The "leaf" categories used in DOLCE could adequately model the members of a collective: *Agentive Physical Object*, *Non-Agentive Physical Object*, *Social Agent* and *Non-Agentive Social Object*. However, the collective as a whole could not be modelled. DOLCE does contain the category *Arbitrary Sum* but this would be inappropriate for many collectives since it implies that there is not a real reason to consider a group of individuals as a collective.

2.2. BFO

Like DOLCE, BFO (Basic Formal Ontology) is not designed to be a "universal" ontology capable of dealing with multiple domains. In order to use BFO in a particular domain it must be integrated with information specific to that domain [10]. BFO has been designed to represent entities and relations that exist in a 'mind-independent' world [10]; therefore, those collectives which only exist by virtue of our deeming them to could not be represented in BFO. Despite this, BFO uses tools and relations that could prove useful in an ontology capable of modelling collectives.

BFO aims to combine the different perspectives of the three- and four-dimensionalist views by combining two different ontologies: SNAP, which represents endurants, and

SPAN, which represents perdurants [11]. Owing to this combination entities can be represented on two different levels of granularity [11]. As already discussed, collective phenomena exist on two granular levels: at a low level the individuals and at a higher-level the collective. For example, at a molecular level we view a number of individual water molecules but when viewed at a higher-level the water molecules form a body of water. The static properties of the higher-level view may depend on dynamic aspects of the elements discernible at the lower level (e.g., a static water body consists of many molecules in rapid motion, and its temperature reduces to the kinetic energy of that motion). Thus both SNAP and SPAN are implicated in a subtle interplay.

Within the SNAP ontology, BFO has a category *object aggregate*, 'an independent continuant entity that is a mereological sum of separate objects' [10]. Could this be used to model a collective as a single unit? The problem is how to capture the dependence of the collective on its members within the terms provided by the BFO ontology. Simply shifting between granular levels within the SNAP ontology does not seem to do justice to the true nature of this dependence, which at least in many cases must involve SPAN as well as SNAP.

2.3. An Ontology Designed to Represent Collective Phenomena

To try and overcome the problem of representing collective intentionality, Bottazzi *et al.* [7] have looked at the representation of collectives where the members can be ascribed intentionality. This analysis has resulted in an ontology designed to model this particular subset of collective phenomena. Although their ontology is not designed to represent collectives in general, some of their ideas could be incorporated or extended within an ontology that was able to represent a wider range of collective phenomena. However some aspects of the ontology would not be suitable.

- They state that a collective must have at least two members, but is this always true? Consider a queue of people at an ATM; the number of participants within that queue can go down to one and possibly then increase when more participants join at the end. During the whole of this period, we still consider the queue as an enduring entity. For more discussion on this point see section 3.1.
- Within their ontology, every collection is covered by at least one 'role' where a 'role' could be something as simple as 'being a member'. At what point would an individual be deemed to become part of a collection?
- The terminology that they use appears to be very similar to words used daily in the English language but since they are used differently by different people, could be found to be confusing if not ambiguous. For example, 'plan', 'description' and 'concept'. We would wish to ensure that this would not happen within any new ontology that we developed by the introduction of new terms such as 'collectivity' (see section 3). Although some may say that the introduction of more terms will add to the confusion, we believe that in this instance it is appropriate.

2.4. Why are Existing Ontologies Inappropriate?

Existing ontologies provide some useful ideas that could be incorporated into an ontology capable of representing a wide range of collective phenomena: DOLCE's use of relations such as 'temporary parthood' and BFO's allowing entities to be represented on two levels of granularity. Instead of creating a new ontology, Bottazzi *et al.* [7] have extended the existing ontology DOLCE to produce D&S. Further research would be needed to see if this idea is indeed plausible in order to incorporate collective phenomena. However, initially this option seems unlikely owing to the granularity issue (i.e. the ontology must exist on two levels of granularity).

Although existing ontologies such as DOLCE and BFO do possess some tools that allow features of collective phenomena to be represented, this is only a very small set of the total features. Relations such as 'temporary parthood', 'participation' and 'dependence' only allow us to model which individuals are members of a collective. There appears no way to model:

- if the members can change or the cardinality of members can change without the collective ceasing to exist (see section 3.1);
- how the motion of the collective relates to that of its members (see section 3.2);
- the reason we deem it to be a collective (section 3.3);
- the roles that can be played by different members of a collective (section 3.4);
- attributes that can be ascribed to the collective but not to the individual members (section 4).

Despite being capable of representing only a particular subset of collective phenomena, [7] improves on this in some way but not to the extent needed.

Ultimately, there is no way, in any of the three existing ontologies that have been examined, to model (i) the collective as a whole, (ii) the relation that exists between a collective and its individual members, and (iii) attributes that can be applied to the collective but not to its individual members.

3. Criteria for Classifying Collective Phenomena

In the literature a series of terms have been used to refer to different types of phenomena that consist of more than one individual: 'collection', 'collective', 'group' and 'social group'. We believe that these words have become overused, and in many cases it is not made clear how or indeed whether these terms refer to different types of entity. For this reason we would like to introduce the new term 'collectivity', to be used as a concrete count noun, referring to what has previously been called a 'collective phenomenon'.

A collectivity is specified by means of a time-dependent membership relation which is assumed to be crisp (i.e., not fuzzy). More formally, the predicate $Collectivity$ is defined in terms of the primitive predicate $Member$ as follows:

$$Collectivity(X) =_{\text{def}} \forall t \exists S \forall x (Member(x, X, t) \leftrightarrow x \in S).$$

Here S is the set of members of the collectivity at time t; this may, but need not, be different at different times.

If this is the most general notion of collectivity, the question arises as to what identifiable subclasses exist which can form part of a comprehensive classification of collectivities. In order to build an adequate ontology or framework to represent a wide variety of collectivities we must know and understand how they relate to each other. We have studied a number of collectivities and the results from this have led us to classification criteria based on the following areas:

- membership,
- location,
- coherence,
- differentiation of roles.

We have selected these criteria because they have arisen as key distinguishing features of the various different collectivities we have considered. We believe they provide a useful basis for classifying collectivities, and we will illustrate the criteria by means of examples of particular kinds of collectivities. We do not however, wish to be dogmatic about the allocation of any particular example to a particular class, as this may often be dependent on delicate issues where there is ample scope for disagreement. For example, the criteria involving intentionality may or may not be applicable to flocks of animals, depending on whether the animals in question can be said to have intention. This is not a matter on which we wish to take a stand in this paper.

3.1. Membership

This criterion refers to both the cardinality and identity of the members (i.e. the individuals) that make up the collectivity. The main division is between those collectivities whose members are the same at all times during the lifetime of the collectivity (*constant membership*) and those which can have different members at different times (*variable membership*) [12]. A second division is between collectivities of constant cardinality (e.g. a string quartet, a football team, a jury), and those where the cardinality may vary with time (e.g. a crowd, a queue, an audience). Amongst collectivities of variable cardinality, some cease to exist when the cardinality drops below two (e.g. a crowd, a flock), but others can survive this depletion (e.g. a queue). In the case of collectivities with fixed cardinality, we can consider also whether the identity of the membership must be fixed too. An example of a collectivity with fixed membership would be all bank notes with serial numbers from y to z; more normally, turnover of members is permitted, as e.g. in the case of a football team or a string quartet. However, this does depend on time scale. A quartet considered for the duration of a single concert season generally has fixed membership; over a number of years replacement may occur. In general, the properties we ascribe to a collectivity, and thus the way we classify it, may be quite sensitive to such shifts of temporal scope.

[M1] Constant membership
[M2] Variable membership

  [M2a] Constant cardinality
  [M2b] Variable cardinality

    [M2bi] Cardinality necessarily > 1
    [M2bii] Cardinality may reduce to 1

3.2. Location

To classify collectivities from a spatial point of view we could look not only at the location of the collectivity but also at the locations of its members, and how they are distributed through the region of space that the collectivity occupies. The location of a collectivity could be fixed (i.e. static) or variable (i.e. dynamic). If the location of a collectivity is fixed, then the position of its individual constituents could either be fixed or variable. Consider a forest. The location of both the collectivity itself (i.e. the forest) and its individual members (the trees) are fixed, at least over a period of years rather than decades or centuries. In comparison, the location of a queue is essentially fixed but the positions of its members are variable since each individual will move towards the front of the queue. If the location of a collectivity is variable then the position of its members could also either be fixed or variable. For example, a Mexican wave can be seen as a collectivity which migrates through a set of fixed individuals.

If the location of the collectivity and its individuals are both variable then the classification can be split further according to how the motion of the individuals is related to that of the collectivity as a whole. Of course, the latter motion ultimately arises from the sum of the former motions; but compare the case of a platoon on the march, where the path followed by the platoon as a whole is essentially the same as that followed by each individual soldier, with a crowd where individuals are milling about in seemingly random directions but the crowd as a whole gradually drifts off in one particular direction.

[L1] Location of collective is fixed.

- [L1a] Location of members is fixed.
- [L1b] Location of members is variable.

[L2] Location of collective is variable.

- [L2a] Location of members is fixed.
- [L2b] Location of members is variable.
  - [L2bi] Motion of individuals and collectivity is coordinated.
  - [L2bii] Motion of individuals and collectivity is not coordinated.

Where the members of the collectivity display a coordinated motion that is shared by the collectivity as a whole, this coordination may be what pronpts us to recognise the collectivity as a collectivity; but typically the coordination will arise from some anterior cause which is the true ground of the collectivity's being. When the motions are not so coordinated it is in any case clear that the motions of the members cannot provide such a ground. The collectivity may owe its coherence to a number of factors, which we shall examine in the next section.

3.3. Coherence

This criterion refers to the types of behaviour that are exhibited by the collectivity. Does the collectivity exhibit coherent behaviour and if so, what is the source of this coherence? A collectivity exhibits coherent behaviour if the behaviour of the collective is more than the aggregated behaviour of its individual members. See section 4 for more discussion on we mean by this. A clear link can be seen here to the spatial criteria described above.

Our investigation highlighted two common sources of coherent behaviour in collectivities: purpose and cause. A purposive collectivity is held together by some goal or purpose. Purposive collectivities can be split into two further categories: those where the collective behaviour arises from a shared collective purpose (i.e., a common intention to achieve some goal collectively), and those where it arises from the simultaneous exertion of individual purposes (i.e., concurrent intentions to achieve some goal individually). Examples of the former kind include processions and committees; examples of the latter kind include queues and most kinds of crowd. In a queue, each member wishes, as an individual, to reach the front, where it will then be able to carry out whatever transaction forms its motive for joining the queue. By contrast, in an orchestra, the goal of each individual player is to make his or her designated contribution to the overall performance, on the understanding that all the other players have goals coordinated with this. The distinction here corresponds to that between 'we-intention' and 'I-intention' [13], or 'we-mode' and 'I-mode' [14].

The second source of coherent behaviour arises when the members of the collectivity result from some non-purposive cause or causes; the causes may be either internal or external to the collective. For example, raindrops do not interact with each other but instead are all responding to the earth's gravitational attraction as an external cause. An external cause can in fact arise from the intentionality of an agent external to the collectivity (controlling it) — e.g, a stamp collection owes its coherence to the intention of the collector in bringing the stamps together and ordering them in albums, etc. Perhaps a flock of sheep being marshalled by a sheepdog responding to signals from the shepherd constitutes another example of this. In contrast, a star cluster is kept together by the mutual gravitation of its constituent stars; this is a clear case of internal cause.. It is important to note that it is not always clear whether the source of coherent behaviour is due to purpose or cause; very often it is a mixture of both. Consider a group of penguins; they huddle together in the winter months to keep themselves warm and it is the interactions between the penguins which are keeping them together as a collective. In this respect, a huddle of penguins is a purposive collectivity but this is triggered by the external cause of winter.

[C1] Cause

  [C1a] External cause
  [C1b] Internal cause

[C2] Purpose

  [C2a] Individual purposes
  [C2b] Collective purpose

3.4. Differentiation of Roles

Our next classification criterion concerns the roles played by the individual members in a collectivity; in particular, do all the individuals play the same role within the collectivity or are they differentiated by the roles they play? For example, the individuals within a crowd are not differentiated by role but the individuals within a string quartet are: first violin, second violin, viola and cello. If they play different roles we can look at how these different roles are organised within the collectivity.

There are very many different ways in which a collectivity may be structured in terms of the roles played by its members, but amongst those collectivities in which there is at least some differentiation of roles, we single out here three types that seem to be particularly important.

One frequent pattern is where a single individual or a small number of individuals play a leadership role and the remaining members of the collectivity have no special role other than to be followers (they are the ‘rank and file’). We shall call this the ‘leader-follower’ or ‘oligarchic’ type.

At the other extreme we find collectivities in which each member has its own unique role to play. A clear example is a string quartet, in which each individual plays one of four roles: violin I, violin II, viola or cello. Other examples of collectivities which follow this model are many committees, and a cabinet (in the political sense). We shall call this the ‘individualistic’ type.

Intermediate between these two types is what we shall call the ‘partitioned’ type: here there are a small number of differentiated roles each played by many members. In this case we can speak of *subcollectivities* — a subcollectivity consists of the members of the collectivity who play a particular role, for example, the string section of an orchestra. Being collectivities in their own right, the subcollectivities could also be classified by the differentiation of role, and this subdivision may in some cases be continued recursively, leading to complex hierarchical structures. In an orchestra, we might first identify four main subcollectivities, the string, woodwind, brass, and percussion sections, and within each of these identify further subcollectivities such as, within the string section, the first violins, second violins, violas, cellos, and basses. Each of these has a section leader but otherwise no further differentiation of roles, thus following the oligarchic pattern; but within the brass, wind and percussion sections the subcollectivities typically are of individualistic type, so that one distinguishes e.g., clarinet I, clarinet II, and clarinet III.

It is important to note that the three types of role differentiation presented here are not intended to be exhaustive, and there are clearly gradations between them leading to many other possible models.

[R1] Members of collectivity are not differentiated by role.
[R2] Members are at least partly differentiated by role.

- [R2a] Oligarchic
- [R2b] Partitioned
- [R2c] Individualistic

Dynamically changing roles (e.g the individuals in a queue all take turns to become the head of the queue) have not been considered here. Further research is be required to incorporate this into our current classification system. Relevant related work is [7] and [15].

3.5. Illustrative Examples

A number of example collectivities are used below to illustrate the classification system introduced in Section 3. These examples provide several instances of the point made above about the importance of the time scale over which a collectivity is considered to exist.

- A football team:
  - ∗ During a match - [M1, L1b, C2b, R2b/c]
  - ∗ Over a long period - [M2a, L2a/b, C2a/b, R1]
- An orchestra or choir
  - ∗ During a performance - [M1, L1a, C2b, R2b]

  * Over a period - [M2bi, L2b, C2a/b, R2b]

- A quartet
  * During a performance - [M1, L1a, C2b, R2c]
  * Over a period - [M2a, L2b, C2a/b, R2c]

- Queue at a post office - [M2bii, L1b, C2a, R1]

- Raindrops in a rainstorm - [M2bi, L2b, C1a, R1]

- Trees in a forest
  * Over a short period - [M1, L1a, C1a/b, R2b]
  * Over a long period - [M2bi, L1a, C1a/b, R2b]

4. Further Work

Our view is that by taking the approach of systematically surveying a diverse range of criteria appropriate for the classification of collectivities, we have opened the way to a rich field of enquiry that as yet has barely begun to impinge on the field of formal ontology. During the development of the typology of collectivities a number of questions arose, all of which will necessitate further investigation:

- An initial investigation [16] has been carried out which looks as the relationship between our classification of collective phenomena and a classification of movement patterns [17]. This investigation has shown that a number of collectives are dependent on movement and if a framework is to be developed that adequately represents collective phenomena and the relationship between the collective as a whole and its individual members, the classification system of collective phenomena must be supplemented with a comprehensive classification of the movement patterns of collective phenomena.
- What do we mean when we say a collectivity's behaviour is more than just the aggregated behaviour of its individual members? Perhaps a more suitable question would be 'to what extent does the collectivity exhibit behaviour which cannot meaningfully be attributed to its members?' *The football team won the match* — but no individual player does; *the herd stampeded* — but an individual animal cannot stampede; *the leaves from the tree carpeted the ground* — but no individual leaf does. In each of these examples, a behaviour or property can be meaningfully attributed to the collective but not to its individual members. In some of these cases the collective behaviour is simply the aggregation of the individuals' behaviour, e.g., a stampede is the sum of the actions of the individual animals, whereas in others it is more complex than this, e.g., the football team winning a match. How do we identify the important attributes (i.e., the attributes that result in a collectivity exhibiting coherent behaviour)?
- Should any final ontology of collectivities consider collectivities that do not exhibit coherent behaviour? At the beginning of this research, we only wished to consider the representation of collectivities that did exhibit coherent behaviour. However, there are a large subset of collectivities that do not exhibit coherent be-

haviour but are still important in certain domains, e.g., a waiting list of patients due to undergo surgery.

- It has been noted that some of the classification criteria are not clear-cut but exhibit some form of gradation. This applies in particular to the coherence and role criteria. More research is needed to determine the extent to which a more graded system is required within the classification.
- It is clear that collectivities are dependent on granularity since a phenomenon is only viewed as a collectivity at a high-level of granularity; at a lower-level all that can be seen are the individual members. Can the new ontology that we develop be built on two levels, one for the individuals and one for the collectivity as a whole, but still articulate the relationship between the two levels?
- There are collectivities whose members are also collectivities. For example universities and schools are made up of departments. It is possible that such hierarchical decomposition of collectives might introduce another important dimension into the classification. This links in also with our earlier observation of the granularity-dependence of certain forms of collectivity.
- It can be seen that location and coherence criteria are closely related. Are they more suited to different types of collectivity? For example, would location criteria be more suited to classify collectivities which consist of animals instead of using coherence criteria where one must assign intentions and purposes to the individual members?
- A typology of intentional collectives is given in [7]. Can this be used to expand the coherence classification criterion that we have used?
- We have begun putting example collective phenomena into our proposed classification system. More examples will need to be gathered to (i) validate our system, (ii) see if any patterns emerge.
- Once a suitable classification system has been decided upon, a formalisation will be produced. However, it would be premature to do this until we fully understand what we are trying to model.

5. Conclusion

Three existing ontologies have been analysed. From this analysis it has become clear that they are ill-suited for representing collectivities because they do not adequately provide the resources for representing information about a collectivity as a whole as well as its individual members and the relation between them. We believe that the ontological modelling of collective phenomena requires existing ontologies to be replaced or extended. To prepare the way for the formal work needed for this it is necessary to make clear the nature of what we are modelling. To this end, a typology of collectivities has been proposed. This typology will need to be validated by means of further examples, and possibly extended, before it is used to build an ontology of collectives.

References

[1] Ali. W and B Moulin. 2D–3D multiagent geosimulation with knowledge-based agents of customers' shopping behaviour in a shopping mall. In David M. Mark and Anthony G. Cohn, editors, *Spatial*

Information Theory: Proceedings of International Conference COSIT 2005, Lecture Notes in Computer Science, pages 445–458, Ellicottville, NY, USA, 2005. Springer.

[2] M. Batty, J Desyllas, and E. Duxbury. The discrete dynamics of small-scale carnival events: agent-based models of mobility in carnivals and street parades. *Int. J. of Geographical Information Science*, 17(7):672–697, 2003.

[3] A. Rector, J. Roger, and T. Bittner. Granularity scale and collectivity: When size does and doesn't matter. *Journal of Biomedical Informatics*, 39(3):333–349, 2006.

[4] S. Schulz, E. Beisswanger, J. Wermter, and U. Hahn. Towards an upper level ontology for molecular biology. In *Proceedings of the American Medical Informatics Association Symposium (AMIA'06)*, pages 694–698, 2006.

[5] Frederick F. Schmitt. *Socializing Metaphysics: The Nature of Social Reality*. Rowman & Littlefield Publishers, 2003.

[6] T. Bittner, M. Donnelly, and B. Smith. Individuals, universals, collections: On the foundational relations of ontology. In A.C. Varzi and L. Vieu, editors, *Proceedings of the Third Conference on Formal Ontology in Information Systems (FOIS-04)*, pages 37–48, 2004.

[7] E. Bottazzi, C. Catenacci, A. Gangemi, and J. Lehmann. From collective intentionality to intentional collectives: An ontological perspective. *Cognitive Systems Research*, 7:192–208, 2006.

[8] N. Guarino. Formal ontology and information systems. In N. Guarino, editor, *Formal Ontology in Information Systems: Proceedings of FOIS'98*, pages 3–15, Italy, 1998.

[9] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. WonderWeb deliverable D18. Technical report, Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy, 2003.

[10] A. D. Spear. Ontology for the twenty first century: An introduction with recommendations. http://www.ifomis.uni-saarland.de/bfo/manual/manual.pdf, 2006.

[11] P. Grenon and B. Smith. SNAP and SPAN: Towards dynamic spatial ontology. *Spatial Cognition and Computation*, 4(1):69–104, 2004.

[12] P. Simons. *Parts. A Study in Ontology*. Clarendon Press, Oxford, 1987.

[13] John Searle. Collective intentions and actions. In P. Cohen, J. Morgan, and M. Pollack, editors, *Intentions in Communication*, pages 401–415. The MIT Press, Cambridge, Massachusetts, 1990.

[14] Raimo Tuomela. *The Philosophy of Sociality: The Shared Point of View*. Oxford University Press, 2007.

[15] C. Castelfranchi. Modelling social action for AI agents. *Artificial Intelligence*, 103(1-2):157–182, 1998.

[16] Z. Wood and A. Galton. Collectives and how they move: A tale of two classifications. Submitted for Publication, 2008.

[17] S. Dodge, R. Weibel, and Lautenschütz. Taking a systematic look at movement: Developing a taxonomy of movement patterns. *The AGILE workshop on GeoVisualization of Dynamics, Movement and Change*, 2008.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-121

Artefacts and Roles: Modelling Strategies in a Multiplicative Ontology

Laure VIEU [a,b], Stefano BORGO [a] and Claudio MASOLO [a]
[a] *LOA-ISTC-CNR, Trento*
[b] *IRIT-CNRS, Toulouse*

Abstract. The purpose of this paper is to examine different modelling strategies available in a multiplicative formal ontology, and the principles that drive their choice. This study is based on the results of recent work aiming at extending the foundational ontology DOLCE to grasp two quite different notions, that of artefact and that of role. These results, summarized in the paper, show that two multiplicative modelling strategies, entity stacking and property reification, are essential in both cases.

Keywords. representation methodology, ontology of social reality, multiplicativism, kinds of entities, reification

Introduction

Some topics pose very general modelling puzzles in formal ontology. For instance, how should we handle artefacts? As properties or as separate categories of individuals? If *the paperweight on my desk* refers to the same individual as *this pebble*, then "being a paperweight" could simply be a property instantiated by the pebble individual. But if the paperweight has some properties that arguably the pebble doesn't have, like being meant to hold papers, then, *the paperweight* could denote a distinct individual. Similarly, are roles properties or individuals? Does *the Chancellor of Germany* refer to a specific person, Angela Merkel, enjoying a certain property, or to a more abstract entity, an 'institutional role', that can be 'played' by several people across time? And is there any difference between Angela Merkel and "Angela Merkel as Chancellor"?

The purpose of this paper is to show that the answers to such questions do not simply rely on personal taste, and that a careful examination of data is needed before deciding on the property vs. individual issue. The overall picture is even more complex for two reasons: there exist several theories of properties [23] and concepts [14] and several ways to formalize them. If simple predication, which semantically corresponds to class membership in classical first-order logic, is a standard means to express properties [3], some authors advocate the use of tropes or individual properties [5], while others use universals and a non-extensional instantiation relation [1].

In this paper we will present two recently proposed theories of artefacts and roles which make use of two modelling strategies involving the introduction in the domain of new entities in addition to less controversially existing ones. Such strategies are sometimes indicated as *entity multiplication* [6].

This work can be seen as a first attempt to clarify what, beyond predication, are the modelling strategies useful in formal ontology. This means recognizing when predication is not adequate and identifying the adequate alternative strategy. For each of these alternative strategies, if they are multiplicative, it is also essential to clarify the identity criteria of the new entities and analyse the different relations they enjoy with the other ones. Of course, such a discussion, which we consider important for the field, won't be exhausted by the present paper.

This paper is structured as follows. In the next section, we will briefly review some multiplicative approaches used in philosophy. We focus on two strategies, respectively called *entity stacking* and *property reification*. They both result in the extension of the domain with new individuals, but these are of different sorts and enjoy different relations with respect to the other individuals. Sections 2 and 3 will then present two different works on the modeling of, respectively, artefacts and roles, which make use of the two strategies. The theory of artefacts is a first attempt to encode in a formal system recent philosophical analyses of this notion. The theory of roles builds on a larger literature, in both computer science and philosophy, which has raised a number of subtle modeling issues and is thus a more elaborated one. Finally, in Section 4 we discuss why and how the two strategies have been used in these works. To clarify the differences between the two, we also try to contrast these uses when the choices do not appear obvious.

1. Brief review of multiplicative approaches in philosophy

Parsimony is a principle that no-one, philosopher or computer scientist, renounces to. But this principle is more often than not faced with puzzles of different kinds, and philosophers have come up with various answers requiring and motivating the multiplication of entities and/or entity sorts in various ways and degrees. As a result, one finds a wide range of philosophical stands, from the extreme unifier (reductionist) that will accept only one sort of entities —and the least quantity of them— e.g., bunches of molecules, to the extreme multiplier for whom each non-synonymous linguistic description denotes a different entity. In this paper, we place ourselves in a framework which is neither extremes, i.e., a moderate multiplicativist one. We put forward motivations for multiplication that we find compelling, although we are well aware that these arguments are motivated more from a representational point of view than a philosophical one.

Here, we briefly examine two different multiplicative approaches that correspond to two ways a first-order theory can be enriched by introducing new entities in the domain of quantification: *entity stacking* and *property reification*.

Entity Stacking. Entity stacking is grounded on the notion of identity, or rather, non-identity between entities. Two identity criteria for entities are widely used: entities are identical if and only if they have the same proper parts (mereological extensionality), or if and only if they display the same properties (Leibniz's law). Many philosophers interpret mereological extensionality (in the case of concrete entities) as a rendering of the motto "no two things at the same place at the same time", assuming parthood amounts to spatial inclusion [25,31]. The identification of *having the same proper parts* with *being co-located* and the adoption of mereological extensionality yields that spatial co-location implies identity.

This position raises famous puzzles when the two identity criteria clash: the *statue and the clay* and *the ship of Theseus* [25, Introduction]. In the first puzzle, the statue is co-located at all times with the clay but their modal properties are different: the clay but not the statue can be reshaped and the statue but not the clay can loose tiny parts. The second puzzle is more complex as it involves identity across time. Assuming that the ship of Theseus undergoes a successive change of all of its planks, and supposing that the old planks are kept apart and eventually re-assembled into a ship, which one is the ship of Theseus? Considering Leibniz's law, one would tend to chose the first, since preserving ownership and some form of spatio-temporal continuity, while considering mereological extensionality, one would chose the second.

Some authors are compelled by such puzzles to conclude that co-location doesn't imply identity, and that the relation between the statue and the clay or between the ship and the aggregate of planks is instead one of *constitution*. Constitution is a form of existential dependence between co-located entities, i.e. an asymmetric relation that gives rise to *levels* or *substrates* of different kinds of entities with specific identity criteria, e.g. matter, physical object, intentional agent, collective... Assuming this *entity stack*, the puzzles disappear because having the same proper parts is no longer equivalent to (but only implies) co-location (see [31] for a detailed discussion).

A similar but slightly different problem regards the identity criteria for players of *roles*. In particular, the *counting problem* [15] makes evident that in order to *count* the passengers of an airline in a year we cannot count the persons that flew that airline. The properties of passengers are different from those of persons, so here too entity-stacking can solve the issue (although alternative solutions exist). In Section 3, this strategy is applied introducing new individuals, called *qua-entities*: *John qua Alitalia passenger of flight 123 on day D* is co-located with and *inheres in* John. As we will see, there are differences between *constitution* and *inherence*.

Property reification. In FOL, properties are usually represented by *predicates*, which have *sets* as semantic counterpart. Predicates necessarily model extensional, static, and a-contextual properties and are closed under logical connectives. Many philosophical and/or cognitive theories of properties drop some of these assumptions: for example *universalists* (see [1] for a review) refuse extensionality and Boolean closure, while *conceptualists* (see [18] for a review) tend to think that concepts are properties created and possibly destroyed at specific times, dependent on human minds or societies, etc.

To relax the previous assumptions and to talk of temporal extensions and dependences of properties, i.e., to predicate over properties, staying in a FOL framework, a *reification* process that introduces properties as individuals in the domain of quantification and in the language is necessary. And just as done in the general models for reducing a fragment of second-order logic to first-order [30], in addition to reifying properties, new primitive relations of *instantiation* (one for each arity) are required to replace predication in the language.[1] Finally, in this approach, we are obliged to characterize the additional sort of individual, say, *Universal* or *Concept*.

The reification process has been adopted in philosophy of language or computer science for reifying more complex logical constructs, like propositions, facts or states

[1]In the general models, the domain of interpretation of properties is (a subset of) the standard one, i.e. properties denote sets or sets of tuples of (other) individuals. Dropping the assumption that properties are extensional, as Universalists do, requires a different interpretation, thus different models.

of affairs, as in the cases of *events* in Davidsonian semantics [8] or of *situations* in the *situation calculus* [26]. We will nevertheless consider in this paper only the reification of properties.

2. A Multiplicative Approach to Artefacts

Let's first examine the proposal for a theory of physical artefacts put forward in [4] as an extension of the foundational ontology DOLCE [20,3]. This proposal is based on the recognition of the creator's intentions as an essential property of artefacts and thus the distinction of physical artefacts into a separate category from the physical objects that constitute them.

2.1. Physical Artefacts as a Separate Category

The notion of *physical artefact* is a slippery one even when limiting ourselves to non-agentive artefacts: we all think we know what we mean when talking about these objects and we can provide good examples of them. When asked to name artefacts, most of our examples point to manufactured items. Indeed, the recognition of physical manipulation on an item gives us a strong indication that that entity is an artefact. Nonetheless, physical manipulation is not a key element for artefacts: tons of manipulated entities are definitely not counted as artefacts (e.g., sawdust, cut-off hair, mowed grass) while a moment of thought suffices to find physically unaltered objects that actually make up artefacts (e.g., the pebble used as paperweight of the introduction, the unworked shells used as money in the past).

The fundamental element to single out artefacts is intentionality [9,2]: we intentionally select objects in order to use them for a purpose perhaps physically modifing them to suit our tasks. Intentionality then is part and parcel of the process of attributing functionalities (capacities) to objects, i.e., of the process in which artefacts are created. The intentionality involved in this process is not a property *of* artefacts and even less so of the selected entities, it's a property of the agents who created them. Artefacts are the results of agents' intentionality so their existence depends on an action of entities external to them. These observations lead to consider artefacts as ontologically separated from other physical entities like water and trees, and therefore to entity-stacking: the paperweight is not the pebble, it is co-located with it and constituted by it. Indeed, it can be argued that the pebble does not depend on any creation event, nor on any agent, that it is not meant to hold papers, and that it is older than the paperweight.

This approach constitutes an alternative to the more obvious option considering artefactuality as a property that physical objects may or may not have or acquire. This would mean though, rejecting the widely recognized sortal nature of artefacts [10], as well as failing to acknowledge that being the object of intentions is an *essential* property of artefacts, and being unable to account for the just-mentioned dependence relations and the other commonsensical differences between the pebble and the paperweight. As we will see shortly, the multiplicative approach has another advantage which the predicative approach has difficulties to cope with. Artefacts can maintain their identity trough repair, i.e., through the change of the entity that constitute them.

The class of artefacts addressed in [4], which is also the most studied in the philosophical literature [2,17,10,29], collects entities constituted by entities in two subcategories of Endurant in the DOLCE taxonomy [20]: amounts of matter (olive oil, pieces of glass) and non-agentive physical objects (statues, boats, microchips).[2] The result is the extension of DOLCE with the category Physical Artefact, proposed as a new subcategory of Physical Endurant, along with the given Amount of Matter, Physical Object, and Feature.

2.2. *Intentional Selection and Attributed Capacities*

We have motivated the view that artefacts have an ontological status and are essentially the result of an intentional act of an agent (or group of agents) called creator. In [4] this intentional act, the creation, is considered as an act of *intentional selection* of the entity to constitute the artefact. This intentional selection is not enough, though. In the example of the paper-weight made of a pebble, the artefact is the result of some agent intentionally selecting the pebble and *attributing* to it some *capacities* (holding paper without ruining it, being easily grasped by hand, being firm etc). Of course, the artefact might turn out not to have the capacities the agent attributed to it, as it could be flawed or malfunctioning, but that does not affect the existence of the artefact itself.

The notion of capacity is taken from Cummins [7] and characterizes the dispositions [24] or behaviors a physical endurant is able to express, independently of any agent, even in the specific case of artefacts. Capacities are a type of DOLCE individual *qualities* possessed by elements in (at least) categories Amount of Matter, Non-agentive Physical Object, and Physical Artefact. Individual qualities in DOLCE are each mapped to a value in the quality space (e.g., the space of colors, the space of times, etc.) that characterizes how they are structured [3]. Although the notion of *capacity space* is quite complex and not yet well understood, we can say that the value corresponding to the capacity quality of an entity at a given time is a region of the capacity space collecting all the various dispositions the entity is able to express at that time. For instance, the capacity of this pen has now the value of writing finely in black when swept on paper, fitting in one's hand when grasped, making a certain noise when struck on the table…

The *attributed capacity* is a distinct individual quality of entities in Physical Artefact only that maps to the *same* space as the capacity quality, and characterizes the purpose or function of the artefact as determined by its creator. The pen above certainly has the attributed capacity to write finely in black when swept on paper and to fit in one's hand when grasped, but most probably not to make a certain noise when struck on the table. The fact that (actual) capacities and attributed capacities are elements of a same space has a number of advantages since it allows to define malfunctioning (see [4]) and reconcile the physical and mental nature of artefacts [17]. Note that, although capacities and attributed capacities map into the same space, the first are physical qualities whereas the latter are intentional. Also, they differ in their dependence on time: the attributed capacity is fixed by the creation event and does not change over time.

Although we do not discuss the formalization here (see [4] on this), we point out that some elements new to DOLCE are needed to characterize the category of artefacts. In addition to the qualities *capacity* and *attributed capacity*, we assume the ontological formalization of a primitive relation of *intentional selection* which characterizes the

[2]The notion of artefact, though, arguably covers individuals constituted by entities of yet other categories: agentive physical objects (robots), features (speed bumps, folds in a skirt), perdurants (judgements, performances, wars) and more abstract ones like pieces of music, laws or social institutions.

events that we called *creations*. $IntentionalSel(e, p, x, y, q)$ stands for "e is the event of the agent p intentionally selecting the amount of matter or non-agentive physical object y and attributing to it the attributed capacity q, obtaining the artefact x". Then the category Physical Artefact is characterized by an axiom positing that artefacts are the result of such events (among other axioms).

2.3. Entity stacking

A central element in this formalization of artefacts is the assumption that the artefact (the paper-weight) *is not* the endurant of which it is made (the pebble). As said above, the paper-weight starts existing when it is created, generally well after the pebble does; the two objects, although co-located when both present, have different properties, in particular different lifetimes, and are therefore different. In addition, the paper-weight depends on — here, is constituted by — the pebble but not vice versa. We thus adopt the entity-stacking strategy described in Section 1.

DOLCE already adopts such a multiplicative approach, in particular to distinguish the statue (as a physical object) from the clay, the amount of matter that constitutes it. However, for artefacts it is important to note that what constitutes the paper-weight is the physical object pebble, and not simply the amount of (rock) matter that in turn constitutes the pebble.[3] The pebble is not an amount of rock since it is shape-dependent: the amount of rock persists after crushing, but the pebble doesn't. Artefacts therefore add still another layer, an intentional one, to the DOLCE constitution hierarchy. As a result, in the traditional example of the statue which actually is an artefact, we need to distinguish three co-located entities, and not simply two as argued in DOLCE and more generally in the literature on material constitution (see Section 1): the intentionally created statue, the specifically shaped and structured physical object,[4] and the mereologically determined amount of matter.[5]

Since artefacts are distinct from physical objects and amounts of matter, they obey different identity criteria as suggested by the fact that artefacts can be repaired and undergo parts substitution without losing their identity. In the ship of Theseus example, substituting a plank doesn't destroy the artefact, although it does destroy the original physical object that constituted it, i.e., the plank assembly.[6] So, part substitution implies the disappearing of the original constituting entity and the coming into existence of a new one, with some degree of spatio-temporal continuity between the two. This explains that an artefact cannot "jump" from one material entity to a separate one at will, as your home does when you move, the two separate houses both preexisting to and surviving the move. This observation brings some light on the important distinction between artefacts (e.g., a house) and roles (e.g., a home) which we will address in Section 4. To further illustrate the approach on the ship of Theseus story, when the original planks are assembled again, depending on the identity criteria given to physical objects, one could

[3] The amount of matter also constitutes the artefact, as constitution is transitive in DOLCE.

[4] DOLCE doesn't provide generic identity criteria for physical objects. Here we assume that shape and internal structure are involved, although this requires further studies.

[5] There are also cases in which the artefact is directly selected out of an amount of matter as in the cases of a cup of water selected as a cake ingredient, or of the plastic produced in a factory.

[6] The amount of wood doesn't disappear since the old plank is not annihilated but kept apart. It is simply no longer a self-connected amount of matter.

argue that the original physical object comes back into existence. The artefact that is created then, though, is a different one, with perhaps another creator and other attributed capacities.

The identity criteria of artefacts are based on their intentional aspect, i.e., their attributed capacity, and their constituting entities. Two artefacts are the same if they have the same attributed capacity, were originally selected out of the same physical object or amount of matter, and are constituted of the same entities at all times. Identity criteria should also state under which conditions an entity persists or disappears all together. Ordinary malfunctioning does not make an artefact disappear, so the identity criteria cannot impose a simple match between attributed capacity and capacity. The artefact's disappearing is not simply due to its constituting entity's disappearing either, since the latter can be substituted as we have just seen. A combination of the two aspects, modulated by appropriate notions of granularity and vagueness, is required. The persistence of artefacts thus combines a *significant* degree of spatio-temporal continuity of the successive constituting entities, the existence of all specific essential parts if any (e.g., for a car, its frame), and the actuality of a *significant* part of the attributed capacity. The latter is modeled by a significant overlap between (the value of) the attributed capacity and (that of) the capacity. Note that since the attributed capacity is not restricted to the overall or main function of the artefact and covers structural specifications like size, shape, weight, composition etc., a malfunctioning artefact does possess most of its attributed capacity. Even an ill-designed artefact, e.g., a medieval flying machine, may possess most of its attributed capacity.

3. Roles, concepts and qua-entities

The second case examined in this paper is that of *relational roles* (henceforth called roles) of objects as analyzed in [22,21]. Typical examples of roles are socially relevant notions such as *student*, *president* or *customer*, but *catalyzer* is an example in another domain. Following the main literature on this topic (see [28]), roles are considered in this work as dynamic, anti-rigid, and relationally dependent properties.

The first two aspects (dynamism and anti-rigidity) regard, respectively, the temporal and modal nature of the relation between roles and their *players*. Entities could play a role only during a specific time interval (in a possible world or set of possible worlds). For instance, a person could be a student for only two years, and even in the case she is a student for her whole life, it is not necessary for her, i.e. persons are not necessarily students. In order to represent these aspects, standard modelling approaches consider a modal (possibly temporal) logic and assume roles as unary predicates, or introduce a parameter (possible world or time) and assume roles as binary predicates.

As far as the third aspect is concerned, intuitively, a property is relationally dependent when it depends — via a *pattern of relationships* [27] — on additional "external" properties.[7] [22] adopts a generalization of the notion of *definitional dependence* introduced by Kit Fine [12]: a property ϕ is definitionally dependent on a property ψ if, nec-

[7] A property ϕ (generically) depends on an external property ψ if, necessarily, for every instance x of ϕ there exists an instance y of ψ which is an *entity external* to x. The notion of "external entity" is not straightforward. Note though that notions like part, constituent, and quality typically identify entities that are "internal" to other entities.

essarily, any *definition* of ϕ ineliminably involves ψ. In particular, roles can be defined on the basis of a relation whose arguments are characterized by specific properties. This aspect is standardly represented by defining the predicates that correspond to a role. Let us consider, for example, the role of 'being a customer' defined as: "a customer is a person that (repeatedly) buys (something) from a company". In this case, the (unary) predicate 'being a customer' is defined on the basis of 'buy', 'being a person', and 'being a company".[8] On the basis of the same predicates we can define the role 'being a seller' as "a seller is a company from which a person (repeatedly) buys (something)". Formally we have:

(Dc) $Customer(x) \triangleq Person(x) \wedge \exists y(Buy(x, y) \wedge Company(y))$,
(Ds) $Seller(x) \triangleq Company(x) \wedge \exists y(Buy(y, x) \wedge Person(y))$.

3.1. Concept Reification

The novelty of the approach introduced in [22] consisted in taking seriously into consideration two additional aspects of roles, their *intensional* and *conventional* nature. The intensional nature relies on the fact that the previous definition (Dc) not only specifies the extension of the role 'customer' but *defines* what a customer is. While in classical logic two co-extensional predicates are necessarily indistinguishable, one would like to consider that two co-extensional roles are different if they are defined in different ways. The conventional nature implies an existential dependence on some society that produced the conventions, sometimes described as context-dependence. For instance, the role of president (of a country) depends on the existence of that country, but also on the existence of some sort of constitutive text defining what 'being a president' means in that country: constraints on who can be player, ways in which players are appointed, norms constraining what the player may or may not do, etc. In addition, this role can be dated: it has been created at some point, and so exists in time.

As discussed in Section 1, in a FOL framework, these latter aspects can be captured by reifying the roles, and the social conventions or contexts that define them, so that the definition and dependence relationships can be expressed. Making roles, and, more generally, socially defined concepts, part of the domain of discourse also enables making justice to their temporal dimension.

The approach followed in [22] is based on a clear distinction between (i) the properties in the *ground ontology*, represented as predicates and therefore assumed as static, rigid, extensional, and not explicitly defined or linked to a social context (e.g., the primitive predicates of the theory); and (ii) the properties (called "concepts") reified at the object level, that are not necessarily static, rigid, and extensional and for which it is possible to explicitly describe some aspects of the social contexts that define them (called "descriptions").

Concepts (*CN*) are *defined* (DF) by *descriptions* (*DS*) and they *classify* (CF) other individuals: $\mathsf{DF}(x, y)$ stands for "the *concept* x is defined by the *description* y"; $\mathsf{CF}(x, y, t)$ stands for "at the *time* t, the *individual* x is classified by the *concept* y", i.e., "at the *time* t, the *individual* x satisfies all the constraints stated in the description of the

[8] This definition obviously is very rough, and just for expository purposes. In particular, we do not consider the aspects linked to time and modality.

$$\begin{tikzcd}
\mathit{Person} & \mathit{CN} & \mathit{DS} & \mathit{CN} & \mathit{Company} \\
\mathtt{luc} \arrow[u, "inst"'] \arrow[r, "\mathsf{CF}_t"'] & \mathtt{customer} \arrow[u, "inst"'] \arrow[r, "\mathsf{DF}"'] & \mathtt{d} \arrow[u, "inst"'] & \mathtt{seller} \arrow[u, "inst"'] \arrow[l, "\mathsf{DF}"] & \mathtt{fiat} \arrow[u, "inst"] \arrow[l, "\mathsf{CF}_t"]
\end{tikzcd}$$

Figure 1. The customer/seller example with the reification of roles.

concept y".[9] Because we want to talk about the creation or destruction of concepts, we consider them as a special kind of endurants that are present (i.e., exist) in time. In addition, concepts must be defined (DF) by an unique description that cannot change during the life of the concept, i.e. new descriptions define new concepts and new concepts are defined by new descriptions.[10] Assuming that *Person* and *Company* are predicates in the ground ontology, the previous customer/seller example can be represented as in Figure 1 (where an arrow labelled with *inst* stands for standard predication, an arrow labelled with CF_t between `a` and `cn` stands for $\mathsf{CF}(\mathtt{a}, \mathtt{cn}, t)$, and an arrow labelled with DF between `cn` and `d` stands for $\mathsf{DF}(\mathtt{cn}, \mathtt{d})$).

3.2. Qua-entities

An important characteristic of roles consists in the possibility of introducing 'new' attributes or of hiding attributes of the players: students but not persons have a registration number, passengers but not persons have a flight number, customers but not persons have a code or a purchase number, persons but not customers have weights...

Let us consider, for example, the 'customer code' attribute. `luc` can be (simultaneously) the customer of different companies, and therefore he can have different codes, one for each company he is customer of. But if *code* is an attribute of `customer` then `luc` can have only one code value.[11] In the case of customer codes, it is possible to solve the problem modelling *code* by means of a function with two arguments — the customer and the company — but the problem can be more serious as in the case of two classical puzzles: the *counting problem* [15] and the *conflicting properties paradox* [11]. In these cases we could be forced to add parameters. For example "Luc as customer of Fiat spent 15K euros last year, while as customer of Sony just 2K euros". The 'having spent 15K euros' and 'having spent 2K euros' cannot apply to the person `luc` on pain of inconsistency. Therefore we need to introduce the customer code as additional parameter on 'having spent 15K euros'. For counting passengers, as the same person can fly several times the same company, we need to consider a temporal parameter.

In [28], an alternative multiplicative solution is assumed. This solution presupposes the existence of *adjunct entities*, instances of *Customer*, that existentially depend but are disjoint from instances of *Person*. Each instance of *Customer* has a customer code. Instead of the notion of *adjunct entity*, some philosophers (see [13] for example) have introduced that of *qua-entity*. [13] considers qua-entities (called qua-objects) to solve the

[9]This is for unary concepts, but the approach can be extended to concepts of any arity.

[10]This is a strong assumption that makes impossible to account directly for the intuition that some concepts evolve in time. In this approach, this intuition can be handled by a series of related concepts.

[11]Note that it is not possible to introduce one code attribute for each company, because the general theory of customer cannot be based on which companies exist.

statue-and-clay puzzle (see Section 1): "the statue may be identified with that matter under the description of having Goliath shape", or with that matter qua-Goliath shaped. Indeed the same idea can be applied to entities that are playing some roles. So, in addition to `luc` there are two new individuals: `luc` *qua* Fiat customer (`luc`$_{\text{qua}}$`fiat_cust`) and `luc` *qua* Sony customer (`luc`$_{\text{qua}}$`sony_cust`) that *inhere in* `luc` (see Figure 2). The inherence relation i is typically addressed in *trope* theory. It is an asymmetrical relation that specializes the *existential specific dependence* (eSD) by the non-*transferability* principle introduced in [19]: $i(x, y) \wedge i(x, z) \rightarrow y = z$. Note that both relations of inherence and constitution are asymmetric dependences, but while inherence holds at all times, so `luc`$_{\text{qua}}$`sony_cust` always depends on `luc`, the ship of Theseus can change the physical object and amount of matter constituting it during its life. In addition, while constitution necessarily implies spatial co-location, in the case of inherence this constraint is not made explicit, even though in the example of customer/seller co-location holds.

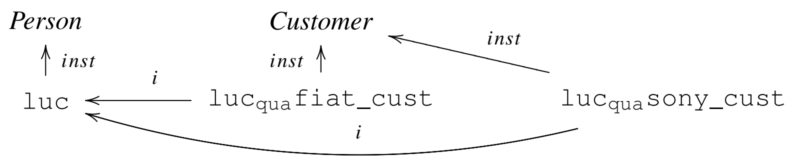


Figure 2. Customers are different from persons.

Putting together the reification of roles and the introduction of qua-entities, we obtain the solution illustrated in Figure 3. Note that qua-entities existentially depend not only on the entities they inhere in but also on the roles (`customer` in the example) and on the respective *selling* companies (`fiat` and `sony`), i.e. as expressed in (Dc) and (Ds), the qua-entities depend on the fact that a 'buy' relation holds between one specific person and one specific company.

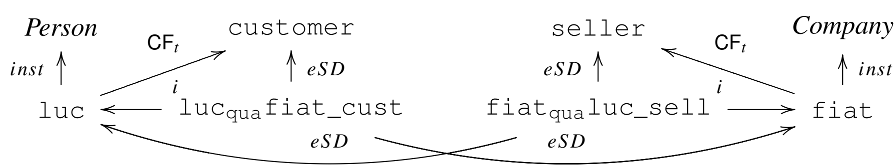


Figure 3. Putting together the reification of roles and the introduction of qua-entities.

Using qua-entities may appear excessively multiplicative, and one may wonder whether alternative solutions could be found. For instance, to solve the *counting problem* we could think to count *events* instead of qua-entities. In the passenger example, the problem indeed disappears counting "carrying events" because they have a one-to-one correspondence with qua-entities. And similarly with counting (singular) "buying events" for the role customer. In these cases, roles can be seen as specific ways of participating in events (like thematic roles). However, firstly note that the conflicting properties paradox as described above is not solved because the properties considered don't apply to events but to persons, and the same person can buy different things at different prices. The commitment to a four-dimensional account of persons and to their *temporal* *slices* does not help either because a person who participates (as customer) in simultaneous buying events has just one temporal slice during these events so the two inconsistent properties apply to this slice. Secondly, counting events is not enough when the same person participates in the same single event (or more generically in the same kind of events) with several roles. For instance, let us suppose that Berlusconi participated to some industrial meeting both as Italian Prime Minister and as the President of Mediaset. If we want to count the *representatives* present at the meeting we cannot count persons (just Berlusconi), we cannot count temporal slices (just Berlusconi during the meeting), and we cannot count events (just the meeting). That qua-entities do participate as such in events appears compelling in this example. It also shows that is not so obvious that one can do without qua-entities.

4. Discussion

Let's summarize the sorts of evidence that motivated the adoption of entity-stacking and property reification in the above studies, and the issues at stake.

Entity-stacking. Applying Leibniz's law, entities are to be distinguished when different properties apply to them. Of course, not all apparent differences in properties are to be taken at face value. In particular, one must be cautious with distinguishing *de re* assertions from *de dicto* ones before claiming that we are facing differences [31]. So in this task, one may want to focus on *essential properties* of the entities to be modelled, and rely on philosophical analyses to uncover them.

As already said, the main drawback of the entity stacking strategy is the expansion of the domain of quantification. Because philosophers tend to accept the Quinean principle "to be is to be the value of a variable", the consequence is a stronger *ontological commitment*, which is not accepted by all. However, in a modelling perspective, constraints on the expressive power of the adopted representation language and the analysis of (possibly different) ontological positions assumed in different existing models need to be taken into account. In such cases it is often necessary to enrich the domain with entities that are *useful* from a conceptual and practical point of view, even though one may claim their ontological ground is shaky. In addition, as noted by Heil [16], unification (reductionism) can be quite impractical because a too complex reduction can make some high-level patterns and relations, e.g. political decisions or social interactions, "invisible at the level of physics". On the other hand, if modelling is done for a particular application in mind, and with no reuse or interoperability perspectives, one should of course limit the range of properties under consideration to relevant ones. So, for instance, if function and purpose are not relevant in a given application dealing with artefacts (however surprising this might be) one is better off without entity stacking at all.

If entity stacking is adopted, in addition to the respective identity criteria, one must pay special attention to the nature of the existential dependence relations linking the entities of different layers. We have seen above that physical artefacts require constitution and qua-entities inherence. Indeed, a physical artefact needs a physical object (or amount of matter) in order to exist, but at different times the same artefact can be constituted by different entities, as it might be repaired. On the other hand, qua-entities inhere in the same *host* during their whole life (the host can undergo changes, e.g. Luc can become fatter, but he remains Luc), i.e. the dependence is specific. Simple existential dependence is not enough for entity-stacking: in both cases, at a given time, the artefact is (directly) constituted by and the qua-entity inheres in a unique entity.[12] Further studies are needed to examine the range of dependence relations possibly involved in entity-stacking. For example, if extending the work on artefacts to non-physical ones, constitution, which entails spatial co-location, is no longer appropriate.[13]

Let's note that our use of entity-stacking is peculiar: the motto "no two objects of the same kind at the same place at the same time" adopted by philosophers accepting co-location, such as Wiggins [32], doesn't apply to artefacts and qua-entities considered as kinds. Indeed, an entity usually simultaneously plays several roles and therefore gives rise to several qua-entities inhering in it at the same time. For artefacts, this is less obvious, but still happens when the same physical object is repeatedly selected for different purposes. The pebble which was selected for making a paperweight might at some point be selected again for making a pestle or a hammer, without making the paperweight disappear.

Property reification. The motivation for property reification appears to be less ontological, i.e., of a more practical nature. The need to predicate over properties forces reification in a first-order framework but not in a second-order one. And applied ontology does not use second-order languages for obvious tractability reasons. However, all philosophers studying properties propose analyses characterizing the properties of properties. So, following the Quinean principle, they all ontologically commit to the existence of (first-order) properties, although probably for them there is no different ontological commitment in introducing a property in the basic domain of quantification (as an individual) or at the meta-level (as a predicate in a second-order language).

One should be aware of the major drawback of property reification. If specifying the logical structure of complex properties on the basis of simple ones is required, one has to face the technical problem of introducing in the theory the whole 'logical' language necessary to do so. Not only we have to characterize the instantiation relation between reified properties and other individuals in the domain, but we might need new relations that stand for logical connectives and quantifiers. This can both be very expensive and lead to serious formal troubles.

We have seen that property reification is required to account for the social (intensional and conventional) aspects of roles. This is actually needed for artefacts too. Above we considered only the notion of individual artefact, but *artefact types* are no less important. Bell invented the telephone, but didn't create the telephone that sits on your desk. Most engineers actually create designs (artefact types), although they also often create some prototypes (individual artefacts) in order to test these designs. In [4] a proposal is made to characterize artefact types, i.e., those concepts — the reified properties introduced in [22] and described above — that classify (individual) artefacts.

We have argued above that individual artefacts require entity-stacking, i.e., cannot be dealt with simple predicates describing a property of their constituting entities. For

[12] Qua-entities depend also on other entities like the role and the other instantiations of the arguments of the relation on the basis of which the role is defined, but inhere only in the player of the role.

[13] Entity-stacking is needed in the abstract domain too. Theories have to be distinguished from their semantic contents: Turing machines and recursive functions are proved to be equivalent, but still are different theoretical objects.

the same reasons, a simple reified property cannot do. It might be less obvious that roles and qua-entities are not adequate (see, e.g., Fine's proposal for considering the statue as a qua-entity in Section 3.2). But artefacts cannot be the qua-entities generated by a role of physical objects: as explained above, constitution and inherence behave differently, so we would be unable to account for artefact repairing. There are of course reified properties *of artefacts*, for instance the artefact types just evoked. Other such properties are roles; for instance, *home* is a role of houses, and *product*, in the sense of item in the selling list of a merchant, is a role that most often has artefacts as players.

In this paper we described, illustrated, and compared two multiplicative modelling strategies, entity-stacking and property reification. We believe such a study is useful in applied ontology, when computer scientists are faced with practical modelling choices. But this study is by no means complete. In particular, we have not examined a third, important, multiplicative strategy, the one calling for individual properties or tropes. While waiting for a full methodological 'manual' we hope will be made available in a near future, the reader may refer to [3] for a detailed discussion of the motivations for the use of *qualities* as individual properties in DOLCE.

References

[1] D. M. Armstrong. *Universals: An Opinionated Introduction*. Westview Press, 1989.

[2] L. R. Baker. The ontology of artifacts. *Philosophical explorations*, 7(2):99–111, 2004.

[3] S. Borgo and C. Masolo. Foundational choices in DOLCE. In R. Poli, Healy M., and Kameas A., editors, *Theory and Applications of Ontology*, volume 2, chapter 12. Springer Verlag (to appear), 2008.

[4] S. Borgo and L. Vieu. Artifacts in formal ontology. In A. Meijers, editor, *Handbook of the Philosophy of the Technological Sciences*, volume 2: Artifact ontology and artifact epistemology. Elsevier (to appear), 2008.

[5] K. Campbell. *Abstract Particulars*. Basil Blackwell, Oxford, 1990.

[6] R. Casati and A. Varzi, editors. *Events*, chapter "Introduction". Dartmouth, Aldershot, 1996.

[7] R. Cummins. Functional analysis. *Journal of Philosophy*, 72:741–765, 1975.

[8] D. Davidson. The logical form of action sentences. In N. Rescher, editor, *The Logic of Decision and Action*, pages 81–95. University of Pittsburgh Press, Pittsburgh, 1967.

[9] R. Dipert. *Artifacts, Art Works, and Agency*. Temple University Press, Philadelpia, 1993.

[10] C. Elder. On the place of artifacts in ontology. In E. Margolis and S. Laurence, editors, *Creations of the Mind: Theories of Artifacts and Their Representation*, pages 33–51. Oxford University Press, Oxford, 2007.

[11] E. Fales. The ontology of social roles. *Philosophy of Social Sciences*, 7(2):139–161, 1977.

[12] K. Fine. Ontological dependence. *Proceedings of the Aristotelian Society*, 95:269–90, 1995.

[13] K. Fine. Acts, events and things. In *Sixth International Wittgenstein Symposium, Kirchberg-Wechsel (Austria)*, pages 97–105, 1982.

[14] R. L. Goldstone and A. Kersten. Concepts and categorization. In A. F. Healy and R. W. Proctor, editors, *Comprehensive handbook of psychology*, volume 4, pages 599–621. Wiley, 2003.

[15] A. Gupta. *The Logic of Common Nouns: an investigation in quantified modal logic*. Phd thesis, Yale University, 1980.

[16] J. Heil. *From an Ontological Point of View*. Oxford Univeristy Press, 2005.

[17] P. Kroes and A. Meijers. The dual nature of technical artefacts. *Studies in History and Philosophy of Science*, 37(1):1–4, 2006.

[18] E. Margolis and S. Laurence, editors. *Concepts: Core Readings*. MIT Press, 1999.

[19] C. B. Martin. Substance substantiated. *Australasian Journal of Philosophy*, 58, 1980.

[20] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. Ontology Library (Wonder-Web Deliverable D18). Available at http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf, 2003.

[21] C. Masolo, G. Guizzardi, L. Vieu, E. Bottazzi, and R. Ferrario. Relational roles and qua-individuals. *AAAI Fall Symposium on Roles, an interdisciplinary perspective, November 3-6, 2005, Hyatt Crystal City, Arlington, Virginia*, 2005.

[22] C. Masolo, L. Vieu, E. Bottazzi, C. Catenacci, R. Ferrario, A. Gangemi, and N. Guarino. Social roles and their descriptions. In *Ninth International Conference on the Principles of Knowledge Representation and Reasoning*, Whistler Canada, 2004.

[23] D. H. Mellor and A. Oliver, editors. *Properties*. Oxford University Press, 1997.

[24] S. Mumford. *Dispositions*. Oxford University Press, Oxford, 1998.

[25] M. Rea, editor. *Material Constitution. A Reader*. Rowman and Littlefield Publishers, Lanham, MD, 1997.

[26] R. Reiter. *Knowledge In Action*. MIT Press, Cambridge, Massachussetts, 2001.

[27] J. F. Sowa. Using a lexicon of canonical graphs in a semantic interpreter. In M. W. Evens, editor, *Relational models of the lexicon*. Cambridege Univerity Press, 1988.

[28] F. Steimann. On the representation of roles in object-oriented and conceptual modelling. *Data and Knowledge Engineering*, 35:83–106, 2000.

[29] A. Thomasson. Artifacts and human concepts. In E. Margolis and S. Laurence, editors, *Creations of the Mind: Theories of Artifacts and Their Representation*, pages 52–73. Oxford University Press, 2007.

[30] J. van Benthem and K. Doets. Higher-order logic. In D. Gabbay and F. Guenthner, editors, *Handbook of philosophical logic*, volume I, Elements of Classical Logic, pages 275–329. Reidel, 1983.

[31] A. Varzi. The extensionality of parthood and composition. *The Philosophical Quarterly*, 58:108–133, 2008.

[32] D. Wiggins. On being at the same place at the same time. *Philosophical Review*, 77:90–95, 1968.

4. Reasoning and Integration

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-137

Towards Principles for Ontology Integration

Özgür L. ÖZÇEP [1]
Department for Informatics (AB WSV), MIN Faculty, University of Hamburg
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany
oezcep@informatik.uni-hamburg.de

Abstract. Resolving conflicts based on ambiguities in the public vocabulary is one of the challenges in semantic integration. Though different suggestions for resolving (ambiguity) conflicts with semantic integration operators exist, there is still a need for clear formalizations of adequacy criteria for the operators. In this article, adequacy criteria for semantic integration similar to rationality postulates of classical belief revision but adjusted to the semantic integration scenario are formalized. The criteria are intended to capture integration settings in which the integration candidates are well developed ontologies with a shared public vocabulary. In such cases, both ontologies have to be preserved in some form in the integration result and have to be recoverable from the integration result. Additionally, the integration result has to be consistent and provide connections between the integrated ontologies. The criteria are applied by evaluating a small collection of integration operators that solve conflicts deriving from ambiguities in the public vocabulary.

Keywords. Ontology integration, belief revision, semantic mapping, reinterpretation

1. Introduction

An ontology for some domain is an important means for knowledge sharing among communication partners. It provides formal descriptions of relevant concepts, relations and individuals of the domain. Additionally, in most cases an ontology is represented in some formalism for which tractable reasoning mechanisms exist [1]. Though an ontology is intended to enable communication, in practice there may be many possibly heterogeneous ontologies. Heterogeneities between ontologies can lead to conflicts (mismatches) prohibiting the seamless interoperability between the communication partners. Semantic integration is concerned with the problem of making information from different knowledge sources interoperable by integrating them—taking into account the possible heterogeneity of the knowledge sources.

There are different types of mismatches. One of the mismatches on the ontology level are ambiguities. Ambiguous terms occur frequently in natural languages but can be found in formal representations of ontologies, too. E.g., in one bibliographical ontology *Article* may denote all documents that are published in a journal. In another bibliograph-

[1] I would like to thank Carola Eschenbach for her valuable feedback on previous drafts of this article and I would like to thank the reviewers for their helpful comments.

ical ontology *Article* may denote the wider concept of all documents that were published in a journal, proceedings or in a collection ([2], p. 4). In this article, the focus is on conflicts that can be explained by ambiguities in the public vocabulary. Here, a symbol in the public vocabulary is termed ambiguous if it belongs to different terminologies in the sense that it has different specifications in different ontologies.

In the semantic-integration literature, different strategies for yielding interoperability of heterogeneous ontologies are suggested. Though in most cases it is reasonable to claim that the strategies yield adequate integration results, there is still a need to formally explicate the underlying principles or adequacy criteria. Formal principles for semantic integration provide a means for specifying the properties of a given strategy and form the basis for comparing different integration strategies. Depending on the integration setting, different adequacy criteria may result. In the intended setting of this article, two ontologies are integrated. The ontologies are meant to be used in the same domain, hence it is assumed that the ontologies are kindred. Furthermore, the ontologies are well-tried and well developed and especially have no internal conflicts. Both ontologies have a shared public vocabulary which can lead to ambiguity conflicts between them. Due to the similarity condition an adequate integration result for this setting has to provide connections between the ontologies. As the initial ontologies are free from conflicts also the outcome should be free from conflicts. Moreover, as the ontologies are well developed, an adequate integration result should preserve both ontologies in some form.

I formalize the adequacy criteria in the line of postulates which have proved of value in classical belief revision [3]. The postulates axiomatize classes of binary integration operators suitable for the intended integration setting by explicating the properties for the integration operators. As the whole set of postulates formalizing the adequacy criteria is inconsistent, I define (two) classes of integration operators based on uniform reinterpretation that fulfill (two different) subsets of the postulates and thereby show that the subsets of the whole set of postulates are consistent. Additionally, I use the postulates to analyze a selection of integration operators proposed in the literature and compare them with the uniform-reinterpretation operators.

2. The Role of Semantic Mappings in Semantic Integration

Research on (ontology-based) semantic integration is centered on the formal representation of semantic mappings, the reasoning with semantic mappings, and the discovery of semantic mappings [1].

One representation format for semantic mappings is given by *bridging axioms* [2]. In [2] ontologies are represented in different name spaces so that the vocabularies of the ontologies are disjoint. Bridging axioms are defined as sentences in some superset of the ontologies' vocabularies that relate corresponding terms of the ontologies. The ontologies and bridging axioms together constitute a theory which allow to apply a uniform reasoning procedure. In operationalizing the principles for adequate integration I use bridging axioms to map corresponding concepts and roles of different ontologies.

Another representation format for semantic mappings, called *views*, is used in the ontology integration systems framework [4]. Views are functions that map concepts and roles of a global ontology to corresponding concepts and roles of local ontologies. In the formulation of the preservation criterion for semantic integration, I use substitutions that relate concepts and roles of the original ontology and its preserved version in the integration result. The substitutions are comparable with views.

A third kind of semantic mapping implicitly contained in the framework of Qi et al. [5], which is outlined in Section 5, are weakening functions. These functions map axioms to other (weaker) axioms rather than symbols to other symbols.

The last representation format for semantic mappings to be mentioned here is embedded in the general framework of distributed systems (DS) [6] or the more specific framework of distributed description logics [7]. A distributed system consists of local ontologies and ontology alignments, which are defined as the whole set of semantic mappings between two ontologies. Different from bridging axioms, the semantic mappings between the local ontologies are not axioms in a global ontology but additional components relating corresponding concepts and roles in the local ontologies. Therefore, a DS needs a reasoning procedure different from the reasoning procedures of the local ontologies. In [6] and [7] the semantic basis for the reasoning procedures is established by defining special distributed semantics for distributed systems which also explicate the semantic role of the semantic mappings. In the case of distributed semantics, it does not matter whether the local ontologies share a vocabulary or not. There is no distinction between a public vocabulary for communication between the ontologies and a private vocabulary. Hence, ambiguities in a public vocabulary cannot occur.

There are different methods to establish semantic mappings. One method is based on shared upper ontologies like DOLCE [8]. Upper ontologies are meant to provide a common understanding of general terms which can be used by lower level domain-specific ontologies extending the upper ontologies. If ontologies are built on a common upper ontology, the number of ambiguity conflicts may be reduced in comparison with the case where no common upper ontology exists. But still ambiguities can occur between terms not explicated in the upper ontology. Therefore, additional mechanisms are necessary for resolving ambiguity conflicts.

Another method for establishing mappings uses heuristics, [9], [10], [11] or machine learning techniques [12]. The heuristics are guided by, e.g., natural-language descriptions, concept descriptions or structural or logical properties of the ontologies. Adequate semantic integration demands an integration result which displays clear connections to the original ontologies and relates (parts of) the integrated ontologies. Hence, whatever kind of heuristics is used, it has to facilitate the construction of semantic mappings that realize these connections and relations.

3. Adequacy Criteria for Ontology Integration Operators

A formal representation of adequacy criteria for integration strategies in some integration setting has the advantage of specifying exactly the properties an integration strategy has or should have. Additionally, different strategies for the same integration setting can be compared with regard to the formal adequacy criteria. Postulates as used in the area of belief revision have proved useful for specifying properties of revision strategies. Grounded in the pioneering work of Alchourrón, Gärdenfors and Makinson [3] (AGM), belief revision was designed to capture rational change of beliefs by proposing postulates that have to be fulfilled by contraction resp. revision functions. The basic AGM postulates in the adapted form described in [5] are the starting point for the following postulates for semantic integration but have to be adapted and extended for the intended integration setting.

In the integration setting for which the following postulates are intended, an agent holding an ontology O_1 wants to integrate a kindred ontology O_2 received from a different agent into his ontology O_1. Both ontologies are defined over a common public vocabulary $\mathcal{V}$ of non-logical symbols, formally expressed with $\mathcal{V}(O_1), \mathcal{V}(O_2) \subseteq \mathcal{V}$. As there may be symbols in $\mathcal{V}$ that have different specifications in O_1 and O_2 ambiguity conflicts in the public vocabulary can occur and have to be resolved. That an ontology is free from conflicts can formally be described by the concept of consistency. The set $Mod(O)$ denotes the set of *models* of O, i.e., the set of interpretations that make all sentences of O true. An ontology is *consistent* (free from conflicts) iff it has a model. The outcome of the integration is denoted by $O_1 \circ O_2$ where $\circ$ is a binary integration operator. As both ontologies, O_1 and O_2, are assumed to be well developed, there is a strong need to preserve both ontologies in $O_1 \circ O_2$.

The preservation of the ontologies can be formalized with the help of *substitutions*. Substitutions are functions mapping non-logical symbols to terms of the same type. As I intend to use substitutions for the preservation of ontologies in integration settings with ambiguity conflicts, I define the subclass of Ambiguity Resolution Substitutions. Let $\mathcal{V}_p$ be a (public) vocabulary and let $\mathcal{V}'$ be a disjoint (private) vocabulary, $\mathcal{V}_p \cap \mathcal{V}' = \emptyset$. The set of *Ambiguity Resolution Substitutions* $ARS(\mathcal{V}_p, \mathcal{V}')$ (or just ARS) is the set of injective substitutions that map a non-logical symbol in $\mathcal{V}_p$ either to itself or to a new non-logical symbol (of the same type) in $\mathcal{V}'$. E.g., $\sigma \in ARS(\mathcal{V}_p, \mathcal{V}')$ could map a concept symbol $K_1 \in \mathcal{V}_p$ to itself $K_1 = \sigma(K_1)$ or to another concept symbol $K_1' = \sigma(K_1) \in \mathcal{V}'$. The set of symbols $s \in \mathcal{V}_p$ for which $\sigma(s) \neq s$ is called the *support* of σ. Substitutions are extended to sentences and sets of sentences in the usual way. The ontology $\sigma(O)$ or alternatively $O\sigma$ is called a *substitution variant* of O.

The following postulates are intended to describe operators that are guaranteed to resolve terminology-dependent inconsistencies between two ontologies without loosing (parts of) the ontologies. An operator $\circ$ *fulfills* a postulate if and only if the postulate is true for all input ontologies O_1 and O_2.

As ontologies provide a conceptualization of a domain, the mere syntactic difference in the representation of an ontology should not lead to semantic changes of the integration strategies referring to the ontology. This criterion is expressed by the *extensionality postulates* (O1.1) and (O1.2). Additionally, the criterion is underlying the other postulates as they refer to the models of the ontologies and not to their syntactic structure.

(O1.1) If $\mathcal{V}(O_1) = \mathcal{V}(O_1')$ and $Mod(O_1) = Mod(O_1')$,
then $Mod(O_1 \circ O_2) = Mod(O_1' \circ O_2)$

(O1.2) If $\mathcal{V}(O_2) = \mathcal{V}(O_2')$ and $Mod(O_2) = Mod(O_2')$,
then $Mod(O_1 \circ O_2) = Mod(O_1 \circ O_2')$

If two ontologies are compatible, there is no reason to assume that their terminologies are different. In this case (like for AGM-revision-operators) an adequate way to integrate the ontologies is to form their union.

(O2) If $O_1 \cup O_2$ is consistent, then $Mod(O_1 \circ O_2) = Mod(O_1 \cup O_2)$

Terminology-dependent conflicts between the ontologies occur because they share a common public vocabulary. A resolution of the conflicts should preserve as much of the terminologies and of the shared vocabulary as possible. Otherwise the integrated ontology $O_1 \circ O_2$ would not support communication (integration) based on the interface vocabulary. The *monotony postulate* (O3.1) says that all sentences derivable in O_1 are derivable in the resulting ontology and thus expresses a strict form of preservation of O_1. The *success postulate* (O3.2) expresses a strict form of preservation of O_2 in the sense that all sentences derivable in O_2 are still derivable after the integration.

(O3.1) $Mod(O_1 \circ O_2) \subseteq Mod(O_1)$
(O3.2) $Mod(O_1 \circ O_2) \subseteq Mod(O_2)$

Since the integration result $O_1 \circ O_2$ should be consistent if possible (see Postulate (O6) below) the criteria formalized by (O3.1) and (O3.2) cannot be fulfilled by an integration operator at the same time unless $O_1 \cup O_2$ is consistent. Hence (O3.1) and (O3.2) are not principles for all semantic integration operators but suggest the definition of two alternative types of operators (see Section 4). In the case of conflicts, the integration process has to abandon parts of O_1 or O_2.

As the ontologies are well developed and well-tried, it is desirable to preserve both ontologies in some form. One way to preserve an ontology is to transfer it to a substitution variant in a different name space. The *preservation postulates* (O4.1) and (O4.2) demand the existence of substitutions such that the substitution variants of O_1 resp. O_2 are contained in the resulting ontology.

(O4.1) There is a substitution σ_1 with: $Mod(O_1 \circ O_2) \subseteq Mod(O_1\sigma_1)$
(O4.2) There is a substitution σ_2 with: $Mod(O_1 \circ O_2) \subseteq Mod(O_2\sigma_2)$

Postulate (O4.1) can be considered as a generalization of the monotony postulate (O3.1) (letting σ_1 be the identity function) and (O4.2) as a generalization of the success postulate (O3.1). Although there are no operators that fulfill (O3.1) and (O3.2) in the presence of (O6), it is possible to define operators that fulfill (O3.1) and (O4.2) or (O3.2) and (O4.1) in the presence of (O6). The feature of preserving both ontologies in the integration result is the essential point at which integration operators differ from classical belief-revision and update operators. Belief-revision and update operators fulfill success, i.e., integrate O_2 as a whole, but only at cost of loosing parts of the ontology O_1. The preservation postulates demonstrate that the goal of semantic integration is different from the goals of belief revision and belief update. Belief revision aims at solving conflicts due to false information. Belief update aims at solving conflicts due to outdated information. Semantic integration (based on reinterpretation) aims at solving conflicts due to ambiguous use of terms.

The following postulates represent additional generalizations of (O3.1) and (O3.2). Postulates (O5.1) and (O5.2) require the existence of substitutions such that the old ontologies O_1 and O_2 can be recovered by applying the substitution to the integration result. These postulates are termed *substitution recovery postulates*.

(O5.1) There is a substitution σ_1 with: $Mod((O_1 \circ O_2)\sigma_1) \subseteq Mod(O_1)$
(O5.2) There is a substitution σ_2 with: $Mod((O_1 \circ O_2)\sigma_2) \subseteq Mod(O_2)$

If an operator fulfills both preservation postulates, (O4.1) and (O4.2), then for any ontologies O_1 and O_2 that can be integrated there are substitutions σ_1, σ_2 such that $Mod(O_1 \circ O_2) \subseteq Mod(O_1\sigma_1) \cap Mod(O_2\sigma_2) = Mod(O_1\sigma_1 \cup O_2\sigma_2)$. Consequently, the existence of σ_1, σ_2 such that $O_1\sigma_1 \cup O_2\sigma_2$ is consistent is a necessary condition for the consistency of the integration result. This condition expresses the fact that there are no vocabulary-independent conflicts between the ontologies. Two ontologies O_1, O_2 are called *reinterpretation compatible* iff substitutions σ_1, σ_2 exist such that $O_1\sigma_1 \cup O_2\sigma_2$ is consistent. The *weakened consistency postulate* (O6) makes reinterpretation compatibility of O_1 and O_2 a sufficient criterion for the consistency of the result.

(O6) If O_1 and O_2 are reinterpretation compatible, then $O_1 \circ O_2$ is consistent

If two ontologies are reinterpretation compatible, then each ontology is consistent. As in the central cases of semantic integration the integrated ontologies are consistent, there is no need for postulates that specify the operator for inconsistent O_1 or inconsistent O_2.

Belief-revision and belief-update operators fulfill stronger versions of (O6). Belief-revision operators guarantee consistency of the result if O_2 is consistent. Belief-update operators guarantee consistency if O_1 and O_2 are consistent [13].

Postulates (O1.1), (O2), (O3.2) correspond to the AGM postulates named extensionality, vacuity and success [3]. Postulate (O6) is a weakening of the consistency postulate of AGM. Postulates (O4.1), (O4.2) and (O5.2), (O5.2) are additional postulates capturing the ideas of preservation and substitution recovery.

4. Uniform Reinterpretation Operators

The aim of this section is to operationalize the postulates discussed in Section 3. I will show that there exist (two) subsets of the postulates that are fulfilled by (two) different reinterpretation operators, respectively.

The postulates of Section 3 do not presuppose an exact specification of the representation format for ontologies. For this and the following section I assume that an ontology is represented as a finite set of sentences in a description logical (DL) language.[2] But it is equally possible to represent the ontologies with predicate logics as the definitions of the operators below do not depend on the use of DLs. In a DL language, the set of non-logical symbols, denoted by $\mathcal{V}$ or indexed variants, consists of constants, denoted by a, b, c and indexed variants, concept symbols, denoted by K and indexed variants, and role symbols, denoted by R and indexed variants. $\mathcal{V}(O)$ is the set of all non-logical symbols occurring in O. $\mathcal{V}_c$ is the set of all constants, $\mathcal{V}_{CR}$ the set of all concept and role symbols occurring in $\mathcal{V}$. Concept descriptions are built using concept constructors and are denoted by the meta-variables C, D and indexed variants. The set of concept constructors used in this article contains concept negation $\neg$, concept conjunction $\sqcap$, and concept disjunction $\sqcup$.

The sentences of an ontology can be classified as *TBox axioms*, which express terminological knowledge, and the *ABox axioms*, which express world knowledge. TBox axioms are of the form $C \sqsubseteq D$ (All C are D), so called GCIs (General Concept Inclusions), or of the form $R_1 \sqsubseteq R_2$ for role symbols R_1, R_2, so called role inclusion axioms. The ABox axioms are of the forms $C(a)$ (a is a C) or $R(a, b)$ (a is R-related to b).

The core idea for the definition of reinterpretation operators is developed in [15]. The operators of [15] get as first argument an ontology and as second argument a *literal*, i.e., a sentence of the form $K(a)$ or $\neg K(a)$ for a concept symbol K.

[2]For details regarding the definitions and the syntax of DLs see [14].

In this article, I extend the operators of [15] to ontology-integration operators allowing ontologies as second arguments. Therefore, not only one symbol but a set S of symbols has to be considered for disambiguation. Additionally, not only concept symbols but also role symbols or constants are considered as candidates for disambiguation.

In order to develop the definition of the uniform-reinterpretation operators, let O_1 and O_2 be two ontologies with a common vocabulary $\mathcal{V} \supseteq \mathcal{V}(O_1 \cup O_2)$. Let $\mathcal{V}_{12} = \mathcal{V}(O_1) \cap \mathcal{V}(O_2)$ denote the shared set of non-logical symbols. Further, let $\mathcal{V}'$ be a new vocabulary of private symbols, $\mathcal{V}' \cap \mathcal{V} = \emptyset$. Further, let $S \subseteq \mathcal{V}_{12}$ be a subset of the set of common non-logical symbols and let $\rho_S \in ARS(\mathcal{V}, \mathcal{V}')$ be a substitution with support S.

If $O_1 \cup O_2$ is inconsistent and O_1 and O_2 are reinterpretation compatible, the inconsistency can be explained by ambiguous symbols $S \subseteq \mathcal{V}_{12}$ in the shared vocabulary. The inconsistency can be resolved by decoupling the ontologies O_1 and O_2 with respect to S, either yielding $O_1 \cup O_2\rho_S$ or $O_2 \cup O_1\rho_S$. In the first decoupling, the terminology of O_1 is preserved. For O_2, the first decoupling results in a shift in the meaning of the common symbols in S, they are reinterpreted. In the second decoupling, the terminology of O_2 is preserved and a shift in the meaning of the common symbols in S for O_1 results. As the substitution ρ_S is applied to all occurrences of symbols in O_1 resp. O_2 that stem from S the reinterpretation operators are termed *uniform*. A successful decoupling should select a symbol set $S \subseteq \mathcal{V}_{12}$ such that the decoupled ontologies $O_1 \cup O_2\rho_S$ resp. $O_2 \cup O_1\rho_S$ are consistent. The set $MRS(O_1, O_2)$ of Minimal Reinterpretation Symbols

$$MRS(O_1, O_2) = \{S \subseteq \mathcal{V}_{12} \mid Mod(O_1 \cup O_2\rho_S) \neq \emptyset \text{ and for all } S_1 \subset S: \\ Mod(O_1 \cup O_2\rho_{S_1}) = \emptyset\} \tag{1}$$

describes all inclusion-minimal symbol sets that lead to a consistent union of decoupled ontologies. A direct consequence of the definition is that $MRS(O_1, O_2) = MRS(O_2, O_1)$ and that $MRS(O_1, O_2) = \{\emptyset\}$ iff $O_1 \cup O_2$ is consistent. The set $MRS(O_1, O_2)$ is empty iff O_1 and O_2 are not reinterpretation compatible. In this case the reinterpretation operators cannot resolve the inconsistency between O_1 and O_2. (Compare Postulate (O6)). Choosing inclusion-minimal symbol sets realizes the idea of reinterpreting only those symbols that are involved in a conflict. As there are no formal criteria which of the sets in $MRS(O_1, O_2)$ have to be chosen, the operator definition will be parameterized by a selection function γ. A *selection function* γ is a function that maps a set to a subset, such that $\gamma(\emptyset) = \emptyset$ and for all sets $M \neq \emptyset$: $\emptyset \neq \gamma(M) \subseteq M$. So, if $MRS(O_1, O_2)$ is not empty, a selection function γ picks a non-empty subset of all inclusion-minimal symbol sets S that result in a consistent decoupling of O_1 and O_2. The union of these sets results in $S^* = \bigcup \gamma(MRS(O_1, O_2))$ which is the set of all disambiguated symbols in the reinterpretation process.

As O_1 and O_2 are assumed to be kindred ontologies, the reinterpretation operators relate the different readings for the disambiguated symbols in S by special terminological axioms which function as bridging axioms. The axioms relate a concept resp. role symbol s to a new private concept resp. role symbol $\sigma(s)$. For concept and role symbols $s \in S$ the set $\{s \sqsubseteq \sigma(s) \mid s \in S_{CR}\}$ contains a lower bound for $\sigma(s)$ and $\{\sigma(s) \sqsubseteq s \mid s \in S_{CR}\}$ contains an upper bound for $\sigma(s)$. The set $A(S, \sigma)$ of all possible bounds additionally contains identities for constants in S.

$$A(S, \sigma) = \{s \sqsubseteq \sigma(s), \sigma(s) \sqsubseteq s \mid s \in S_{CR}\} \cup \{s = \sigma(s) \mid s \in S_c\} \tag{2}$$

Given an ontology O, a set of symbols S and a substitution σ with support S, the set $MB(S,\sigma,O)$ (set of Maximal sets of Bounds) contains all inclusion-maximal sets of bounds that can be consistently added to O. To describe the set $MB(S,\sigma,O)$ formally, I use a construction similar to the remainder-sets construction in partial-meet revision [16]. For sets of sentences A, B let $A \oplus B$ denote the set of inclusion maximal subsets of A that are compatible with B.

$$A \oplus B = \{X \subseteq A \mid Mod(X \cup B) \neq \emptyset \text{ and for all } Y \subseteq A : \text{If } X \subset Y, \\ \text{then } Mod(Y \cup B) = \emptyset\} \tag{3}$$

So $MB(S,\sigma,O)$ can be defined by

$$MB(S,\sigma,O) = A(S,\sigma) \oplus O \tag{4}$$

A second selection function γ_2 is used to select a subset of the maximal sets of bounds. Similar to the approach in partial meet revision, the set of bounds to be added is the intersection of inclusion maximal sets selected by γ_2.

Using the notation above, type-1 and type-2 operators can be defined.

Definition 1 *Let γ_1, γ_2 be selection functions and let $\overline{\gamma} = (\gamma_1, \gamma_2)$. Let $\mathcal{V}, \mathcal{V}'$ be disjoint vocabularies, and let O_1, O_2 be ontologies with $\mathcal{V}(O_1 \cup O_2) \subseteq \mathcal{V}$. Let $S^* = \bigcup \gamma_1(MRS(O_1, O_2))$ and let $\rho_{S^*} \in ARS(\mathcal{V}, \mathcal{V}')$ be a substitution with support S^*. Then the* weak uniform-reinterpretation operators of type-1 *respectively* of type-2 based on $\overline{\gamma}$ and ρ_{S^*} *are defined by:*

$$O_1 \otimes_1^{\overline{\gamma},\rho_{S^*}} O_2 = O_1 \cup O_2\rho_{S^*} \cup \bigcap \gamma_2\Big(MB(S^*, \rho_{S^*}, O_1 \cup O_2\rho_{S^*})\Big) \tag{5}$$

$$O_1 \otimes_2^{\overline{\gamma},\rho_{S^*}} O_2 = O_2 \cup O_1\rho_{S^*} \cup \bigcap \gamma_2\Big(MB(S^*, \rho_{S^*}, O_2 \cup O_1\rho_{S^*})\Big) \tag{6}$$

A direct consequence of the definition is the interdefinability of $\otimes_1^{\overline{\gamma},\rho_{S^*}}$ and $\otimes_2^{\overline{\gamma},\rho_{S^*}}$, i.e., $O_1 \otimes_1^{\overline{\gamma},\rho_{S^*}} O_2 = O_2 \otimes_2^{\overline{\gamma},\rho_{S^*}} O_1$. The main observation concerns the fulfillment of the postulates.

Observation 1 *Let O_1, O_2 be ontologies, let γ_1, γ_2 be selection functions, let $S^* = \bigcup \gamma_1(MRS(O_1, O_2))$ and let ρ_{S^*} be a substitution with support S^*. Then*

1. $\otimes_1^{\overline{\gamma},\rho_{S^*}}$ *fulfills (O1.1), (O1.2), (O2), (O3.1), (O4.1), (O4.2), (O5.1), (O5.2) and (O6) but does not fulfill (O3.2) (success).*
2. $\otimes_2^{\overline{\gamma},\rho_{S^*}}$ *fulfills (O1.1), (O1.2), (O2), (O3.2), (O4.1), (O4.2), (O5.1), (O5.2) and (O6) but does not fulfill (O3.1) (monotony).*

The effect of the operators can be illustrated with a small example on bibliographic ontologies. Let two ontologies be given by

$$O_1 = \{Article \sqsubseteq \forall publ.Journ, Journ \sqsubseteq \neg Proceed, \\ publ(med01, procFOIS08), Proceed(procFOIS08)\} \tag{7}$$

$$O_2 = \{Article(med01)\} \tag{8}$$

Assume that the holder of O_1 integrates O_2 with a weak type-2 operator, i.e., decides to preserve O_2 in its original form. According to O_1, all articles are published in journals, and journals are different from proceedings. The ABox-part says that $med01$ is published in the proceedings of FOIS08. The TBox of O_2 is empty. The ABox-part says that $med01$ is an article. These two ontologies are not compatible. The set of minimal reinterpretation symbols $MRS(O_2, O_1)$ consists of the sets $\{med01\}$ and $\{Article\}$, i.e., reinterpreting the constant $med01$ or the concept symbol $Article$ of O_1 is sufficient to get decoupled consistent ontologies.

Deciding to reinterpret $med01$—thereby introducing a new symbol $med01' = \rho_{\{med01\}}(med01)$ for the constant $med01$ of O_1—means that $med01$ is thought to be ambiguous. This choice can be formalized by a selection function γ_1 with $\gamma_1^1(MRS(O_2, O_1)) = \{\{med01\}\}$. The choice fits to situations where $med01$ is used in O_1 to denote a publication in the proceedings of FOIS08 and $med01$ is used in O_2 to denote the follow-up article published in a journal. The weak operators do not relate constants, so for all bound selection functions γ_2^1 the integration result is

$$O_1 \otimes_2^{(\gamma_1^1,\gamma_2^1),\rho_{\{med01\}}} O_2 = O_{1[med01/med01']} \cup O_2 \tag{9}$$

The choice to reinterpret $Article$ fits to situations in which both ontologies speak about the same publication in the proceedings of FOIS2008 but in which $Article$ in O_2 is used in a broader sense than $Article$ according to O_1. This can be formalized by a selection function γ_1^2 with $\gamma_1^2(MRS(O_2, O_1)) = \{\{Article\}\}$. The unique maximal consistent set of bridging axioms is $\{Article' \sqsubseteq Article\}$, so all selection functions γ_2^2 fulfill $\gamma_2^2(\{\{Article' \sqsubseteq Article\}\}) = \{Article' \sqsubseteq Article\}$. The integration result is

$$O_1 \otimes_2^{(\gamma_1^2,\gamma_2^2),\rho_{\{Article\}}} O_2 = O_{1[Article/Article']} \cup \{Article' \sqsubseteq Article\} \cup O_2 \tag{10}$$

By broadening the set of potential bounds $A(S,\sigma)$ it is possible to define stronger versions of the operators. For the following comparison with [5] it is sufficient to adapt the strong operators of [15], which are defined only for literals as second arguments. In adapting the definitions of [15], I use *nominals*, i.e., concept descriptions $\{a\}$ whose extension consists exactly of the individual denoted by a. As only the type-2 operators will be compared with the operators of [5], the definitions of the type-1 operators are skipped. A *concept literal* is either a concept symbol K or a negated concept symbol $\neg K$. The meta-variable $\hat{K}$ is used for concept literals.

Definition 2 *Let O be an ontology over the vocabulary $\mathcal{V} \cup \mathcal{V}'$ with $\mathcal{V} \cap \mathcal{V}' = \emptyset$, $K \in \mathcal{V}$ be a concept symbol and $\hat{K}$ be a concept literal with $\mathcal{V}(\hat{K}) = \{K\}$. Let σ be a substitution with support $\{K\}$. The* strong uniform-reinterpretation operators of type 2 $\odot_2^C$ *that reinterpret the concept symbol are defined for literals by*

$$O \odot_2^C \hat{K}(a) = \begin{cases} O \cup \{\hat{K}(a)\} & \text{if } O \cup \{\hat{K}(a)\} \text{ is consistent,} \\ O\sigma \cup \{\hat{K}(a), \sigma(\hat{K}) \sqsubseteq \hat{K}, \hat{K} \sqsubseteq \sigma(\hat{K}) \sqcup \{a\}\} & \text{else} \end{cases} \tag{11}$$

The additional axiom $\hat{K} \sqsubseteq \sigma(\hat{K}) \sqcup \{a\}$ contributes to the strength of the operator. It says that a denotes the only individual that is $\hat{K}$ but not $\sigma(\hat{K})$. For an analysis of weak and strong operators for triggering literals and iterated applications confer [15].

5. Other Approaches to Integration

In the following, the postulates given in Section 3 are used to compare the belief-revision-oriented frameworks of Delgrande and Schaub [17] and Qi, Liu and Bell [5] and the semantic-integration-oriented framework of Goeb, Reiss, Schiemann and Schreiber [18] with the uniform-reinterpretation approach.

5.1. Private and Public Vocabularies in Belief Revision

The idea of using different vocabularies (private vs. public) in the integration process is not new to the belief-revision literature. Delgrande and Schaub [17] use this idea to define two belief-revision operators $\dot{+}, \dot{+}_c$. The belief-revision operators $\dot{+}$ (skeptical revision) and $\dot{+}_c$ (choice-revision) take as input two propositional knowledge bases over a public vocabulary $\mathcal{V}$ of proposition variables. Inconsistencies between the knowledge bases O_1, O_2 are resolved by completely decoupling O_1 and O_2. All proposition variables p in O_1 are substituted by new symbols $p' \in \mathcal{V}'$ where $\mathcal{V}'$ is a private vocabulary with $\mathcal{V}' \cap \mathcal{V} = \emptyset$. The decoupling yields the knowledge base $O_1\rho_{\mathcal{V}} \cup O_2$. The decoupled knowledge bases are related by adding maximal sets of biimplications $p \leftrightarrow p'$ between the old symbols p and the new symbols p' that are consistent with the $O_1\rho_{\mathcal{V}} \cup O_2$. In the case of $\dot{+}$ the intersection of all maximal sets of biimplications are added to the decoupled knowledge bases, in the case of $\dot{+}_c$ one maximal set selected by a selection function c is added to the decoupled knowledge bases. The result of the addition is closed with respect to a classical inference operator and intersected with the set of sentences containing only proposition variables from the public vocabulary.

The operators $\dot{+}, \dot{+}_c$ and the uniform-reinterpretation operators have many properties in common. For example, both classes of operators introduce a new private vocabulary in order to resolve conflicts. Moreover, the biimplications used in the definitions of $\dot{+}, \dot{+}_c$ can be considered as bridging axioms. But there are some essential differences. Delgrande and Schaub define $\dot{+}, \dot{+}_c$ for propositional knowledge bases which are not suitable for representing ontologies. Furthermore, [17] only considers biimplications $p \leftrightarrow p'$ and not implications $p \rightarrow p'$ which would directly correspond to subsumption relations $C \sqsubseteq C'$ between concept symbols used in the definitions of the reinterpretation operators. Lastly, the revision outcomes with respect to $\dot{+}$ or $\dot{+}_c$ are knowledge bases over the public vocabulary $\mathcal{V}$. The new symbols are introduced only as auxiliary variables for the revision procedure and do not occur in the revision result. Therefore the revision operators $\dot{+}, \dot{+}_c$ do not fulfill the preservation or substitution-recovery postulates for the first argument. But, $\dot{+}, \dot{+}_c$ fulfill extensionality in both arguments, vacuity, success and weakened consistency.

Observation 2 *The revision operators $\dot{+}, \dot{+}_c$ defined in [17] fulfill (O1.1), (O1.2), (O2), (O3.2), (O4.2), (O5.2) and (O6) but they do not fulfill (O4.1), (O5.1).*

5.2. Non-uniform Reinterpretation

Goeb et al. [18] describe an algorithm for the integration of a sender's ontology O_2 into a receiver's ontology O_1. Their framework tackles ontology integration in a very similar way as the uniform-reinterpretation operators. The main distinction relies in the nonuniformity of the operators described in [18]. For the following, let TB_i denote the TBox, AB_i the ABox of O_i, $i \in \{1,2\}$. The outcome of the integration is denoted by $O^{\circledast}$.

The algorithm has two main steps. In the first step, the ontologies are completely decoupled with respect to the common vocabulary $\mathcal{V}(O_1) \cap \mathcal{V}(O_2)$. This is similar to the decoupling in the case of the operators $\dot{+}, \dot{+}_C$ in [17]. For every concept and role symbol s (but not constants) two new symbols are introduced, a symbol $\sigma_1(s)$ for the receiver's symbol s, and a symbol $\sigma_2(s)$ for the sender's symbol s. The symbols are related in so called *triangles*, i.e., subsumption relations of the form $\sigma_1(s) \sqsubseteq s$ and $\sigma_2(s) \sqsubseteq s$ for $\sigma_1, \sigma_2 \in ARS$. After the reinterpretation, s denotes a super-concept of the receiver's and sender's s-concepts. The super-concept s neither belongs to the terminology of the receiver nor does it belong to the terminology of the sender. As there may be symbols with respect to which no decoupling is necessary for yielding consistency some symbols are re-translated into their original form. The decoupling is reduced to a inclusion-minimal set $S \in MRS(O_1, O_2)$ of symbols that cannot be consistently re-translated.

In the second step of the algorithm, additional consistent re-translations are applied. But this time the re-translations do not have to be uniform, i.e., different occurrences of the same symbol may be treated differently with respect to the decision to re-translate or not. The outcome of the re-translation can formally be described by applying substitutions to different parts of the ontologies. Let

$$\Sigma_1^S = \{\sigma : S \longrightarrow \sigma_1(S) \cup S \mid \text{For all } s \in S : \sigma(s) = s \text{ or } \sigma(s) = \sigma_1(s)\} \tag{12}$$

denote the set of substitutions that possibly substitute less symbols with new ones than σ_1. Similarly Σ_2^S is defined. Using this notation, the following representation of the ontology $O^{\circledast}$ results. There are

- symbol sets $S_1, S_2 \subseteq S$;
- substitutions $\tau_{AB_1} \in \Sigma_1^{S_1}$ and $\tau_{AB_2} \in \Sigma_2^{S_2}$, such that $\tau_{AB_1}(s) \notin S$ for all $s \in S_1$ and $\tau_{AB_2}(s) \notin S$ for all $s \in S_2$;
- partitions of the TBoxes $TB_1 = \biguplus_{1 \le i \le k} TB_{1i}$ and $TB_2 = \biguplus_{1 \le i \le l} TB_{2i}$;
- substitutions $\tau_{11}, \dots, \tau_{1k} \in \Sigma_1^{S_1}$ and $\tau_{2i}, \dots, \tau_{2l} \in \Sigma_2^{S_2}$

such that

$$\begin{aligned} O^{\circledast} = {} & \bigcup_{1 \le i \le k} T_{1i}\tau_{1i} \cup \bigcup_{1 \le i \le l} T_{2i}\tau_{2i} \cup (AB_1)\tau_{AB_1} \cup (AB_2)\tau_{AB_2} \cup \\ & \{\tau_{AB_1}(s) \sqsubseteq s \mid s \in S_1\} \cup \{\tau_{AB_2}(s) \sqsubseteq s \mid s \in S_2\} \end{aligned} \tag{13}$$

The representation of $O^{\circledast}$ in Eq. (13) says that for all $\alpha \in O_1$ a substitution variant (for a substitution in $\Sigma_1^{S_1}$) occurs in $O^{\circledast}$. Accordingly, for all $\alpha \in O_2$ there exists a substitution variant in $O^{\circledast}$. The substitutions τ_{AB_1}, τ_{AB_2} are uniform semantic mappings for the receiver's resp. the sender's ABox. The sets $\{\tau_{AB_1}(s) \sqsubseteq s \mid s \in S_1\}$ and $\{\tau_{AB_2}(s) \sqsubseteq s \mid s \in S_2\}$ can be considered as bridging axioms. The fact that $(AB_1)\tau_{AB_1} \subseteq O^{\circledast}$ means that a substitution variant of the receiver's ABox as a whole is contained in $O^{\circledast}$. Accordingly, $(AB_2)\tau_{AB_2} \subseteq O^{\circledast}$ means that a substitution variant of the sender's ABox as a whole is contained in $O^{\circledast}$. But note that in general it cannot be guaranteed that substitution variants of the receiver's or sender's ontology are contained in $O^{\circledast}$. Consequently, the operators of [18] do not fulfill the preservation postulates (O4.1), (O4.2) or recovery postulates (O5.1), (O5.2). Hence, also (O3.1), (O3.2) are not fulfilled.

Due to the non-uniform reinterpretations realized by the different substitutions τ_{1i} and τ_{2j} neither the left (O1.1) nor the right (O1.2) extensionality postulates are fulfilled. Thus, only postulates (O2) and (O6) are fulfilled by the operators of [18].

Observation 3 *The integration operator described in [18] fulfills (O2) and (O6) but it does not fulfill (O1.1), (O1.2), (O3.2), (O3.1), (O4.2), (O4.1), (O5.2) and (O5.1).*

5.3. Ontology Revision Based on Weakening Axioms

The framework of Qi, Liu and Bell [5] provides binary belief-revision operators for the revision of an ontology with another ontology.[3] The ontologies are represented by multisets of description logical axioms. The operators do not introduce new symbols. Rather, axioms α of the first ontology are mapped (weakened) to axioms $(\alpha)_w$ such that $Mod(\alpha) \subseteq Mod((\alpha)_w)$. The weakenings can be considered as semantic mappings that map axioms of one ontology to other axioms. Some axioms are mapped onto themselves, other axioms are mapped to weaker axioms. As Qi, Liu and Bell consider two different types of weakening, also two binary operators are defined, a revision operator $\circ_w$ and a refined revision operator $\circ_{rw}$. For a comparison with the uniform-reinterpretation operators, I focus on $\circ_w$.[4]

The weakening $(\cdot)_w$ on which $\circ_w$ relies is based on the idea of exception lists. GCIs α of the form $C \sqsubseteq D$ are weakened to GCIs $(\alpha)_w$ of the form $C \sqcap \neg\{a_1\} \sqcap \ldots, \sqcap\neg\{a_n\} \sqsubseteq D$, which means that all Cs, except for the individuals $a_1, \ldots, a_n$, are Ds. An ABox-axiom is mapped to itself or radically weakened to the tautology $\top$, which amounts to deleting it. A degree function $d(\cdot)$ counts the number of exceptions. The degree of $C \sqcap \neg\{a_1\}, \ldots, \sqcap\neg\{a_n\} \sqsubseteq D$ is n, the degree of an ABox-axiom α is 0 if it is mapped to itself and 1 if it is mapped to the tautology $\top$. O' is a weakened ontology of O_1 with respect to O_2, formally $O' \in Weak^w_{O_2}(O_1)$, iff $O' \cup O_2$ is consistent and there exists a bijection f from O_1 to O' such that for all $\alpha \in O_1$ the axiom $f(\alpha)$ is a weakening of α. The degree $d(O')$ of O' is the sum of the degrees of its axioms. The operator $\circ_w$ is defined by

$$\begin{aligned} O_1 \circ_w O_2 = \{O_2 \cup O_i \mid O_i \in Weak^w_{O_2}(O_1) \text{ and there exists no} \\ O_j \in Weak^w_{O_2}(O_1) \text{ such that } d(O_j) < d(O_i)\} \end{aligned} \tag{14}$$

The revision result contains unions of O_2 with d-minimal weakenings of O_1. The set is interpreted as the disjunction of the ontologies it contains.

The main common idea of the approach presented in [5] and the approach of uniform reinterpretation is that of weakening. E.g., let $O_1 = \{K_1 \sqsubseteq K_2, K_1(a)\}$ and $O_2 = \{\neg K_2(a)\}$, then $\{K_1 \sqcap \neg\{a\} \sqsubseteq K_2, K_1(a), \neg K_2(a)\} \in O_1 \circ_w O_2$. The axiom $K_1 \sqsubseteq K_2$ is weakened to $K_1 \sqcap \neg\{a\} \sqsubseteq K_2$.

The reinterpretation operator $\odot^C_2$ applied to the same ontologies results in $O_1 \odot^C_2 \neg K_2(a) = \{K_1(a), K_1 \sqsubseteq K_2', \neg K_2' \sqsubseteq \neg K_2, \neg K_2 \sqsubseteq \neg K_2' \sqcup \{a\}\}$. (Here $\rho_{\{K_2\}}(K_2)$ is abbreviated by K_2'). As in the case of $\circ_w$ the weakened axiom $K_1 \sqsubseteq K_2 \sqcup \{a\}$ follows from $O_1 \odot^C_2 \neg K_2(a)$. The perspective on weakening in [5] is a little bit different from the perspective on weakening for the uniform-reinterpretation operators. Because $\circ_w$ weakens axioms

[3] Qi et al. do not use the term *ontology* but only description logical knowledge bases.

[4] The weakening underlying the operator $\circ_{rw}$ allows non-trivial weakenings of axioms of the form $(\forall R.C)(a)$

while the reinterpretation operators weaken atomic concepts. For the example above one sees that weakening $K_1 \sqsubseteq K_2$ to $K_1 \sqcap \neg\{a\} \sqsubseteq K_2$ realizes an implicit weakening of the concept K_2 to $K_2 \sqcup \{a\}$ as $K_1 \sqcap \neg\{a\} \sqsubseteq K_2$ is logically equivalent to $K_1 \sqsubseteq K_2 \sqcup \neg\{a\}$. But it is not always possible to interpret the weakening of axioms as the weakening of some atomic concept. The implicit weakening of a complex concept description D in $C \sqcap \neg\{a\} \sqsubseteq D$ only concerns D as a whole, and not the weakening of an atomic concept.

As the consistency resolution in $\circ_w$ is guided by degree minimality and is not terminology-oriented like the uniform-reinterpretation operators, axiom-oriented consistency resolution is possible. Hence, different occurrences of the same concept symbol can be handled differently in resolving the inconsistencies. For the example, note that $\{\top, K_1 \sqsubseteq K_2, \neg K_2(a)\} \in O_1 \circ_w O_2$, i.e., here, the axiom $K_1(a) \in O_1$ is weakened to $\top$ while the axiom $K_1 \sqsubseteq K_2 \in O_1$ which also contains K_1 is not weakened. Consequently, the operator $\circ_w$ does not fulfill the postulates (O4.1) and (O5.1) which demand the preservation and recovery of O_1. The axiom-oriented resolution of conflicts is also responsible for non-extensionality in the left argument of $\circ_w$. The following observation lists the postulates (not) fulfilled by $\circ_w$.

Observation 4 *The operator $\circ_w$ of weakening-based revision fulfills (O1.2), (O2), (O3.2), (O4.1), (O5.1) and (O6) but does not fulfill (O1.1), (O4.1) and (O5.1).*

As Qi, Liu and Bell show, the weak revision operators fulfill a stronger version of (O6). They can guarantee consistency of the integration result if O_2 is consistent. In particular, if O_1 is inconsistent, the operator $\circ_w$ resolves the inconsistencies independently of O_2.

6. Conclusion

Formal adequacy criteria for the integration of ontologies allow a precise specification of the properties of integration strategies and provide a basis for a comparison of different strategies for the same integration setting. The criteria are formalized in the article by postulates in the same line as discussed in the area of belief revision [3]. Though the classical AGM postulates are the core of the integration postulates, the set of additional preservation and substitution-recovery postulates (O4.1), (O4.2), (O5.1) and (O5.2), which accommodate for the setting's presumption that both ontologies are well-tried cannot be fulfilled by classical belief-revision and belief-update functions.

The uniform-reinterpretation operators of type 1 and type 2 show that there are two maximal subsets of the postulates that can be fulfilled by two different classes of operators each of which guarantees a consistent integration result and the preservation and recovery of both ontologies as a whole. Both operators resolve possible conflicts between the ontologies by mapping one of the ontologies in a different name space. The type-1 operators preserve the first ontology in its original form, the second ontology is translated to a substitution variant by using an ambiguity resolution substitution. In the case of type-2 operators the second ontology is preserved and the first ontology is translated to a substitution variant. The substitutions used in the operators function as semantic mappings between the public terminology used for communication and the private terminology resulting from the integration process. Semantic mappings relating the concepts and roles of the different ontologies are represented by inclusion axioms which have the role of bridging axioms.

The revision operators of [17] and the weakening-based belief-revision operators defined in [5] can guarantee a consistent integration result (O6) in which O_2 is preserved in its original form (O3.2). But only parts of O_1 are preserved in the result, i.e., (O4.1), (O5.1) are not fulfilled. The operators defined in [18] can guarantee a consistent integration result (O6). But, as the ambiguous terms are not reinterpreted uniformly neither of the ontologies is preserved as a whole in the integration result, i.e., (O3.1), (O3.2), (O4.1), (O4.2), (O5.1) and (O5.2) are not fulfilled.

References

[1] N. F. Noy, "Semantic integration: A survey of ontology-based approaches.," *SIGMOD Record*, vol. 33, no. 4, pp. 65–70, 2004.

[2] D. Dou, D. V. McDermott, and P. Qi, "Ontology translation by ontology merging and automated reasoning," in *Proceedings of EKAW2002 Workshop on Ontologies for Multi-Agent Systems*, pp. 3–18, 2002.

[3] C. Alchourrón, P. Gärdenfors, and D. Makinson, "On the logic of theory change: partial meet contraction and revision functions.," *Journal of Symbolic Logic*, vol. 50, pp. 510–530, 1985.

[4] D. Calvanese, G. D. Giacomo, and M. Lenzerini, "Ontology of integration and integration of ontologies.," in *Description Logics*, vol. 49 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2001.

[5] G. Qi, W. Liu, and D. Bell, "A revision-based approach to handling inconsistency in description logics," in *Proceedings of the 11th International workshop on Non-Monotonic Reasoning (NMR06)*, 2006.

[6] A. Zimmermann and J. Euzenat, "Three Semantics for Distributed Systems and their Relations with Alignment Composition," in *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, Proceedings* (I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, eds.), pp. 16–29, Springer, 2006.

[7] A. Borgida and L. Serafini, "Distributed description logics: Assimilating information from peer sources," *Journal on Data Semantics*, vol. 1, pp. 153–184, 2003.

[8] A. Gangemi, N. Guarino, C. Masolo, and A. Oltramari, "Sweetening WORDNET with DOLCE," *AI Magazine*, vol. 24, no. 3, pp. 13–24, 2003.

[9] N. Noy and M. Musen, "The prompt suite: Interactive tools for ontology merging and mapping," *International Journal of Human-Computer Studies*, vol. 59, no. 6, pp. 983–1024, 2003.

[10] J. Euzenat and P. Valtchev, "Similarity-based ontology alignment in OWL-Lite," in *ECAI* (R. L. de Mántaras and L. Saitta, eds.), pp. 333–337, IOS Press, 2004.

[11] P. Bouquet, L. Serafini, and S. Zanobini, "Semantic coordination: A new approach and an application," in *The Semantic Web - ISWC 2003, Second International Semantic Web Conference, Proceedings* (D. Fensel, K. P. Sycara, and J. Mylopoulos, eds.), pp. 130–145, Springer, 2003.

[12] A. Doan, J. Madhavan, R. Dhamankar, P. Domingos, and A. Halevy, "Learning to match ontologies on the semantic web," *The VLDB Journal*, vol. 12, no. 4, pp. 303–319, 2003.

[13] H. Katsuno and A. Mendelzon, "On the difference between updating a knowledge base and revising it," in *KR'91: Principles of Knowledge Representation and Reasoning* (J. F. Allen, R. Fikes, and E. Sandewall, eds.), pp. 387–394, San Mateo, California: Morgan Kaufmann, 1991.

[14] F. Baader, "Description logic terminology," in *The Description Logic Handbook* (F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, eds.), pp. 495–505, Cambridge UP, 2002.

[15] Ö. L. Özçep and C. Eschenbach, "On the conservativity and stability of ontology revision operators based on reinterpretation," in Beierle and Kern-Isberner [19], pp. 84–99.

[16] C. Alchourrón and Makinson, "Hierarchies of regulations and their logic," in *New Studies in Deontic Logic*, pp. 125–148, D. Reidel Publishing, 1981.

[17] J. P. Delgrande and T. Schaub, "A consistency-based approach for belief change," *Artificial Intelligence*, vol. 151, no. 1-2, pp. 1–41, 2003.

[18] M. Goeb, P. Reiss, B. Schiemann, and U. Schreiber, "Dynamic T-box-handling in agent-agent-communication," in Beierle and G.Kern-Isberner [19], pp. 100–117.

[19] C. Beierle and G. Kern-Isberner, eds., *Dynamics of Knowledge and Belief. Proceedings of the Workshop at the 30th Annual German Conference on Artificial Intelligence, KI-2007*, Fernuniversität Hagen, 2007.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-151

Complexity of Reasoning With Expressive Ontology Mappings

Chiara GHIDINI [a], Luciano SERAFINI [a] and Sergio TESSARIS [b]

[a] *FBK-irst. Via Sommarive 18 Povo, 38050,Trento, Italy*
[b] *Free University of Bozen - Bolzano. Piazza Domenicani 3. 39100 Bolzano, Italy*

Abstract. State of the art formalisms for distributed ontology integration provide ways to express semantic relations between homogeneous components of different ontologies; namely, they allow to map concepts into concepts, individuals into individuals, and properties into properties. However, the extensive usage of multiple distributed ontologies requires the capability for expressing different forms of mappings, which extend the semantic relations among homogeneous components studied so far. In recent papers extensions of the Distributed Description Logic (DDL) have been proposed to represent mappings between heterogeneous elements; i.e. mappings connecting concepts and relations. In this paper we investigate the computational properties of reasoning with mappings between homogeneous as well as heterogeneous elements in distributed ontologies, and an effective decision procedure for reasoning with multiple ontologies bridged with both forms of mappings.

1. Introduction

The extensive usage of multiple distributed ontologies requires the capability for expressing different forms of mappings, which extend the semantic relations among homogeneous components studied so far. Consider the three top-level ontologies DOLCE-Lite (Descriptive Ontology for Linguistic and Cognitive Engineering)[1], GUM (Generalised Upper Model)[2], and GIST (The minimalist upper ontology)[3]. If we look at the way they describe top-level knowledge we can easily see that different styles of modelling are used to represent very similar things. As a first example consider the modelling of *spatial location*. The entities chosen by GUM and DOLCE to model spatial location are different:

[1]www.loa-cnr.it/DOLCE.html
[2]www.fb10.uni-bremen.de/anglistik/langpro/webspace/jb/gum/
[3]gist-ont.com

Entity	Identifier	Comment
Class	GUM:SpatialLocating	Any configuration whose function it is to locate some physical object in space. Instances must have one locatum and one placement.
Relation	DOLCE-Lite:physical-location	Analytical location holding between physical endurants and physical regions.

While GUM models the spatial location of a physical object as a class, whose function is to connect the physical object and its placement, DOLCE prefers to model it directly as a relation between the object and its location. If we move to another part of the ontologies, modelling the relationship of *cause*, we can see a similar pattern:

Entity	Identifier	Comment
Class	GUM:Causal	This concept defines a Configuration, in which one of the participants is the cause of the other. Hence, one of the participants occupies the role "cause", and the other one "effect" ([2]).
Relation	GIST:CausedByDirect	*No comment present, but from reading the ontology this is the relation holding between two objects in which one is the direct cause of the other.*

Again, while GUM models the causal relation as a class whose function is to connect the "cause" and the "effect", GIST prefers to model it directly as a relation between two objects. Differently from the first example here the two entities do not exactly overlap, but GUM:Causal is more general than GIST:CausedByDirect as GIST:CausedByDirect is restricted only to pairs where one object is the direct cause of the other, while GUM:Causal does not impose any restriction of contiguity between cause and effect. Finally, if we consider how GIST and DOLCE-Light model *membership* we can note the following:

Entity	Identifier	Comment
Class	GIST:Membership	Declared membership in some group or collection (such as the ACM).
Relation	DOLCE-Lite:TemporaryProperPart	Being proper part at time t. It holds for endurants only. This is important to model proper parts that can change or be lost over time without affecting the identity of the whole.

Intuitively, while GIST models membership as a class connecting an object and the group or collection in which this object is part, DOLCE-Light models it as a relation between the two. Again, while the two entities have a strict semantic relation, they do not exactly overlap. In fact GIST:Membership seem to relate only to be part of a group or a collection, and does not seem to include for, instance, being proper part in geographical terms or other different instances of a general relation of being proper part of. Thus, we can say that GIST:Membership is more specific than DOLCE-Lite:TemporaryProperPart.

Heterogeneous representations of this sort are instances of, so-called, *schematic differences*, a problem well studied in schema integration (see, e.g., [3]) but still not deeply investigated in ontology integration. Despite the lack of investigation, schematic differences occur often also in ontologies as they are due to the personal choice of different designers, who can decide to model the same knowledge using different constructs, depending upon their own evaluation of the most appropriate one. In recent papers [11], we have started to address this problem and we have proposed an extension of Distributed Description Logics (DDL) with mappings that allow to map concepts of an ontology into relations of another ontology. Here we investigate the computational properties of reasoning with mappings between homogeneous as well as heterogeneous elements in distributed ontologies: we show that the problem of checking subsumption between concepts is decidable and EXPTIME complete. Moreover, we provide a naïve decision procedure for checking subsumption running in EXPTIME. In addition we generalise the characterisation of the effects of mappings, originally shown in [11] only for the limited case of two ontologies, to an arbitrary network of connected ontologies.

2. A Rich Language for Mappings

Distributed Description Logic (DDL) [17] is a generalisation of the Description Logic (DL) framework designed to formalise multiple ontologies *pairwise* linked by semantic mappings. In DDL, ontologies correspond to description logic theories (T-boxes), while semantic mappings correspond to collections of *bridge rules* ($\mathfrak{B}$). In the following we recall the basic definitions of DDL as defined in [11].

2.1. Distributed Description Logics: the Syntax

Given a non empty set I of indexes, used to identify ontologies, let $\{\mathcal{D}L_i\}_{i\in I}$ be a collection of description logics[4]. For each $i \in I$ let us denote a T-box of $\mathcal{D}L_i$ as $\mathcal{T}_i$. In this paper, we assume that each $\mathcal{D}L_i$ is description logic weaker or at most equivalent to $\mathcal{ALCQI}_b$, which corresponds to $\mathcal{ALCQI}$ with role union, conjunction and difference (see [19]). We indicate with $\{\mathcal{T}_i\}_{i\in I}$ a family of T-Boxes indexed by I. Intuitively, $\mathcal{T}_i$ is the DL formalisation of the i-th ontology. To make every description distinct, we will prefix it with the index of ontology it belongs to. For instance, the concept C that occurs in the i-th ontology is denoted as $i : C$.

Semantic mappings between different ontologies are expressed via collections of *bridge rules*. In the following we use A, B, C and D as place-holders for concepts and

[4]We assume familiarity with Description Logic and related reasoning systems, described in [1].

R, S, P and Q as place-holders for roles. We instead use X and Y to denote both concepts and roles.

Definition 1 (Bridge rules) *A* bridge rule *from i to j is an expression defined as follows:*

$$i : X \xrightarrow{\sqsubseteq} j : Y \qquad \text{(into bridge rule)} \qquad (1)$$

$$i : X \xrightarrow{\sqsupseteq} j : Y \qquad \text{(onto bridge rule)} \qquad (2)$$

where X and Y are concepts or atomic roles.

Into and onto rules between pairs of concepts (pairs of roles) are called *homogeneous bridge rules*. Otherwise they are called *heterogeneous bridge rules*. As the semantics will make clear, into and onto bridge rules are used to express that, from the j-th point of view the element X in i is less general (into rule) or more general (onto rule) than its local element Y. The expression $i : X \xrightarrow{\equiv} j : Y$ is used as a shorthand for a pair of into and onto rules mapping $i : X$ to $j : Y$. Therefore if we look at our initial examples, the exact overlapping between DOLCE-Lite:physical-location and GUM:SpatialLocating can be stated with an equivalence bridge rule of the form

$$\text{GUM:SpatialLocating} \xrightarrow{\equiv} \text{DOLCE-Lite:physical-location} \qquad (3)$$

while the fact that GUM:Causal is more general than GIST:CausedByDirect can be expressed with the onto bridge rule

$$\text{GUM:Causal} \xrightarrow{\sqsupseteq} \text{GIST:CausedByDirect} \qquad (4)$$

and, finally, the fact that GIST:Membership is more specific than DOLCE-Lite:TemporaryProperPart can be expressed with the into bridge rule

$$\text{GIST:Membership} \xrightarrow{\sqsubseteq} \text{DOLCE-Lite:TemporaryProperPart} \qquad (5)$$

Definition 2 (Distributed T-box) *A* distributed T-box *(DTB) $\mathfrak{T} = \langle \{\mathcal{T}_i\}_{i \in I}, \mathfrak{B} \rangle$ consists of a collection $\{\mathcal{T}_i\}_{i \in I}$ of T-boxes, and a collection $\mathfrak{B} = \{\mathfrak{B}_{ij}\}_{i \neq j \in I}$ of bridge rules between them.*

2.2. Distributed Description Logics: the Semantics

The semantic of DDL assigns to each ontology $\mathcal{T}_i$ a *local interpretation domain*. The first component of an interpretation of a DTB is a family of interpretations $\{\mathcal{I}_i\}_{i \in I}$, one for each T-box $\mathcal{T}_i$. Each $\mathcal{I}_i$ is called a *local interpretation* and consists of a *possibly empty* domain $\Delta^{\mathcal{I}_i}$ and a valuation function $\cdot^{\mathcal{I}_i}$, which maps every concept to a subset of $\Delta^{\mathcal{I}_i}$, and every role to a subset of $\Delta^{\mathcal{I}_i} \times \Delta^{\mathcal{I}_i}$. The interpretation on the empty domain is used to provide a semantics for distributed T-boxes in which some of the local T-boxes are inconsistent. The reader interested in this aspect of DDL can refer to [17].

The second component of the DDL semantics are families of domain relations. Domain relations define how the different T-box interact and are necessary to define the satisfiability of bridge rules.

Definition 3 (Domain relation) *A* domain relation r_{ij} *from i to j is a subset of* $\Delta^{\mathcal{I}_i} \times \Delta^{\mathcal{I}_j}$.

A domain relation r_{ij} represents a possible way of mapping the elements of $\Delta^{\mathcal{I}_i}$ into its domain $\Delta^{\mathcal{I}_j}$, seen from j's perspective.

Domain relations are used to interpret homogeneous bridge rules and are illustrated in detail in [17], but do not provide sufficient information to interpret heterogeneous bridge rules. As an example we would like to map the triple

$$\langle \texttt{ballRolling}, \textsf{CausedByDirect}, \texttt{ChiaraKicksBall} \rangle$$

of an ontology modelling the causal relation according to the GIST top-level ontology, into a cause `cause123` of the second ontology, with the intuitive meaning that `cause123` is the cause which correspond to the causal relation between the actions of chiara kicking a ball and the ball rolling.

Let us formally introduce a triple $R[d_1, d_2]$. Let $\mathcal{I}_i$ be a *local interpretation* for $\mathcal{D}L_i$. Let $\mathcal{R}$ be the set of all atomic roles of $\mathcal{D}L_i$. We indicate with $[\mathcal{R}]^{\mathcal{I}_i}$ the set of all triples $X[d_1, d_2]$ such that $X \in \mathcal{R}$ and $(d_1, d_2) \in X^{\mathcal{I}_i}$. We call $[\mathcal{R}]^{\mathcal{I}_i}$ the set of *admissible triples* for $\mathcal{I}_i$. Given a role $R \in \mathcal{R}$, we write $[R]^{\mathcal{I}_i}$ to denote all the admissible triples in $[\mathcal{R}]^{\mathcal{I}_i}$ of the form $X[d_1, d_2]$ with $X \sqsubseteq R$. Intuitively, $\langle \texttt{ballRolling}, \textsf{CausedByDirect}, \texttt{ChiaraKicksBall} \rangle$ is an admissible triple in $\Sigma^{\mathcal{I}_i}$ if ball rolling is the direct effect of Chiara kicking the ball, or more formally if the pair $(\texttt{ballRolling}, \texttt{ChiaraKicksBall})$ belongs to the interpretation of $\textsf{CausedByDirect}$ in $\mathcal{I}^i$.

Definition 4 (Concept-role and role-concept domain relation) *A* concept-role domain relation cr_{ij} *from i to j is a subset of* $\Delta^{\mathcal{I}_i} \times [\mathcal{R}]^{\mathcal{I}_j}$ *such that if* $(d, X[d_1, d_2]) \in cr_{ij}$ *and* $X^{\mathcal{I}_j} \subseteq Y^{\mathcal{I}_j}$ *with Y atomic role, then* $(d, Y[d_1, d_2]) \in cr_{ij}$. *A* role-concept domain relation rc_{ij} *from i to j is a subset of* $[\mathcal{R}]^{\mathcal{I}_i} \times \Delta^{\mathcal{I}_j}$ *such that if* $(X[d_1, d_2], d) \in rc_{ij}$ *and* $X^{\mathcal{I}_i} \subseteq Y^{\mathcal{I}_i}$ *with Y atomic role, then* $(Y[d_1, d_2], d) \in rc_{ij}$.

The domain relation rc_{ij} represents a possible way of mapping pairs of $R^{\mathcal{I}_i}$ into elements of $C^{\mathcal{I}_j}$, seen from j's perspective. For instance,

$$(\langle \texttt{ballRolling}, \textsf{CausedByDirect}, \texttt{ChiaraKicksBall} \rangle, \texttt{cause123}) \in cr_{ij} \quad (6)$$

represents the fact that, from the point of view of j, `cause123` is an object in its own ontology corresponding to the causal relation between `ballRolling` and `ChiaraKicksBall` in ontology i. The additional condition on $X^{\mathcal{I}_j} \subseteq Y^{\mathcal{I}_j}$ is used to ensure that cr_{ij} is consistent with the hierarchy of roles. Analogously for cr_{ij}.

Definition 5 (Distributed interpretation) *A* distributed interpretation $\mathfrak{I}$ *of a DTB* $\mathfrak{T}$ *consists of the 4-tuple* $\langle \{\mathcal{I}_i\}_{i \in I}, \{r_{ij}\}_{i \neq j \in I}, \{cr_{ij}\}_{i \neq j \in I}, \{rc_{ij}\}_{i \neq j \in I} \rangle$.

In order to define the satisfiability of bridge rules we introduce some functional notation for domain relations and for roles. Given a (regular, concept-role, role-concept) domain relation dr_{ij}, we write $dr_{ij}(t)$ to denote the set of objects t' such that (t, t') is in dr_{ij}. Analogously, given a set $T = \{t_1, t_2, \ldots\}$, we write $dr_{ij}(T)$ to denote the union of all $dr_{ij}(t_i)$ with $t_i \in T$.

Definition 6 (Satisfiability of bridge rules) *A distributed interpretation $\mathfrak{I}$ satisfies a bridge rule br, written as $\mathfrak{I} \models br$, when*

- *homogeneous bridge rules*

$$\mathfrak{I} \models i : X \xrightarrow{\sqsubseteq} j : Y \quad \textit{if} \quad r_{ij}(X^{\mathcal{I}_i}) \subseteq Y^{\mathcal{I}_j} \tag{7}$$

$$\mathfrak{I} \models i : X \xrightarrow{\sqsupseteq} j : Y \quad \textit{if} \quad r_{ij}(X^{\mathcal{I}_i}) \supseteq Y^{\mathcal{I}_j} \tag{8}$$

- *heterogeneous bridge rules*

$$\mathfrak{I} \models i : C \xrightarrow{\sqsubseteq} j : R \quad \textit{if} \quad cr_{ij}(C^{\mathcal{I}^i}) \subseteq [R]^{\mathcal{I}_j} \tag{9}$$

$$\mathfrak{I} \models i : C \xrightarrow{\sqsupseteq} j : R \quad \textit{if} \quad cr_{ij}(C^{\mathcal{I}^i}) \supseteq [R]^{\mathcal{I}_j} \tag{10}$$

$$\mathfrak{I} \models i : R \xrightarrow{\sqsubseteq} j : C \quad \textit{if} \quad rc_{ij}([R]^{\mathcal{I}^i}) \subseteq C^{\mathcal{I}_j} \tag{11}$$

$$\mathfrak{I} \models i : R \xrightarrow{\sqsupseteq} j : C \quad \textit{if} \quad rc_{ij}([R]^{\mathcal{I}^i}) \supseteq C^{\mathcal{I}_j} \tag{12}$$

Satisfiability of into bridge rules forces the appropriate domain relation to map objects of the left hand side element of the bridge rule into objects of the right hand side element. Analogously all the onto bridge rules ensure that each object of the right hand side element has at least a pre-image, via the appropriate domain relation, which is in the left hand side element of the rule.

A distributed interpretation $\mathfrak{I}$ *satisfies* DTB $\mathfrak{T}$ if all the T-boxes $\mathcal{T}_i$ are satisfied by their local interpretation $\mathcal{I}_i$, and if $\mathfrak{I}$ satisfies all the bridge rules in $\mathfrak{B}_{ij}$. Entailment and satisfiability of a single concept are defined in the usual way by means of the satisfiability of a distributed T-Box. The reader interested in the formal definitions can refer to [11].

3. The Effects of Bridge Rules

Bridge rules can be thought of as inter-theory axioms, which constrain the models of the theories representing the different ontologies. An important characteristic of mappings specified by DDL bridge rules is that they are directional, in the sense that they are defined from a source ontology O_s to a target ontology O_t, and they allow to transfer knowledge only from O_s to O_t, without any undesired back-flow effect. As a consequence, given a simple DTB $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$, composed of two T-boxes $\mathcal{T}_i$ and $\mathcal{T}_j$ and a set of bridge rules $\mathfrak{B}_{ij}$ from i to j, we can characterise the knowledge propagated from i (the source) to j (the target) using a set of *propagation rules* of the form:

$$\frac{\begin{array}{l}\text{axioms in } i \\ \text{bridge rules from } i \text{ to } j\end{array}}{\text{axiom in } j}$$

which must be read as: if $\mathcal{T}_i$ entails all the axioms in i, and $\mathfrak{B}_{ij}$ contains the bridge rules from i to j, then $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$ satisfies axioms in j. In [11] we show that the propagation rules shown in Figure 1 provide a sound and compete characterisation of the way knowledge propagates from i to j. Note that in order to simplify the notation of the rules, we assume that for every (into or onto) bridge rule between roles, the same rule appears also for the inverses of the roles themselves. I.e., if $i : P \longrightarrow j : R$ is in $\mathfrak{B}_{ij}$, then also $i : inv(P) \longrightarrow j : inv(R)$ is in $\mathfrak{B}_{ij}$ (where $inv(X)$ is the inverse of X). Also in the left hand side rule, bottom row, we assume that $i : S \xrightarrow{\sqsubseteq} j : D$ can also be of the form $i : \bot_R \xrightarrow{\sqsubseteq} j : \bot$ where $\bot_R$ is the empty role and $\bot$ the empty concept.

$$\frac{\begin{array}{l} i : A \sqsubseteq \bigsqcup_{k=1}^{n} B_k \\ i : A \xrightarrow{\sqsupseteq} j : C \\ i : B_k \xrightarrow{\sqsubseteq} j : D_k, \text{ for } 1 \leq k \leq n \end{array}}{j : C \sqsubseteq \bigsqcup_{k=1}^{n} D_k}$$

$$\frac{\begin{array}{l} i : \exists(P \sqcap \neg(\bigsqcup_{h=1}^{l} Q_h)).(\neg \bigsqcup_{h=1}^{p} A_h) \sqsubseteq (\bigsqcup_{h=1}^{m} B_h) \\ i : P \xrightarrow{\sqsupseteq} j : R \\ i : Q_h \xrightarrow{\sqsubseteq} j : S_h, \text{ for } 1 \leq h \leq l \\ i : A_h \xrightarrow{\sqsubseteq} j : C_h, \text{ for } 1 \leq h \leq p \\ i : B_h \xrightarrow{\sqsubseteq} j : D_h, \text{ for } 1 \leq h \leq m \end{array}}{j : \exists(R \sqcap \neg(\bigsqcup_{h=1}^{l} S_h)).(\neg \bigsqcup_{h=1}^{p} C_h) \sqsubseteq (\bigsqcup_{k=1}^{m} D_k)}$$

$$\frac{\begin{array}{l} i : P \sqsubseteq Q \\ i : P \xrightarrow{\sqsupseteq} j : C \\ i : Q \xrightarrow{\sqsubseteq} j : D \end{array}}{j : C \sqsubseteq D}$$

$$\frac{\begin{array}{l} i : A \sqsubseteq \bigsqcup_{k=1}^{n} B_k \\ i : A \xrightarrow{\sqsupseteq} j : R \\ i : B_k \xrightarrow{\sqsubseteq} j : S_k, \text{ for } 1 \leq k \leq n \end{array}}{j : R \sqsubseteq \bigsqcup_{k=1}^{n} S_k}$$

Figure 1. Propagation rules for homogeneous and heterogeneous mappings.

In the next section we recall the technique we have used to prove soundness and completeness for the simple T-box $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$, together with the extension of the proof to the case of an arbitrary distributed T-box, and the characterisation of computational complexity. Instead here we focus on some characteristics and examples of usage of the bridge rules of Figure 1. To provide an intuition of the propagation rules in Figure 1, we consider their simplest form:

$$\frac{\begin{array}{l} i : X_1 \sqsubseteq X_2 \\ i : X_1 \xrightarrow{\sqsupseteq} j : Y_1 \\ i : X_2 \xrightarrow{\sqsubseteq} j : Y_2 \end{array}}{j : Y_1 \sqsubseteq Y_2} \tag{13}$$

Into and onto bridge rules are combined in rule (13) to propagate hierarchies from T_i to $\mathcal{T}_j$: homogeneous bridge rules propagate the concept (role) hierarchy of $\mathcal{T}_i$ into the analogous hierarchy of $\mathcal{T}_j$; while heterogeneous bridge rules allow to transform a concept hierarchy into a role hierarchy and vice-versa.

To exemplify the interaction of the heterogeneous bridge rules let us assume that we have to align different top-level ontologies such as GUM and DOLCE-Light. This alignment task is quite complex, as top-level ontologies often contain exhaustive descriptions of very general concepts, that can be formalised in different manners and at different levels of detail. In addition, as we have seen, the choice of modelling certain entities as concepts or relations can vary and therefore

A computer-based or human-based mapping process could identify the mapping

$$\text{GUM:SpatialLocating} \xrightarrow{\equiv} \text{DOLCE-Lite:physical-location}$$

described in Equation (3), and in addition could also identify a mapping

$$\text{GUM:SpatialTemporalLocating} \xrightarrow{\equiv} \text{DOLCE-Lite:spatio-temporally-present-at}$$

However, while in GUM SpatialTemporalLocating $\sqsubseteq$ SpatialLocating holds, this hierarchical relation between classes is not reflected in the hierarchy of roles of DOLCE-Light, as the relations physical-location and spatio-temporally-present-at are not one a sub-role of the other. As a consequence, the application of rule (13) would allow to infer the new fact spatio-temporally-present-at $\sqsubseteq$ physical-location in DOLCE-Light. Even assuming that the addition of this new fact would not violate consistency of DOLCE-Light, nevertheless it highlights a problem in the proposed alignment. In fact, we have to remember that we are aligning top-level ontologies, which have been carefully studied and crafted. Therefore we have to assume that if the designers of DOLCE-Light have decided not to have an axiom spatio-temporally-present-at $\sqsubseteq$ physical-location then this must remain valid also after the alignment. This could suggests to better check the comments and properties of the entities involved in the mappings, thus discovering that spatio-temporally-present-at and physical-location are relations with different domains, and for instance refine the original mapping

$$\text{GUM:SpatialTemporalLocating} \xrightarrow{\equiv} \text{DOLCE-Lite:spatio-temporally-present-at}$$

into the weaker

$$\text{GUM:SpatialTemporalLocating} \xrightarrow{\sqsubseteq} \text{DOLCE-Lite:spatio-temporally-present-at}$$

following the approach proposed in [15].

4. Decidability, Computational Complexity, and Generalisation of Completeness

In [11] we have defined a sound and complete operator $\mathfrak{B}_{ij}(\cdot)$ which computes the logical consequence relation in a distributed T-boxes of the form $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$. Here we recall that result and we extend it by showing that: (1) the problem of checking the subsumption among concepts in distributed T-boxes of the form $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$ is decidable and ExpTime complete; and (2) we generalise the definition of $\mathfrak{B}_{ij}(.)$ to compute the logical consequence relation in an arbitrary distributed T-box.

Definition 7 (The operator $\mathfrak{B}_{ij}(\cdot)$) *Let $\mathcal{T}_i$ be a T-box with index i and $\mathfrak{B}_{ij}$ be a set of bridge rules from i to j. We define $\mathfrak{B}_{ij}(\mathcal{T}_i)$ as the set containing all the formulas ϕ obtained as conclusion of the rules in Figure 1 such that $\mathcal{T}_i$ satisfies the formulas in i, premises of the rule used to obtain ϕ, and $\mathfrak{B}_{ij}$ contains all the bridge rules of the rule used to obtain ϕ.*

Given a distributed T-box $\mathfrak{T}_{ij} = \langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$ the operator $\mathfrak{B}_{ij}(\cdot)$ allows to characterise *all and only* the inferences that one can get in $\mathcal{T}_j$ by using the facts of T_i and the bridge rules of $\mathfrak{B}_{ij}$. This is stated in the following theorem, the proof of which is described in [9].

Theorem 1 (Soundness and Completeness of $\mathfrak{B}_{ij}(\cdot)$) *Let $\mathfrak{T}_{ij} = \langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$ be a distributed T-box, where $\mathcal{T}_i$ and $\mathcal{T}_j$ are expressed in the $\mathcal{ALCQI}_b$ descriptive language. Then $\mathfrak{T}_{ij} \models j : X \sqsubseteq Y \Longleftrightarrow \mathcal{T}_j \cup \mathfrak{B}_{ij}(\mathcal{T}_i) \models X \sqsubseteq Y$.*

1. $\mathcal{T}_j^* = \emptyset$
2. for any onto-bridge rule between concepts $i : A \xrightarrow{\sqsupseteq} j : C$, and for any combination of into-bridge rules between concepts $i : B_k \xrightarrow{\sqsubseteq} j : D_k$ $(1 \leq k \leq n)$, if $\mathcal{T}_i \models A \sqsubseteq \bigsqcup_{k=1}^{n} B_k$, then $\mathcal{T}_j^* = \mathcal{T}_j^* \cup \{C \sqsubseteq \bigsqcup_{k=1}^{n} D_k\}$.
3. for any onto-bridge rule between roles $i : P \xrightarrow{\sqsupseteq} j : R$, for any combination of into-bridge rules between roles $i : Q_h \xrightarrow{\sqsubseteq} j : S_h$ $(1 \leq h \leq l)$, and into-bridge rules between concepts $i : A_h \xrightarrow{\sqsubseteq} j : C_h$ $(1 \leq h \leq p)$, and $i : B_h \xrightarrow{\sqsubseteq} j : D_h$ $(1 \leq h \leq m)$, if $\mathcal{T}_i \models \exists(P \sqcap \neg(\bigsqcup_{h=1}^{l} Q_h)).(\neg \bigsqcup_{h=1}^{p} A_h) \sqsubseteq (\bigsqcup_{h=1}^{m} B_h)$, then $\mathcal{T}_j^* = \mathcal{T}_j^* \cup \{\exists(R \sqcap \neg(\bigsqcup_{h=1}^{l} S_h)).(\neg \bigsqcup_{h=1}^{p} C_h) \sqsubseteq (\bigsqcup_{k=1}^{m} D_k)\}$
4. for any pair of role-into/onto-concept bridge rules $i : P \xrightarrow{\sqsupseteq} j : C$ and $i : Q \xrightarrow{\sqsupseteq} j : D$, if $\mathcal{T}_i \models P \sqsubseteq Q$, the $\mathcal{T}_j^* = \mathcal{T}_j^* \cup \{C \sqsubseteq D\}$.
5. for any role-onto-concept bridge rules $i : P \xrightarrow{\sqsupseteq} j : C$ if $\mathcal{T}_i \models P \sqsubseteq \bot_R$, the $\mathcal{T}_j^* = \mathcal{T}_j^* \cup \{C \sqsubseteq \bot\}$.
6. for any concept-onto-role bridge rules $i : A \xrightarrow{\sqsupseteq} j : R$, and and for any combination of concept-into-role bridge rules $j : B_k \xrightarrow{\sqsupseteq} j : S_k$ $(1 \leq k \leq n)$, if $\mathcal{T}_i \models A \sqsubseteq \sqcup_{k=1}^{n} B_k$, then $\mathcal{T}_j^* = \mathcal{T}_j^* \cup \{R \sqsubseteq \sqcup_{k=1}^{n} S_k\}$.
7. return $\mathcal{T}_j^*$.

Figure 2. Computing $\mathcal{T}_j^* = \mathcal{T}_j \cup \mathfrak{B}_{ij}(\mathcal{T}_i)$

4.1. Checking Subsumption in a Distributed T-Box

As a preliminary observation, we can use Theorem 1 to observe that if $\mathcal{T}_i$ and $\mathfrak{B}_{ij}$ are finite, it is possible to finitely pre-compile in a sound and complete manner the subsumption information imported into a local ontology $\mathcal{T}_j$ form another ontology $\mathcal{T}_i$ via the bridge rules $\mathfrak{B}_{ij}$. Therefore, a naïve decision procedure for checking subsumption in a distributed T-Box $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$ can be defined first by computing $\mathcal{T}_j^* = \mathcal{T}_j \cup \mathfrak{B}_{ij}(\mathcal{T}_i)$ according to procedure described in Figure 2, and then by applying the following function

$$\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle \models k : X \sqsubseteq Y \text{ iff } \begin{cases} \mathcal{T}_i \models X \sqsubseteq Y & \text{if } k = i \\ \mathcal{T}_j^* \models X \sqsubseteq Y & \text{if } k = j \end{cases}$$

Lemma 1 *Let $\mathfrak{T} = \langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$, such that $\mathfrak{B}_{ij}$ contains cc_i concept-into-concept bridge rules, cc_o concept-onto-concept bridge rules, rr_i role-into-role bridge rules, rr_o role-onto-role bridge rules, rc_i role-into-concept bridge rules, rc_o role-onto-concept bridge rules, cr_i concept-into-role bridge rules, and cr_o concept-onto-role bridge rules, then $\mathfrak{B}_{ij}(\mathcal{T}_j)$ is finite and contains at most the following number of formulas:*

$$cc_o 2^{cc_i} + rr_o(2^{cc_i+1} + 2^{rr_i}) + rc_o(rc_i + 1) + cr_o 2^{cr_i} \tag{14}$$

Proof 1 *It is sufficient to see that (14) is the upper-bound of the number of different conclusions that can be inferred by applying the five propagation rules of Figure 1, which are used by steps (2)-(6) of the procedure shown in Figure 2.*

Having shown that $\mathfrak{B}_{ij}(\mathcal{T}_i)$ is finite we can now use it to check subsumption in a distributed T-Box $\langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle$ using $\mathcal{T}_j^* = \mathcal{T}_j \cup \mathfrak{B}_{ij}(\mathcal{T}_i)$.

Theorem 2 (Decidability) *The problem of checking if $\mathfrak{T} = \langle \mathcal{T}_i, \mathcal{T}_j, \mathfrak{B}_{ij} \rangle \models k : X \sqsubseteq Y$ with $k = i, j$ is decidable and its complexity is between* EXPTIME *and* 2EXPTIME.

Proof 2 (Outline) *If $k = i$, then checking $\mathfrak{T} \models k : X \sqsubseteq Y$ reduces to check if $\mathcal{T}_i \models X \sqsubseteq Y$, which in $\mathcal{ALCQI}_b$ is* EXPTIME *[19]. If $k = j$, we need exponential time to compute $\mathfrak{B}_{ij}(\mathcal{T}_i)$, possibly obtaining a exponential blow up of the target ontology, and again exponential time to check if $\mathcal{T}_j \cup \mathfrak{B}_{ij}(\mathcal{T}_i) \models X \sqsubseteq Y$. Therefore, the complexity is* 2EXPTIME. *The lower bound is provided by the* EXPTIME *completeness of reasoning in $\mathcal{ALCQI}_b$ (see [19]).*

The naive algorithm shown in Figure 2 may lead to an exponential blow-up of the target ontology, resulting in a double exponential upper bound. However, the algorithm can be optimised by noticing that the rules which propagate ISAs have the same concept (role) on the left hand side; therefore most of the combinations of union of concepts in the right hand side would be redundant. This consideration leads to the fact that the set of additional axioms generated by those two rules can be kept linear in the size of the bridge rules. We strongly believe that this applies to the domain and range propagation rule as well. This is part of our current research, and we believe that we can reduce the upper bound to EXPTIME.

4.2. Dealing With Arbitrary Networks of Ontologies

The operator $\mathfrak{B}_{ij}(.)$ used so far, computes the effects of mappings from an ontology source in i to an ontology target in j. In the following we generalise it in order to deal with an arbitrary (possibly cyclic) distributed T-box.

Let $\mathfrak{B} = \{\mathfrak{B}_{ij}\}_{i,j \in I}$ be a family of bridge rules, and $\mathbf{T} = \{\mathcal{T}_i\}_{i \in I}$ be a family of T-boxes. We define the operator $\mathfrak{B}(.)$ as follows:

$$\mathfrak{B}(\{\mathcal{T}_i\}_{i \in I}) = \left\{ \mathcal{T}_i \cup \bigcup_{j \neq i} \mathfrak{B}_{ji}(\mathcal{T}_j) \right\}_{i \in I}$$

The operator $\mathfrak{B}(.)$ is monotone and we can define $\mathfrak{B}^*(.)$ as the fixpoint of $\mathfrak{B}(.)$. $\mathfrak{B}^*(.)$ allows to characterise all and only the inferences that one can get in a generic distributed T-box $\mathfrak{T} = \langle \mathbf{T}, \mathfrak{B} \rangle$.

Theorem 3 (Soundness and Completeness of $\mathfrak{B}^*(.)$) *For every $\mathfrak{T} = \langle \mathbf{T}, \mathfrak{B} \rangle$, $\mathfrak{T} \models j : X \sqsubseteq Y$ if and only if the j-th T-box of $\mathfrak{B}^*(\mathbf{T})$, denoted with $\mathfrak{B}^*(\mathbf{T})_j$, is such that $\mathfrak{B}^*(\mathbf{T})_j \models X \sqsubseteq Y$.*

Proof 3 (Outline) *The proof of soundness is inherited from Theorem 1. To prove completeness we assume that $\mathfrak{B}^*(\mathbf{T})_j \not\models X \sqsubseteq Y$ and we prove that $\mathfrak{T} \not\models j : X \sqsubseteq Y$, that is there exists a model for $\mathfrak{T}$ which does not satisfy $j : X \sqsubseteq Y$.*

We start the construction of this model by taking an $\mathcal{I}_j^0$ as any model of $\mathfrak{B}^*(\mathbf{T})_j$ such that $(X \sqcap \neg Y)^{\mathcal{I}_j^0} \neq \emptyset$, and as $\mathcal{I}_i^0$ with $i \neq j$ as the empty interpretation. For every *natural number $k \geq 0$, for every $j \in I$ and every $i \neq j \in I$, we use the technique used in the proof of completeness of the simple case (Theorem 1) to start from $\mathcal{I}_j^k$ and build a model $\mathcal{I}_{i_j}^{k+1}$ of $\mathcal{T}_i$ and a domain relation $r_{i_j j}^k$ from $\mathcal{I}_{i_j}^{k+1}$ to $\mathcal{I}_j^k$, such that $\left\langle \mathcal{I}_{i_j}^{k+1}, \mathcal{I}_j^k, r_{i_j j}^k \right\rangle$ satisfies the bridge rules in $\mathfrak{B}_{ji}$.*

Without loss of generality we can assume that the domains of all $\mathcal{I}_{i_j}^{k+1}$ with $j \neq i \in I$ are pairwise disjoint. We therefore define $\mathcal{I}_i^{k+1} = \bigcup_{j \neq i \in I} \mathcal{I}_{i_j}^{k+1}$ and $r_{ij}^k = \bigcup_{j \neq i} r_{i_j}^k$, and we are guaranteed by construction that for each $i \neq j \in I$, $\langle \mathcal{I}_i^{k+1}, \mathcal{I}_j^k, r_{ij}^k \rangle$ satisfies the bridge rules in $\mathfrak{B}_{ij}$.

Again we can safely assume that the domains of all $\mathcal{I}_i^k$'s are pairwise disjoint, for each $k \geq 0$. Thus, for each $i \in I$ and $j \neq i \in I$, we define

$$\mathcal{I}_i = \bigcup_{k=1}^{\infty} \mathcal{I}_i^k \qquad\qquad r_{ij} = \bigcup_{k=1}^{\infty} r_i^k$$

Again we can easily show that for all $i \neq j \in I$ we have that $\langle \mathcal{I}_i, \mathcal{I}_j, r_{ij} \rangle \models \mathfrak{B}$. Furthermore, by construction $\mathcal{I}_i \models \mathcal{T}_i$ for all $i \in I$. This means that $\langle \{\mathcal{I}_i\}_{i \in I}, \{r_{ij}\}_{i \neq j \in I} \rangle$ is a model for $\mathfrak{T}$. To prove that $\langle \{\mathcal{I}_i\}_{i \in I}, \{r_{ij}\}_{i \neq j \in I} \rangle$ does not satisfy $j : X \sqsubseteq Y$ it is enough to observe that for all concept expressions C we have that $C^{\mathcal{I}_j^k} \subseteq C^{\mathcal{I}_j}$. Thus from the fact tat $(X \sqcap \neg Y)^{\mathcal{I}_j^0} \neq \emptyset$ we can conclude that $(X \sqcap \neg Y)^{\mathcal{I}_j} \neq \emptyset$, and therefore that $\langle \{\mathcal{I}_i\}_{i \in I}, \{r_{ij}\}_{i \neq j \in I} \rangle$ does not satisfy $j : X \sqsubseteq Y$.

To study the computational complexity of the general case, we can observe that if I is finite and each $\mathfrak{B}_{ij}$ is finite, then there is a positive integer b such that for every family of T-boxes $\mathbf{T} = \{\mathcal{T}_i\}_{i \in I}$, $\mathfrak{B}^b(\mathbf{T}) = \mathfrak{B}^{b+1}(\mathbf{T})$. b is upper-bounded by the sum of the upper-bound of each $\mathfrak{B}_{ij}(\mathcal{T}_i)$ of Lemma 1. Therefore computational complexity of the general case is also between EXPTIME and 2EXPTIME.

5. Related Work

Recently, several proposals go in the direction of providing semantic mapping among different ontologies (e.g. [18,17,4,10,20]). However, to the best of our knowledge there is no specific work on heterogeneous mappings as described in this paper. The closest approach to ours is in [10], where it is presented a framework that allows any kind of arbitrary heterogeneous mapping, which are evaluated in a common "reference" interpretation. Appropriate functions take care of relating local domains with the reference one (*equalising functions*). However, the work contained in [10] is only focused on the representation of heterogeneous mappings and does not provide a calculus for reasoning with the mappings.

This in spite of the fact that there are several attempts at providing some sort of mappings relating non-homogeneous elements. For example in [7], it is possible to express the mapping $\forall x.(\exists y.R(x, y) \to C(x))$; while, in the original version of DDL (see [17]), an analogous mappings can be established by means of the formula $1 : \exists R.\top \xrightarrow{\sqsubseteq} 2 : C$. Note that both cases cannot be considered heterogeneous mappings because they relates the domain of the relation R with the concept C; which are both concepts.

The above mentioned approaches can be generalised by allowing complex queries (e.g. conjunctive queries) in both sides of a mapping (e.g. see [14,6]). Although these techniques solve many integration problems, they don't provide a solution to the cases in which there is a deep mismatch between the relating elements. As exemplified by the example above, some mismatches could be resolved only by relating objects to pairs (tuples) of objects. This cannot be simply done by means of queries without changing the underlying data model. E.g. by adopting an object oriented query language, where tuples have an associated object identifier which could be used to this purpose.

Database schema integration often requires to solve the problem of the reconciliation of type mismatches among elements from different schemata. For this reason, the problem has been considered by several authors (see [3] for a survey). However, most of the proposed solutions require the modification of the original schemata, and they suffer of the same drawbacks of the query based mappings.

The work presented in this paper is clearly connected to the well known modelling process of *reification* (aka *objectification*) adopted in UML or ORM (see [13,16]). As described in [13], this corresponds to think of certain relationship instances as objects. In UML this is supported by means of *association classes*, while in Entity-Relationship diagram this is often mediated by means of *weak entities*. Note that these modelling paradigms do not support rich inter-schema axioms in the spirit of ontology mappings as described in [18].

There are other modelling formalisms which enable the bridging between relations and classes in the context of Description Logics. In particular, the work on $\mathcal{DLR}$ (see [5]), specifically w.r.t. the technique for encoding n-ary relations within a standard Description Logic, and [8]. The advantage of our approach lies in the fact that the local semantics (i.e. the underlying semantics of the single ontology languages) does not need to be modified in order to consider the global semantics of the system. Specifically, there is no need to provide an explicit reification of relations since this is incorporated into the global semantics.

6. Concluding Remarks

The work presented in this paper constitute a genuine contribution in the direction of the integration of heterogeneous ontologies and the study of schematic differences in ontology integration. The language proposed in this paper makes it possible to directly relate concept and relations from different ontologies, and vice-versa. Moreover, the investigation of the computational properties of reasoning with homogeneous and heterogeneous mappings in an arbitrary network of ontologies provides the basis for the definition of more efficient reasoning algorithms.

The kind of heterogeneous mappings supported by the proposed language, in spite of being a step forward in the process of reconciliation of ontology mapping and reification, are not enough to fully capture its semantics. In fact, the reification process not only "lifts" the type of a relation into a class but uses attributes to represent the positional arguments of the relation. We are currently investigating such a problem, and preliminary results are presented in [12].

References

[1] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[2] J. A. Bateman, R. Henschel, and F. Rinaldi. Generalized upper model 2.0: documentation. Technical report, GMD/Institut fuer Integrierte Publikations und Informationssysteme, Darmstadt, Germany, 1995.

[3] C. Batini, M. Lenzerini, and S. B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Comput. Surv.*, 18(4):323–364, 1986.

[4] P. Bouquet, F. Giunchiglia, F. van Harmelen, L. Serafini, and H. Stuckenschmidt. C-OWL: Contextualizing ontologies. In *Second International Semantic Web Conference (ISWC-03)*, volume 2870 of *LNCS*, pages 164–179. Springer Verlag, 2003.

[5] D. Calvanese, D. Berardi, and G. De Giacomo. Reasoning on uml class diagrams. *Artificial Intelligence*, 1(168):70–118, 2005.

[6] D. Calvanese, G. De Giacomo, and M. Lenzerini. A framework for ontology integration. In *Proc. of the 2001 Int. Semantic Web Working Symposium (SWWS 2001)*, 2001.

[7] D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati. Logical foundations of peer-to-peer data integration. In *23rd ACM SIGACT SIGMOD SIGART Sym. on Principles of Database Systems (PODS 2004)*, pages 241–251, 2004.

[8] D. Calvanese, G. De Giacomo, and M. Lenzerini. Structured objects: Modeling and reasoning. In *Deductive and Object-Oriented Databases, DOOD'95*, pages 229–246, 1995.

[9] C.Ghidini, L. Serafini, and S. Tessaris. On relating heterogeneous elements from different ontologies. Technical report, KRDB Technical Reports. Free universiry of Bozen-Bolzano, 2007. Available at http://www.inf.unibz.it/krdb/pub/index.php.

[10] Jérôme Euzenat, François Scharffe, and Antoine Zimmermann. Expressive alignment language and implementation. deliverable D2.2.10, Knowledge Web NoE, 2007.

[11] C. Ghidini, L. Serafini, and S. Tessaris. On relating heterogeneous elements from different ontologies. In *Proc. of the Sixth International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'07)*, volume 4635 of *LNAI*. Springer, 2007.

[12] Chiara Ghidini, Luciano Serafini, and Sergio Tessaris. Bridging heterogeneous representations of binary relations: first results. In Franz Baader, Carsten Lutz, and Boris Motik, editors, *Proceedings of the 21st International Workshop on Description Logics (DL2008)*, volume 353 of *CEUR Workshop Proceedings*, Dresden, Germany, May 13–16 2008.

[13] T. Halpin. Objectification of relationships. In *Proc. of the 10th Int. IfIP WG8.1 Workshop on Exploring Modeling Methods in Systems Analysis and Design (EMMSAD'05)*, 2005.

[14] P. G. Kolaitis. Schema mappings, data exchange, and metadata management. In *PODS '05: Proc. of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 61–75, New York, NY, USA, 2005. ACM Press.

[15] C. Meilicke, H.Stuckenschmidt, and A.Tamilin. Repairing ontology mappings. In *Proceedings of the 22nd Conference on Artificial Intelligence (AAAI-07)*, 2007.

[16] J. Rumbaugh, I. Jacobson, and G. Booch, editors. *The Unified Language Reference Manual*. Addison-Wesley Longman Ltd., 1999.

[17] L. Serafini, A. Borgida, and A. Tamilin. Aspects of distributed and modular ontology reasoning. In *Proc. of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)*, pages 570–575, August 2005.

[18] H. Stuckenschmidt and M. Uschold. Representation of semantic mappings. In *Semantic Interoperability and Integration*, number 04391 in Dagstuhl Seminar Proceedings, 2005.

[19] S. Tobies. *Complexity Results and Practical Algorithms for Logics in Knowledge Representation*. PhD thesis, RWTH Aachen, Germany, 2001.

[20] Antoine Zimmermann. Integrated distributed description logics. In Diego Calvanese, Enrico Franconi, Volker Haarslev, Domenico Lembo, Boris Motik, Anni-Yasmin Turhan, and Sergio Tessaris, editors, *Proceedings of the 20th International Workshop on Description Logics (DL'07)*, volume 250 of *CEUR Workshop Proceedings*, pages 507–514, Brixen-Bressanone, Italy, 8–10 June 2007.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-164

Island Reasoning for $\mathcal{ALCHI}$ Ontologies

Sebastian WANDELT and Ralf MOELLER
Hamburg University of Technology,
21079 Hamburg,
Germany

Abstract. In the last years, the vision of the Semantic Web fostered the interest in reasoning over ever larger sets of assertional statements in ontologies. It is easily conjectured that, soon, real-world ontologies will not fit into main memory anymore. If this was the case, state-of-the-art description logic reasoning systems cannot deal with these ontologies any longer, since they rely on in-memory structures.

We propose a way to overcome this problem by reducing instance checking for an individual in an ontology to a (usually small) relevant subset of assertional axioms. This subset can then be processed by state-of-the-art description logic reasoning systems to perform sound and complete instance checks for the given individual. We think that this technique will support description logic systems to deal with the upcoming large amounts of assertional data.

Keywords. Ontologies, Instance checking, Scalability

Introduction

As the Semantic Web evolves, scalability of inference techniques becomes increasingly important. Even for basic description logic-based inference techniques, e.g. concept satisfiability, it is only recently understood on how to perform reasoning on large ABoxes in an efficient way. This is not yet the case for problems that are too large to fit into main memory.

In this paper we present an approach to execute efficient instance retrieval tests on ontologies, which do not fit into main memory. Existing tableau-based description logic reasoning systems, e.g. Racer [HM01], do not perform well in such scenarios since the implementation of tableau-algorithms is usually built based on efficient in-memory structures. Our contribution is concerned with the following main objective: given an individual a and a concept description C, we want to identify a relevant subset of assertional statements, which are sufficient to decide, whether a is an instance of C or not. The situation is depicted in Figure 1. In the left part there is a graph representing the assertional facts. On the right side a relevant set of assertions is identified for reasoning about individual a. Once we obtained such a (small) subset, called *island*, it can be loaded into a description logic reasoning system for instance checking. Thus, description logic reasoners of any kind can benefit from our proposal.

The intuition is to identify a small island of assertions, s.t. no "new and complex" information can be propagated between the island and the remaining assertional part of the ontology. Although the idea seems straightforward, to the best of our knowledge we

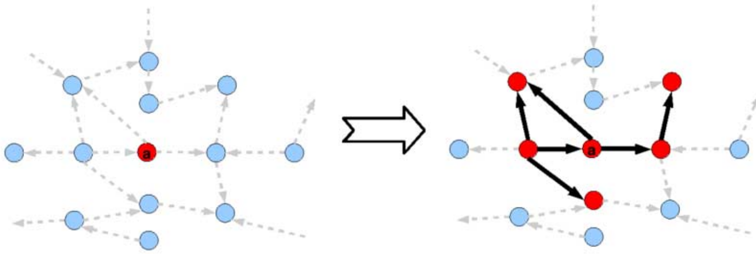


Figure 1. Example: connected subgraph relevant for reasoning about individual a

are the first to propose such an algorithm and evaluate it with respect to large ontologies. There exists previous work on partitioning/modularizing ontologies (e.g. [GH06]) and also on summarization techniques (e.g. [FKM+06]). We will explain below why we think that these techniques are not sufficient in our setting. In a nutshell, the major reasons are:

1. We extract a small subset of assertions, which are worst-case relevant for individual a and concept C.
2. Our approach does not require a precomputation process depending on the ABox. Thus, it is directly applicable to ontologies where the assertional information changes over time. This can be seen as an important step towards dealing with streams of assertional information.

The remaining part of the paper is structured as follows. Section 1 provides the formal background for description logics and also presents some related work. In Section 2 we introduce an example ontology, which will be used throughout the paper. Section 3 analyzes the TBox of given ontologies and Section 4 shows how to use the result of the analysis to rewrite assertional parts of ontologies. Furthermore we we show how instance tests can be restricted to relevant subsets of the assertional information. In Section 5 we present preliminary evaluation of the proposed algorithm and provide further ideas for improvements. We conclude with Section 6.

1. Foundations

1.1. Description Logics

In the following part we will define mathematical notions, which are relevant for the remaining paper. We briefly recall syntax and semantics of the description logic $\mathcal{ALCHI}$. For the details, please refer to [BCM+07]. We assume a collection of disjoint sets: a set of *concept names* N_{CN} , a set of *role names* N_{RN} and a set of *individual names* N_I. The *set of roles* N_R is $N_{RN} \cup \{R^- | R \in N_{RN}\}$. The set of $\mathcal{ALCHI}$*-concept descriptions* is given by the following grammar:

$$C, D ::= \top | \bot | A | \neg C | C \sqcap D | C \sqcup D | \forall R.C | \exists R.C$$

where $A \in N_{CN}$ and $R \in N_R$. We say that a concept description is *atomic*, if it is a concept name. With N_C we denote all atomic concepts. For defining the semantics of concept descriptions and roles we consider *interpretations* $\mathcal{I}$ that consist of a non-empty set $\Delta^{\mathcal{I}}$, the domain, and an interpretation function $\cdot^{\mathcal{I}}$, which assigns to every atomic concept description A a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and to every role R a set $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. For complex concept descriptions the interpretation function is extended as shown in [BCM$^+$07]. The semantics of description logics is based on the notion of satisfiability. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *satisfies* a concept description C if $C^{\mathcal{I}} \neq \emptyset$. In this case, $\mathcal{I}$ is called a *model* for C.

A *TBox* is a set of so-called *generalized concept inclusions*(GCIs) $C \sqsubseteq D$. An interpretation $\mathcal{I}$ *satisfies* a generalized concept inclusion $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. An interpretation is a *model* of a TBox $\mathcal{T}$ if it satisfies all generalized concept inclusions in $\mathcal{T}$. A *RBox* is a set of so-called *role inclusions* $R \sqsubseteq S$ and *role equalities assertion* $R \doteq S$. An interpretation $\mathcal{I}$ *satisfies* a role inclusion $R \sqsubseteq S$ if $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$. An interpretation $\mathcal{I}$ *satisfies* a role equality assertion $R \doteq S$ if $R^{\mathcal{I}} = S^{\mathcal{I}}$. An *ABox* is a set of so-called *concept and role assertions* $a : C$ and $R(a, b)$. An interpretation $\mathcal{I}$ *satisfies* a concept assertion $a : C$ (role assertion $R(a, b)$) if $a \in C^{\mathcal{I}}$ ($(a, b) \in R^{\mathcal{I}}$).

A *ontology* $\mathcal{O}$ consists of a 3-tuple $\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$, where $\mathcal{T}$ is a TBox, $\mathcal{R}$ is a RBox and $\mathcal{A}$ is a ABox. We restrict the concept assertions in $\mathcal{A}$ in such a way that each concept description is an atomic concept or a negated atomic concept. This is a common assumption, e.g. in [GH06], when dealing with large assertional datasets in ontologies. With $Ind(\mathcal{A})$ we denote the set of individuals occurring in $\mathcal{A}$. We say that $\mathcal{O}$ is *inconsistent*, denoted with $INC(\mathcal{O})$, if there exists no model for $\mathcal{O}$. We say that $\mathcal{O}$ is *consistent*, denoted with $CON(\mathcal{O})$, if there exists at least one model for $\mathcal{O}$. Given an individual a and an atomic concept C, we have $\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle \vDash a : C$ if and only if $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \cup \{a : \neg C\} \rangle)$.

In the following we define some additional notions, which will be used throughout the remaining part of the paper. A $\exists$-*constraint* is a concept description of the shape $\exists R.C$, s.t. C is an arbitrary concept description. A $\forall$-*constraint* is a concept description of the shape $\forall R.C$, s.t. C is an arbitrary concept description. A concept description is in *negation normal form* if negation occurs only in front of concept names. We assume the standard transformation $nnf(...)$ into negation normal form[BCM$^+$07].

The *subsumption hierarchy* of parents and children for each concept name can be obtained by classification. For $\mathcal{ALCHI}$ ontology it is possible to compute the subsumption hierarchy in advance given only the TBox $\mathcal{T}$ and RBox $\mathcal{R}$, i.e. without the ABox $\mathcal{A}$. This is possible since $\mathcal{ALCHI}$ does not allow the use of nominals. With $\sqsubseteq_{\mathcal{T}}: N_C \times N_C$ we denote the precomputed subsumption hierarchy obtained by classification, e.g. we have $\sqsubseteq_{\mathcal{T}} (C, D)$ iff $\mathcal{O} \vDash C \sqsubseteq D$ for atomic concepts C and D. The role hierarchy of an $\mathcal{ALCHI}$-ontology can be computed in advance given the RBox $\mathcal{R}$ only. With $\sqsubseteq_{\mathcal{R}}: N_R \times N_R$ we denote the precomputed role hierarchy, e.g. we have $(R, S) \in \sqsubseteq_{\mathcal{R}}$ iff $\mathcal{O} \vDash R \sqsubseteq S$ for roles R and S.

Because of space limitations we do not introduce the notion of tableau proofs w.r.t. description logics, but refer to [BCM$^+$07].

1.2. Related Work

In the following, we discuss selected previously published work related to approximate reasoning and query answering optimization for description logics. Recently, an approach for partitioning large OWL ontologies has been presented in [GH06]. The idea is to partition a large ABox into smaller ABoxes, s.t. reasoning on the smaller assertional subsets is complete, but possibly unsound. Although the authors report impressive results for the increase in performance, we see some problems:

1. To perform instance checking for a particular individual, one has to evaluate all existing partitions. Thus, one ends up loading the whole ontology into memory step-by-step.
2. In fact, the reported average partition size can be orders of magnitudes larger in real-world conditions. Some of these conditions are shown in [Wan08].
3. The approach is not subject to updateable ontologies, i.e. the recomputation of the partitions takes up to several hours/days.

In [FKM$^+$06], the authors propose a method to reduce the number of individuals in an ABox for complete but possibly unsound reasoning. Afterwards, a filtering algorithm is applied to obtain soundness. The idea is to join/summarize similar individuals into one individual and then perform reasoning. This approach is excellent for working with a compact representation of the whole ontology. However, in our setting we are only concerned with these parts of an ontology, which are relevant for a particular (given) individual. In a similar way as the approach given in [GH06], a Summary ABox has to be build in a precomputation step, which depends on the actual ABox. Thus, the approach is not per-se applicable to updateable ontologies.

There is different related work on scalability of query answering by approximation. However, since our work does not involve approximation, we do not discuss these approaches here. After all, our work can be seen as complementary to other work. For more information refer to Section 5. Finally, we should mention the work on QuOnto[ACG$^+$05], which has been the first description logic reasoner that does not use main memory at all to perform ABox reasoning. Their approach rests on the reduced expressiveness of the description logic $DL - Lite$ and can transform queries over ontologies into equivalent and more efficient queries over databases.

2. Guiding example

In the following we define an example ontology, which is used throughout the remaining part of the paper. The ontology is inspired by LUBM [GPH05], a benchmark-ontology in the setting of universities. Although this is a synthetic benchmark, several (if not most) papers on scalability of ontological reasoning consider it as a base reference. We take a particular a snapshot from the LUBM-ontology (TBox, RBox and ABox) and adapt it for presentation purposes. Please note that we do not claim that our snapshot is representative for LUBM. We evaluate our approach w.r.t. to "full" LUBM in Section 5.

Example 1. *Let* $\mathcal{O}_{EX} = \langle \mathcal{T}_{EX}, \mathcal{R}_{EX}, \mathcal{A}_{EX} \rangle$, *s.t.*

$$
\begin{aligned}
\mathcal{T}_{EX} = \{ & \\
& Chair \equiv \exists headOf.Department \sqcap Person, Professor \sqsubseteq Faculty, Book \sqsubseteq Publication, \\
& GraduateStudent \sqsubseteq Student, Student \equiv Person \sqcap \exists takesCourse.Course, \\
& \top \sqsubseteq \forall teacherOf.Course, \exists teacherOf.\top \sqsubseteq Faculty, Faculty \sqsubseteq Person, \\
& \top \sqsubseteq \forall publicationAuthor^{-}.(Book \sqcup ConferencePaper) \\
& \} \\
\mathcal{R}_{EX} = \{ & headOf \sqsubseteq worksFor, worksFor \sqsubseteq memberOf, memberOf \doteq member^{-}\} \\
\mathcal{A}_{EX} = & \texttt{see Figure 2}
\end{aligned}
$$

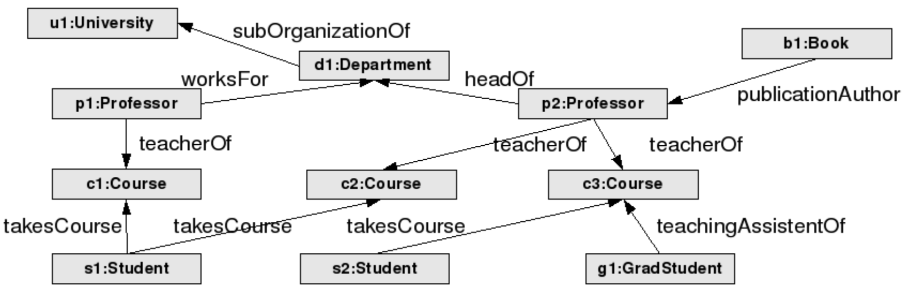


Figure 2. Guiding Example: ABox $\mathcal{A}_{EX}$ for ontology $\mathcal{O}_{EX}$

3. Identification of $\forall$-constraint patterns

In the following part we identify a superset of concepts, which might be propagated over roles during the application of a tableau algorithm to an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$. Please note that we do not claim to find a minimal set of such constraints, but rather a set which allows for worst-case considerations. Although a minimal set would allow a more fine-grained analysis of the ontology $\mathcal{O}$, we think that computing such a minimal set is equivalent to precise reasoning on $\mathcal{O}$, something we assumed as not feasible before. Furthermore, the computation of a minimal set would require us to use information from the ABox in a preprocessing step - something we want to avoid for being able to deal with dynamic assertional information.

First of all, we know that $\forall$-constraints cannot come directly from the ABox, since we only allow for concept assertions of the kind $a : C$, where C is an atomic concept or its negation. Thus, $\forall$-constraints can only be derived from the TBox. Unfortunately, the shape of TBox axioms, i.e. $C \sqsubseteq D$, does not allow to easily read off “possible” $\forall$-constraints. This is due to the implicit presence of negation in the concept C. We propose some kind of normal form, which allows for extraction of a superset of $\forall$-constraints, which can possibly be used in a tableau algorithm.

Definition 1. *A concept description* C *is in* Shallow Normal Form *(SNF), if it has the shape* $C = C_1 \sqcup C_2 \sqcup ... \sqcup C_n$*, s.t. each* C_i *is either*

- *an atomic concept,*
- *a negated atomic concept,*
- *an* $\exists$*-constraint* $\exists R.D$*, s.t. D is an arbitrary concept description in negation normal form*
- *a* $\forall$*-constraint* $\forall R.D$*, s.t. D is an arbitrary concept description in negation normal form*

Please note that we do not enforce anything on concepts "hidden" behind $\forall/\exists$-constraints, but that they are in negation normal form. Thus the name *shallow* normal form.

Lemma 1. *Each GCI* $C \sqsubseteq D$ *can be converted into a set* S *of equivalent concept descriptions in SNF. Here, equivalent means that* $C \sqsubseteq D$ *is unsatisfiable iff the conjunction of the formulas in* S *is unsatisfiable.*

Proof. Sketch: We know that $(C \sqsubseteq D)$ is unsatisfiable if and only if $nnf(\neg C \sqcup D)$ is unsatisfiable. Starting with $nnf(\neg C \sqcup D)$, we can obtain a set of concept descriptions by applying equivalence preserving rules to bring conjunctions to the "outside" and break formulas up into SNF. $\square$

With $Shallow(\mathcal{T})$ we denote the set of concept descriptions, which are equivalent to the TBox inclusions in $\mathcal{T}$. Although the transformation into any conjunctive normal form can yield an exponential blow-up in the worst case, we claim that it is feasible to convert a TBox for our purposes. For details please refer to Section 5. Using Lemma 1, in the following we will assume w.l.o.g that a TBox is given as a set of concept descriptions in SNF. Next, we give a set of concept descriptions in SNF for $\mathcal{T}_{EX}$ from Example 1.

Example 2. *The TBox* $\mathcal{T}_{EX}$ *in SNF is as follows:*

$$
\begin{aligned}
Shallow(\mathcal{T}_{EX}) = \{ \\
& \neg Chair \sqcup \exists headOf.Department, \neg Chair \sqcup Person, \\
& \forall headOf.\neg Department \sqcup \neg Person \sqcup Chair, \neg Professor \sqcup Faculty, \\
& \neg Book \sqcup Publication, \neg GraduateStudent \sqcup Student, \neg Student \sqcup Person, \\
& \neg Student \sqcup \exists takesCourse.Course, \neg Person \sqcup \forall takesCourse.\neg Course \sqcup Student, \\
& \forall teacherOf.Course, \forall teacherOf.\bot \sqcup Faculty, \neg Faculty \sqcup Person, \\
& \forall publicationAuthor^{-}.(Book \sqcup ConferencePaper) \\
& \}
\end{aligned}
$$

In the following we define a structure for managing $\forall$-constrains in an $\mathcal{ALCHI}$-ontology.

Definition 2. *A* $\forall$-info structure *for TBox* $\mathcal{T}$ *is a function* $f_{\mathcal{T}}^{\forall} : N_R \rightarrow \mathcal{P}(N_C \cup \{\neg A | A \in N_C\} \cup \{\bot\}) \cup \{*\}$*, s.t.* N_C *(*N_R*) is a set of atomic concepts (roles) used in* $\mathcal{T}$*. The function* $f_{\mathcal{T}}^{\forall}$ *is used to manage the* $\forall$*-constraints, i.e. the function assigns to each role name in* N_R *one of the following entries:*

Function $build^{\forall}(C, f_{\mathcal{T}}^{\forall})$
Parameter: Concept description C in SNF, $\forall$-info structure $f_{\mathcal{T}}^{\forall}$

1. If $C = C_1 \sqcap ... \sqcap C_n$ or $C = C_1 \sqcup ... \sqcup C_n$ then
   (a) For $1 < i < n$ do $build^{\forall}(C_i, f_{\mathcal{T}}^{\forall})$
2. Else If $C = \exists R.C_1$ then
   (a) $build^{\forall}(C_1, f_{\mathcal{T}}^{\forall})$
3. Else If $C = \forall R.C_1$ then
   (a) If C_1 is an atomic concept or a negated atomic concept or $\bot$ then
      i. If $f_{\mathcal{T}}^{\forall}(R) \neq *$ then $f_{\mathcal{T}}^{\forall}(R) = f_{\mathcal{T}}^{\forall}(R) \cup \{C_1\}$
   (b) else
      i. $f_{\mathcal{T}}^{\forall}(R) = *$
      ii. $build(C_1, f_{\mathcal{T}}^{\forall})$
4. Return

Function $build^{\forall}(\mathcal{T})$
Parameter: TBox $\mathcal{T}$
For each $R \in N_R$ do initialize
$f_{\mathcal{T}}^{\forall}(R) = \emptyset$
For each $C \in \mathcal{T}$ (in SNF) do
$build^{\forall}(C, f_{\mathcal{T}}^{\forall})$
Return $f_{\mathcal{T}}^{\forall}(R)$

Figure 3. Building $f_{\mathcal{T}}^{\forall}$

- $\emptyset$, *if we know that there is no* $\forall$-*constraint for* R *in* $\mathcal{T}$
- *a subset* S *of* $N_C \cup \{\neg A | A \in N_C\} \cup \{\bot\}$, *s.t. there is no other concept but those in* S, *which occur* $\forall$-*bound (i.e. they are a subconcept of a* $\forall$-*constraint) on* R *in* $\mathcal{T}$
- $*$, *if there are arbitrary complex* $\forall$-*constraints on role* R *in* $\mathcal{T}$, *but we don't give additional information on the structure of these constraints.*

The intuition for $f_{\mathcal{T}}^{\forall}$ is defined below. In Figure 3 we propose an algorithm to compute $f_{\mathcal{T}}^{\forall}$. We do not explain the algorithm, since it is a straightforward computation of the closure of $\mathcal{T}$ in SNF. Next, we lift the notion of a $\forall$-*info structure* from a TBox to an ontology.

Definition 3. *A* $\forall$-info structure *for ontology* $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ *is a function* $f_{\mathcal{O}}^{\forall} : N_R \rightarrow \mathcal{P}(N_C \cup \{\neg A | A \in N_C\} \cup \{\bot\}) \cup \{*\}$, *s.t.*

$$f_{\mathcal{O}}^{\forall}(R) = \begin{cases} * & if \exists S \in N_R. \sqsubseteq_R (R, S) \wedge (f_{\mathcal{T}}^{\forall}(S) = *) \\ \bigcup_{R \sqsubseteq_{\mathcal{R}} S} f_{\mathcal{T}}^{\forall}(S) & else \end{cases}$$

Let us look at our ontology $\mathcal{O}_{EX}$ again to give an example for a $\forall$-info structure.

Example 3. *The* ∀-info structure *for ontology $\mathcal{O}_{EX}$ is as follows:*

$$f^{\forall}_{\mathcal{O}_{EX}}(R) = \begin{cases} \{\neg Department\} & \textit{if } R = headOf \\ \{\neg Course\} & \textit{if } R = takesCourse \\ \{\bot, Course\} & \textit{if } R = teacherOf \\ * & \textit{if } R = publicationAuthor^{-} \\ \{\} & \textit{else} \end{cases}$$

The main result of this section is presented in the following lemma:

Lemma 2. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$, the following holds: if $f^{\forall}_{\mathcal{O}}(R) \neq *$, then for each valid tableau proof P of $\mathcal{O}$ and for each application of a ∀-rule (on R and subconcept C) in P, we have that $C \in f^{\forall}_{\mathcal{O}}(R)$.*

Proof. Sketch: By contradiction. W.l.o.g. assume that there exists a tableau proof P, s.t. during the proof the ∀-rule is applied to individual a, which is labeled with $\forall R.C$ and $C \notin f^{\forall}_{\mathcal{O}}(R)$. Since we only allow ABox assertions for atomic concepts, the ∀-constraint must come from the TBox. Thus $\forall R.C$ must be a subconcept of $\mathcal{T}$ in SNF. Since $f^{\forall}_{\mathcal{O}}$ is build by computing the closure of the $\mathcal{T}$, we have either $f^{\forall}_{\mathcal{O}}(R) = *$ or $C \in f^{\forall}_{\mathcal{O}}(R)$. Contradiction. □

4. ABox Rewriting and Island Computation

Given the notions in the previous section, in the following we will derive means to rewrite an ontology $\mathcal{O}$, s.t. inconsistency is preserved, i.e, $INC(\mathcal{O})$ if and only if $INC(\mathcal{O}_{rewritten})$. Inconsistency tests are important for instance checking, since it holds that $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \cup \{a : \neg C\}\rangle)$ if and only if $\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle \vDash a : C$. Please note that we are not going to rewrite the ABoxes in practice, but rather use rewriting for proving soundness and completeness of our algorithm for island computation. The following definition of $\mathcal{O}$-separability is used to determine the importance of role assertions in a given ABox. Informally speaking, the idea is that $\mathcal{O}$-separable assertions will never be used to propagate "complex and new information" (see below) via role assertions.

Definition 4. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$, a role assertion $R(a, b)$ is called $\mathcal{O}$-*separable*, if we have $INC(\mathcal{O}) \iff INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2\}\rangle)$, where*

$$\mathcal{A}_2 = \mathcal{A} \setminus \{R(a, b)\} \cup \{R(a, i_1), R(i_2, b)\} \cup \{i_1 : C | b : C \in \mathcal{A}\} \cup \{i_2 : C | a : C \in \mathcal{A}\},$$

s.t. i_1 and i_2 are fresh individual names.

Given the above definitions, we propose a formal criterion on role assertions w.r.t. an ontology $\mathcal{O}$ to distinguish, whether they are $\mathcal{O}$-separable.

Lemma 3. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ and a role assertion $R(a, b) \in \mathcal{A}$, it holds that $R(a, b)$ is $\mathcal{O}$-separable, if we have*

1. *For each $C \in f^{\forall}_{\mathcal{O}}(R)$*

(a) $C = \bot$ *or*
(b) *we can find a concept description* $D \in \{E|b : E \in \mathcal{A}\}$*, s.t. we have* $D \sqsubseteq_{\mathcal{T}} C$

2. *For each* $C \in f^{\forall}_{\mathcal{O}}(R^{-})$
(a) $C = \bot$ *or*
(b) *we can find a concept description* $D \in \{E|a : E \in \mathcal{A}\}$*, s.t. we have* $D \sqsubseteq_{\mathcal{T}} C$

Proof. We have to show that $INC(\mathcal{O}) \iff INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2\}\rangle)$, where

$$\mathcal{A}_2 = \mathcal{A} \setminus \{R(a,b)\} \cup \{R(a,i_1), R(i_2,b)\} \cup \{i_1 : C|b : C \in \mathcal{A}\} \cup \{i_2 : C|a : C \in \mathcal{A}\},$$

s.t. i_1 and i_2 are fresh individual names. The proof can be done in two steps (directions):

"$\Leftarrow$" To show: $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2\}\rangle) \Rightarrow INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\}\rangle)$.
From $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2\rangle)$ we know that there exists a tableau proof P, which returns a closed tableau. We can apply the same tableau proof to $\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\rangle$, by applying each tableau-rule involving individual i_1 to b and applying each tableau-rule involving individual i_2 to a. Again, we obtain a closed tableau and it holds that $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\}\rangle)$.

"$\Rightarrow$" To show: $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\rangle) \Rightarrow INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2\rangle)$.
Sketch: The idea is that each closed tableau proof P on $\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\rangle$ can be rewritten to a closed tableau proof P_2 on $\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2\rangle$. This is due to the fact that only implicitly known information and immediate clashs are propagated via the split role assertion. This can be shown by induction on $ALCHI$-tableau rules.

□

Let us consider an example for $\mathcal{O}$-separability w.r.t. $\mathcal{O}_{EX}$ from Example 1:

Example 4. *For instance the ABox assertion* $teacherOf(p2, c3)$ *in Example 1 is* $\mathcal{O}_{EX}$*-separable, since we have*

- $f^{\forall}_{\mathcal{O}_{EX}}(teacherOf) = \{\bot, Course\}$ *and* $c3 : Course \in \mathcal{A}_{EX}$
- $f^{\forall}_{\mathcal{O}_{EX}}(teacherOf^{-}) = \{\}$

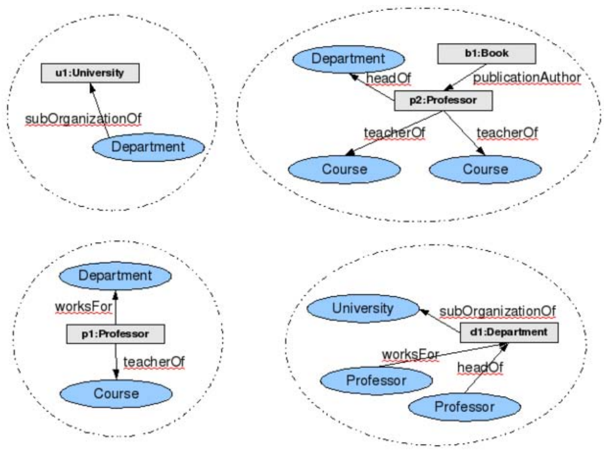
Definition 5. *Given an ontology* $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A}\rangle$*, let* $RED(\mathcal{A})$ *be the ABox computed from* $\mathcal{A}$ *by replacing each* $\mathcal{O}$*-separable role assertion* $R(a,b)$ *by* $\{R(a,i_1), R(i_2,b)\} \cup \{i_1 : C|b : C \in \mathcal{A}\} \cup \{i_2 : C|a : C \in \mathcal{A}\}$ *, s.t.* i_1 *and* i_2 *are fresh individual names.*

Lemma 4. *It holds that* $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\rangle)$ *iff* $INC(\langle \mathcal{T}, \mathcal{R}, RED(\mathcal{A})\rangle)$.

Proof. Easy: By definition of $\mathcal{O}$-separability. □

Definition 6. *An* interconnection-based partitioning *for a ABox* $\mathcal{A}$*, denoted* $P(\mathcal{A}) = \{\mathcal{A}_1, ...\mathcal{A}_n\}$*, is built by role-connectedness, i.e. two individuals are in the same partition iff there exists an explicit role assertion between these two individuals.*

Example 5. *Some parts of the interconnection-based partitioning* $P(RED(\mathcal{A}_{EX}))$ *for* $RED(\mathcal{A}_{EX})$ *are shown below (4 partitions). Rectangular nodes denote individuals from* $\mathcal{A}_{EX}$ *and elliptic nodes denote fresh (unnamed) individuals.*



Lemma 5. *We have* $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle)$ *iff* $\exists \mathcal{A}_i \in P(RED(\mathcal{A})).INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A}_i \rangle)$.

Proof. Easy: If all partitions are disconnected, then there is no interaction between the partitions. □

Definition 7. *Given an ABox* $\mathcal{A}$*, let* $P_a(\mathcal{A})$ *be the partition in* $P(RED(\mathcal{A}))$*, which contains individual* a*.*

Lemma 6. *Given* $CON(\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle)$*, we have that* $INC(\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \cup \{a : C\} \rangle)$ *iff* $INC(\langle \mathcal{T}, \mathcal{R}, P_a(\mathcal{A}) \cup \{a : C\} \rangle)$

Proof. $\Leftarrow$:

Easy, since $P_a(\mathcal{A}) \cup \{a : C\} \subseteq \mathcal{A} \cup \{a : C\}$ (modulo renaming).

$\Rightarrow$:

Let $P_1 = P(RED(\mathcal{A}))$ and $P_2 = P(RED(\mathcal{A} \cup \{a : C\}))$. By definition of RED and P it is clear, that $P_2 \setminus P_1 = \{P_a(RED(\mathcal{A} \cup \{a : C\}))\}$. Since all partitions in P_1 are consistent (by $CON(\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle)$), $P_a(RED(\mathcal{A} \cup \{a : C\}))$ has to be inconsistent (by Lemma 5).

□

Next, we propose an algorithm, which solves the following problem: Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$, check, whether $\mathcal{O} \vDash a : C$ holds for a given individual a and a given concept C, without having to take the whole ontology into consideration. The idea is that we identify a subset S of $\mathcal{A}$, s.t. we have $\mathcal{O} \vDash a : C$ iff $\langle \mathcal{T}, \mathcal{R}, S \rangle \vDash a : C$. The set S is usually orders of magnitudes smaller than the initial ABox $\mathcal{A}$ (for detailed statistics see Section 5). Given Lemma 6, we already have a notion at hand, which identifies a relevant subset of ABox assertions necessary for reasoning about a given individual a. It is easy to define a function, which computes $P_a(\mathcal{A})$, given an individual a (see 4). Soundness and completeness of the algorithm is clear from the definition of $\mathcal{O}$-separability. Termination is ensured by use of a "seen individuals"-list, which avoids following cycles during application of the algorithm.

Function $build(a, seen)$
Parameter: Individual a, list of visited individuals $seen$
Returns: Set S of relevant ABox assertions
Algorithm:

1. If $a \in seen$ then Return $\emptyset$
2. $seen = seen \cup \{s\}$
3. $S = \{a : X \in \mathcal{A}\}$
4. For $R(a, b) \in \mathcal{A}$ do
   (a) If $f^{\forall}_{\mathcal{O}}(R) = *$ or $f^{\forall}_{\mathcal{O}}(R^-) = *$ then
   $S = S \cup \{R(a, b)\} \cup build(b, seen)$
   (b) else if $f^{\forall}_{\mathcal{O}}(R) \neq \emptyset$ or $f^{\forall}_{\mathcal{O}}(R^-) \neq \emptyset$ then
   i. $found = true$
   ii. For $C \in f^{\forall}_{\mathcal{O}}(R)$ do
   If not($C = \bot$ or $(\exists D.b : D \in \mathcal{A} \wedge D \sqsubseteq_{\mathcal{T}} C)$ or $(b : nnf(\neg C) \in \mathcal{A})$) then
   found=false
   iii. For $C \in f^{\forall}_{\mathcal{O}}(R^-)$ do
   If not($C = \bot$ or $(\exists D.a : D \in \mathcal{A} \wedge D \sqsubseteq_{\mathcal{T}} C)$ or $(a : nnf(\neg C) \in \mathcal{A})$) then
   found=false
   (c) If $found = true$ then
   $S = S \cup \{R(a, i_x)\}$, s.t. i_x does neither occur in $Ind(\mathcal{A})$ nor in S
   (d) else
   $S = S \cup \{R(a, b)\} \cup build(b, seen)$
5. For $R(b, a) \in \mathcal{A}$ do
   (a) If $f^{\forall}_{\mathcal{O}}(R) = *$ or $f^{\forall}_{\mathcal{O}}(R^-) = *$ then
   $S = S \cup \{R(b, a)\} \cup build(b, seen)$
   (b) else if $f^{\forall}_{\mathcal{O}}(R) \neq \emptyset$ or $f^{\forall}_{\mathcal{O}}(R^-) \neq \emptyset$ then
   i. $found = true$
   ii. For $C \in f^{\forall}_{\mathcal{O}}(R)$ do
   If not($C = \bot$ or $(\exists D.a : D \in \mathcal{A} \wedge D \sqsubseteq_{\mathcal{T}} C)$ or $(a : nnf(\neg C) \in \mathcal{A})$) then
   found=false
   iii. For $C \in f^{\forall}_{\mathcal{O}}(R^-)$ do
   If not($C = \bot$ or $(\exists D.b : D \in \mathcal{A} \wedge D \sqsubseteq_{\mathcal{T}} C)$ or $(b : nnf(\neg C) \in \mathcal{A})$) then
   found=false
   (c) If $found = true$ then
   $S = S \cup \{R(i_x, a)\}$, s.t. i_x does neither occur in $Ind(\mathcal{A})$ nor in S
   (d) else
   $S = S \cup \{R(b, a)\} \cup build(b, seen)$
6. Return S

Figure 4. Build island for individual a

5. Evaluation and Suggestions for Further Improvements

In the following section we evaluate our proposal for island reasoning. We have implemented the proposed algorithms in Java. For ontology access we used a Java interface and implementation for the Ontology Web Language, called OWLAPI [BVL03]. Given the OWLAPI interface, implementing the above algorithms was straightforward.

Ontology	|Inclusions|	|Equivalences|	$\|N_{RN}\|$	Time for analysis (ms)
LUBM [GPH05]	75	6	25	4
Cyc [CYC05]	43541	2	4853	93
GODaily [HCI04]	28997	0	1	103
Galen [ALRP96]	3388	699	413	55
Pizza [oM08]	57	2	7	3

Figure 5. Extraction of $\forall$-info structures for several ontologies

First, we investigate, whether it is feasible to convert the TBox of a given ontology into Shallow Normal Form and extract the $\forall$-info structure $f_{\mathcal{O}}^{\forall}$. For this purpose we have selected five arbitrary ontologies. For lack of space please refer to the given references for a detailed description of the ontologies. The results of our investigation are shown in Figure 5. We think the numbers indicate that extracting a $\forall$-info structure for most ontologies should be feasible. Even ontologies whose TBox is considered large for current description logic systems do perform quite well.

Second, we look at the average size of extracted islands. Our tests w.r.t. LUBM are quite encouraging. The average size of an island is 29 nodes. The actual island size depends on the chosen individual. Some preliminary statistics on example islands are given in Figure 6. Please note that the island size does not depend on the number of universities, i.e. for LUBM our approach promises quite good scalability.

The big island for $Dep0.Uni0/FullProfessor7$ can be explained as follows. LUBM imposes several $\forall$-constraints on the role $headOf$. Due to the definitions of $Chair$, $Dean$ and $Director$, the atomic concept descriptions $\neg Department$, $\neg College$ and $\neg Program$ can be propagated via role $headOf$. Since all individuals with an incoming $headOf$-edge are of type $Department$, we have to take them completely into account for building the islands. If there were further constraints in the TBox, e.g. disjointness of $Department$, $College$ and $Program$, we could further reduce the size of islands for individuals of type $FullProfessor$.

Individual	**Island size**	**Time for island computation**
$Dep0.Uni0/GraduateStudent128$	9	0 ms
$Dep0.Uni0/Publication2$	4	1 ms
$Dep0.Uni0/FullProfessor7$	93	2 ms
$Dep0.Uni0/Course4$	37	0 ms

Figure 6. Statistics for islands in LUBM

In the following we discuss several ideas for further improvement of island reasoning. With respect to other inference tasks, e.g. instance retrieval, a naive application of island loading can be doomed to failure. For instance, if there are several million named individuals in the ABox of $\mathcal{O}$, then one needs to load one island for each named individual and check, whether this individual is an instance of the concept in question. Thus, while the atomic operation (instance checking) can be performed quite efficiently, the overall run time can be still slow. We propose the following improvements to overcome such problems:

“Preselection” by incomplete/unsound reasoning approaches The idea here is to restrict the set of possible answers to a query by a fast algorithm, which neglects either soundness or completeness. Let $\phi_{sound} : N_C \rightarrow \mathcal{P}(N_A)$ be a sound instance retrieval function, i.e. we have $a \in \phi_{sound}(A) \implies \mathcal{O} \vDash a : A$. Furthermore, let $\phi_{complete} : N_C \rightarrow \mathcal{P}(N_A)$ be a complete instance retrieval function, i.e. we have $\mathcal{O} \vDash a : A \implies a \in \phi_{complete}(A)$.
If we want to perform instance retrieval for an atomic concept A, we only need to check for all individuals $a \in \phi_{complete}(A) \setminus \phi_{sound}(A)$, whether we have $\mathcal{O} \vDash a : A$. For all remaining individuals we do not have to perform explicit instance checking. Please note that the quality of such an improvement depends on the quality of ϕ_{sound} and $\phi_{complete}$. Possible suggestions for such functions are:

$\phi_{complete}$: Initial summary ABox [FKM+06], role condensates [WM07]
ϕ_{sound}: Any precompletion technique [BCM+07]

To sum up, we think that the combination of different scalability-notions will be of importance to cope with the increasing amount of assertional data.

Precomputation of islands in case of static ABoxes Throughout the paper we emphasize the advantages of our proposal for dynamic assertional information. However, if one knows in advance that the ABox will not change at all, our approach can be improved further. For instance, one could precompute the islands for each named individual in advance (offline). Then, whenever the island for individual a is needed, one only needs to load the precomputed island. The approach is even more promising, when you distribute these precomputed island among several computers.

Grouping individuals for "similarity"-islands Let S be a set of individuals for which we want to perform instance checking. In a naive way, we would have to iterate over all individuals in S. However, one could load the *group of islands* for S, by considering $\bigcup_{s \in S} build(s, \emptyset)$. It is easy to see that this group of islands allows for sound and complete reasoning for all individuals in S. Please note that the actual improvement depends on the choice of individuals in S. Usually one would like to group closely-related individuals together. A detailed investigation of this is part of future work.

6. Conclusions and Future Work

We have introduced means to reason over ontologies, which have large amounts of assertional information. Given an individual of interest, island reasoning allows state-of-the-art description logic reasoners to load only relevant subsets of the ABox to perform sound and complete reasoning. In particular, we have proposed a preprocessing step which can be easily performed offline (computation of $\forall$-info structures) and an actual island computation algorithm, which can be run on demand.

To analyze the scalability of our approach, we have investigated several commonly used ontologies. We think that our evaluation shows a clear advantage, since the identified island for a given individual is usually quite small. Furthermore we have proposed additional improvements which are subject to future work. Additionally, we will investigate the applicability of our proposal to more expressive description logics, e.g. SHIQ. The extension for transitive roles is straightforward. The incorporation of min/max-cardinality constraints in a naive way can be done as well. However, it has to be investigated, whether the average island size is still small enough to be feasible in practice.

References

[ACG+05] Andrea Acciarri, Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Mattia Palmieri, and Riccardo Rosati. Quonto: Querying ontologies. In Manuela M. Veloso and Subbarao Kambhampati, editors, *AAAI*, pages 1670–1671. AAAI Press / The MIT Press, 2005.

[ALRP96] J. E. Rogers A. L. Rector and P. A. Pole. The GALEN High Level Ontology. In *Proceedings MIE 96*, pages 174–178. IOS Press, jan 1996.

[BCM+07] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. *The Description Logic Handbook*. Cambridge University Press, New York, NY, USA, 2007.

[BVL03] S. Bechhofer, R. Volz, and P. Lord. Cooking the Semantic Web with the OWL API, 2003.

[CYC05] CYC. Cyc ontology (researchcyc). visited on, February 2005. retrieved from http://www.cyc.com/.

[FKM+06] Achille Fokoue, Aaron Kershenbaum, Li Ma, Chintan Patel, Edith Schonberg, and Kavitha Srinivas. Using Abstract Evaluation in ABox Reasoning. In *SSWS 2006*, pages 61–74, Athens, GA, USA, November 2006.

[GH06] Yuanbo Guo and Jeff Heflin. A Scalable Approach for Partitioning OWL Knowledge Bases. In *SSWS 2006*, Athens, GA, USA, November 2006.

[GPH05] Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. Lubm: A benchmark for owl knowledge base systems. *J. Web Sem.*, 3(2-3):158–182, 2005.

[HCI04] M. A. Harris, J. Clark, and A. et. al. Ireland. The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res*, 32(Database issue), January 2004.

[HM01] V. Haarslev and R. Möller. Description of the RACER System and its Applications. In *Proceedings International Workshop on Description Logics (DL-2001), Stanford, USA, 1.-3. August*, pages 131–141, 2001.

[oM08] The University of Manchester. Pizza ontology. visited on, April 2008. retrieved from http://www.co-ode.org/ontologies/pizza/2007/02/12/.

[Wan08] S. Wandelt. Partitioning OWL Knowledge Bases - Revisited and Revised. In *Proc. International Workshop on Description Logics*, 2008.

[WM07] S. Wandelt and R. Möller. Scalability of OWL Reasoning: Role condensates. In *Proc. International Workshop on Scalable Semantic Web Systems*, 2007.

This page intentionally left blank

5. Scientific Domain Ontologies

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-181

Selecting and Customizing a Mereology Ontology for its Reuse in a Pharmaceutical Product Ontology

Mariano FERNÁNDEZ-LÓPEZ [a, 1]; Asunción GÓMEZ-PÉREZ [b, 2], Mari Carmen SUÁREZ-FIGUEROA [b, 2]
[a]*Escuela Politécnica Superior. Universidad San Pablo CEU*
[b]*Facultad de Informática. Universidad Politécnica de Madrid*

Abstract. This paper presents our experience in reusing mereology ontologies in a Pharmaceutical Product ontology, an ontology built by the EU NeOn project. It shows a set of mereology ontologies implemented in different machine interpretable languages and analyzes them according to the different types of mereology identified by Varzi. Then, it describes the specifications of mereology modeling necessities for Pharmaceutical Product. Finally, it presents the ontology which fits best with the specifications. One of the results of this work is a procedure to reuse general (also called *common*) ontologies.

Keywords. Mereology, implemented ontology, pattern reuse, competency question

Introduction

The part-whole relationship has been analyzed over the ages by philosophers. In the Ancient Greece, by the atomists Plato and Aristotle, in the Middle Ages, by Thomas Aquinas and Raymond Llull, in the Age of Enlightenment by Kant and, at the end of the XIX century and beginning of the XX, by Brentano and Husserl. However, none of them formulated a precise theory on this part-whole relationship. It was Lesniewiski [1] who coined the word *mereology* (from the Greek word *méros*, meaning part) in 1927 to refer to a formal theory he devised in his papers published between 1916 and 1931. However, because Lesniewiski wrote all his papers in Polish, his theory was unknown by most of his contemporary scientists. Leonard and Goodman's work "*The Calculus of the Individuals*" [2] in 1940 made that this formal theory began to be studied in Logics and Ontology. Later on, authors such as Simons [3], Casati and Varzi [4][5],

[1] Urbanización Montepríncipe, s/n; 28668 Boadilla del Monte, Madrid, Spain; E-mail: mfernandez.eps@ceu.es
[2] Campus de Montegancedo, s/n; 28660 Boadilla del Monte, Madrid, Spain; E-mail: {asun@fi.upm.es, mcsuarez@fi.upm.es }

Smith [6][7]and Lambrix [8], among others, made relevant contributions to this field. Lambrix is well known for his contribution to the application of mereology to information systems.

In this paper we have adopted the definition of ontology proposed by Studer et al. ([9], page 185, based on Gruber's definition [10]), "*an ontology is a formal, explicit specification of a shared conceptualization*". When explaining such a definition, the authors interpret the word *shared* as agreed, and the word *formal* as machine interpretable. Therefore, if this definition holds, we can say that a mereology ontology is an agreed mereology implemented in a machine-interpretable language.

Recently, researchers like Gangemi et al. [11] and Massolo et al. [12] have proposed the use of mereology in ontology construction. This idea is embodied in the work of Borst [13], who built and used a mereology ontology in an engineering application for modeling, simulating and designing physical systems. In this article we intend to describe our experience in reusing ontologies that implement mereology ontologies in an EU Project. The main difference between the work shown in this section and Borst's is that we analyze, reuse and customize a mereology ontology already built, and that we do not develop a mereology ontology from scratch.

In Section 1, following Varzi's study, we present the different formalizations that the notion part-whole has undergone. In Section 2, we provide a list of the different ontologies that implement mereological definitions and then, we analyze them according to the formalization variants presented in section 1. In Section 3, we show a case in which these ontologies have been reused for building a Pharmaceutical Products ontology. Finally, in Section 4 we present the conclusions and future lines of research.

The Pharmaceutical Product ontology (PPO) will be used as a bridge between proprietary systems for managing financial and product knowledge interoperability in pharmaceutical laboratories, companies and distributors in Spain [14]. The composition of drugs, the interaction between them, etc., require the formalization of the *part of* relation. Consequently, it seems reasonable to consider the reuse of a mereology ontology.

1. Mereologies

A mereology is a formal theory of parts and associated concepts [13][15]. We have said '*a* mereology' instead of '*the* mereology' because different assumptions can be taken into account in the formalization of the part-whole relationship. Therefore, different mereologies can be proposed.

***Theory* M**

Part-of, otherwise *subclass-of*, is a relation between individuals. Most of the authors agree on the following core of axioms (named with an A) and definitions (named with a D) [5]:

A1) *Reflexivity*. Every object of the universe of discourse is part of itself. For instance, the European Union (EU) is part of the EU.

A2) *Antisymmetry*. If an object x is part of y, and y is part of x, then x and y are the same object. For instance, if the territory T_1 is part of the territory T_2, then the only way for T_2 to be part of T_1 is by being T_1 and T_2 the same territory.

A3) *Transitivity*. If x is part of y, and y is part of z, then x is part of z. For instance, Madrid is part of Spain, and Spain is part of EU, therefore, Madrid is part of EU.

A number of additional mereological predicates can be then introduced by definition. For example:

D1) *Proper part*. A proper part is a part that is other than the individual itself. For example, Spain is a proper part of EU, since Spain is part of EU and they are different entities.

D2) *Direct part*. X is a direct part of y if and only if x is a proper part of y and there is no part between x and y[3]. For example, Italy is a direct part of EU, but Madrid is not, since Madrid is part of EU through Spain.

D3) *Overlap*. The relation *overlap* is defined as a sharing part. That is, x and y overlap if and only if there is a z such that z is part of x and part of y. For instance, Spain and Africa overlap, since Spain has territories in Africa (Canaries, Ceuta, Melilla, etc.).

D4) *Underlap*. The relation *underlap* is defined as a sharing whole. That is, x and y underlap if and only if there is a z such that x and y are parts of z. For example, Portugal, Spain, France and Italy underlap because they share a common whole: EU.

D5) *Disjoint*. The *disjoint* relation is the logical negation of *overlaps*. For example, EU and USA territories are disjoint.

Theory M may be viewed as the embodiment of the common core of any mereological theory. A.1-A.3 should be extended to build a mereology.

***Minimal Mereology* (MM)**

A way to extend M is to assume the following decomposition principle [5]:

A4) *Weak supplementation principle*. Every object x with a proper part y has another part z that is disjoint of y. The domain of territories fulfils this principle. For example, given that Spain is a proper part of the European Union (EU), then EU has other parts that are disjoint of Spain: Portugal, France, Italy, etc.

Most of the authors maintain that A.4 should be incorporated to M as a further fundamental principle of the meaning of *part-of*. Other authors provide scenarios that could be counterexamples of this principle (see [4]). However, it has not been

[3] http://hcs.science.uva.nl/projects/NewKACTUS/library/lib/mereology.html

demonstrated yet that such supposed counterexamples have implications in computer applications.

Extensional Mereology **(EM)**

Another stronger way to express decomposition is the following:

A5) *Strong supplementation*. If y is not part of x, then there is a part of y that does not overlap with x. For example, given that Spain is not part of Africa, there is a part of Spain (e.g. Madrid) that is not part of Africa.

A.5 implies A.4.
This theory is called 'extensional' because a theorem that can be demonstrated is

T1) for all x's and y's, such that x has proper parts or y has proper parts, x and y are identical if and only if x and y have the same proper parts, that is, for all z's , z is proper part of x if and only if z is part of y. For example, the territory of the Community of Madrid is the same as that of the province of Madrid because both territories are composed of the same proper parts, that is, by the same municipalities.

Closure Mereology **(CM)**

Another way of extending *M* is by composition [4]

A6) *Sum principle*. If x and y underlap, then there is a z such that, for all w's, w overlaps z if and only if w overlaps x or w overlaps y. That is, if two objects underlap, then it may be assumed that there is a smallest object of which they are part (an object that exactly and completely exhausts both).

According to (A6), there is an object made up exactly of Madrid and Barcelona.

A7) *Product principle*. If x overlaps y, then there is a z such that for all w's, w is part of z if and only if w is part of x and w is part of y. That is, if two objects overlap, then it may be assumed that there is the largest object that is part of both (the common part at their junction). For example, Spain and Africa overlap, and it may be assumed that there is the largest object overlapped by both: Canaries, Ceuta, Melilla, etc.

The assumption of (A6) and (A7) is controversial. In fact, it is not obvious that the overlap of Spain and Africa makes an entity.

Closure Extensional Mereology **(CEM)**

The result of adding these axioms to *MM* or *EM* yields corresponding Minimal or Extensional Closure Mereologies, that is, *CMM* and *CEM*, respectively. In the presence of (A4), (A7) implies (A5). Consequently, *CMM* and *CEM* are the same theory [4].

The entities whose conditional existence is asserted by (A6) and (A7) must be unique in the presence of extensionality. Thus, *CEM* supports the following definitions:

D6) *Binary sum*. *X* + *y* is the *z* that fulfils that for all *w*'s , *w* overlaps *z* if and only if *w* overlaps *x* or *w* overlaps *y*. That is, *x* + *y* is the smallest object of which *x* and *y* are part.

D7) *Binary product*. *X* . *y* is the *z* that fulfils that for all *w's*, *w* is part of *z* if and only if *w* is part of *x* and *w* is part of *y*. That is, *X* . *y* is the largest object that is part of *x* and *y*.

General (classical) Mereology **(GM)**

Another way of extending *M* is through the following axiom schema:

A8) *Unrestricted fusion principle*. For every satisfied property or condition φ, there is a *z* such that for all y's, *y* overlaps *z* if and only if there is an *x* such that it satisfies φ and overlaps *y*. That is, there is an entity consisting of all those things that satisfy φ. For example, suppose that φ means "country with more than 10 million inhabitants", then there is an object that consists of all the countries with more than 10 million inhabitants.

If (A5) is satisfied, then at most one entity can satisfy the consequent of (A8). Therefore, the operation of general sum (σ) can be defined as follows:

D8) *General sum*. The general sum of all *x's* satisfying φ is that *z* that for every *y,* it overlaps *z* if and only if there is an *x* such that it satisfies φ and overlaps *y*. That is, the sum of ϕs is the entity that consists of all entities that satisfy ϕ.

General Extensional Mereology **(GEM)**

The extensions of *MM* and *EM*, which yield the same extensional strengthening of *GM* [4], is the theory of General Extensional Mereology, or *GEM*, since (A8) implies (A7) and (A7)+(A4) imply (A5) ([3], page 31). It is also clear that *GM* is an extension of *CM* and *GEM* is an extension of *CEM*, since (A6) can be deduced from (A8).

Atomistic Mereology

In an atomistic mereological theory, every element is made up of elements that are building blocks or atoms. To describe such a theory, the following definition can be provided:

D9) *Atom*. It is an element that does not have proper parts.

The atomistic axiom can be formulated in the following way:

A9) *Atomicity*. Every object has at least a part that is an atom. For example, the administrative division of territories follows this axiom, since there are simple divisions that are not divided.

Figure 1 shows a diagram with all the theories presented in this section.

A mereology *X* (e.g. *GEM*) extended with the atomicity axiom is known as ***AX*** (e.g. *AGEM*).

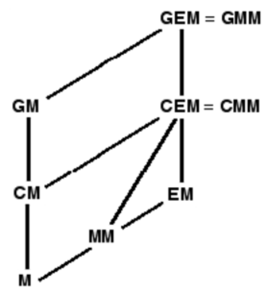


Figure 1. Hasse's diagram of mereological theories (from weaker to stronger, going uphill) [4].

2. Mereology Ontologies

Using a general purpose search engine (Google) and a specific one (Swoogle), we have found a series of mereology ontologies. Some of them have been rejected because they lack documentation; therefore, the final list of ontologies that we have studied is this

- *KACTUS* [16] ontology library, implemented in CML, it is maintained by the University of Amsterdam. Such a library contains the Mereological Ontology (MO), which is an adapted version of Borst's proposals [13](see foot note 3).
- *DOLCE* is one of the ontologies developed within the WonderWeb European project[4] [12]. It is available in KIF and OWL.
- The *Standard Upper Ontology*[5] (SUO) is the result of a joint effort to create a large, general-purpose, formal ontology [17]. It is promoted by the IEEE Standard Upper Ontology working group, and its development began in May 2000. This ontology is implemented in KIF and Protégé format; on the other hand, SUO formally describes mereology and topology terms. The general predicates in this section of the ontology are adapted from Barry Smith, Borgo et al.'s work, and from Casati and Varzi's mereologies.
- *The ontology based on Barry Smith and other authors' work* [6][7] in KIF, which is referred in the SUO web page. It represents various mereological definitions and axioms concerning boundaries and objects[6].

[4] http://wonderweb.semanticweb.org/
[5] http://suo.ieee.org/SUO/ontologies/Guarino.txt
[6] http://suo.ieee.org/

- The mereology based on the work of *Borgo et al.,* which is another ontology referred in SUO web page. These authors describe a set of definitions and axioms concerning mereology in [18]. Such ontology is currently implemented in KIF[7]. The ontology formalizes a CEM mereology (except the product principle).
- *Casati and Varzi's* [19] mereology can be also found implemented in KIF and is also referred in the SUO web page.

The features of these ontologies with regard to the theories previously presented are shown in table 1.

3. Use Case: a Pharmaceutical Product Ontology Built within the NeOn Project

NeOn[8] is a project co-funded by the European Commission's Sixth Framework Programme. Its aim is to advance the state of the art of the use of ontologies for *large-scale* semantic applications in the distributed organizations. Particularly, it aims to improve the capability of handling multiple *networked ontologies* that exist in a particular *context*, that are created *collaboratively*, and that might be highly dynamic and constantly *evolving*.

The Pharmaceutical Product ontology (PPO) will be part of the supportive collaboration of the pharmaceutical industry, concerned with the infrastructure and its APIs to bridge the currently used proprietary systems for managing financial and product knowledge interoperability in the networks/clusters of pharmaceutical laboratories, companies and distributors in Spain [14]. In this section, we will focus just in four (out of 61) competency questions (CQs) that allow us to explain our idea as best as possible (see table 2).

The activities that we have carried out to identify the mereology terms, axioms and definitions required in the PPO are the following:

I) *Ontology search*. The activity implies finding candidate ontologies or ontology modules to be reused. To perform this activity, we have started from the analysis of the PPO competency questions (CQs):

  I.a) *Analysis of the PPO CQs*. The steps carried out are the following:

    I.a.i) *To obtain the concepts and their relations that allow us to represent the terms appearing in the CQs*. For our case, the concepts are *chemical substance* and *drug*, and the relations are *has component* and *has interaction with* (both relations are found between chemical substances), and *has main substance* and *has active ingredient* (both relations are found between drugs and chemical substances). Besides, *drug* is subclass of *chemical substance.* Drug-drug interaction does not necessarily require the physical interaction between their ingredients. While some drugs might inactivate other drugs by bonding with them, others simply compete for the same receptors without interacting physically.

[7] http://suo.ieee.org/SUO/ontologies/Guarino.txt
[8] http://www.neon-project.org

Table 1. Features of ontologies that implement mereotopology theories. The characteristics that appear shaded are those required for the Pharmaceutical Product Ontology.

Theory	**Principles and definitions**	***KACTUS***	***DOLCE***	***SUO***	***Smith et al.***	***Borgo et al.***	***Casati and Varzi***
M	A.1) Reflexivity	No	No	**Yes**	**Yes**	**Yes**	No
	A.2) Antisymmetry	**Yes**	No	**Yes**	**Yes**	**Yes**	No
	A.3) Transitivity	**Yes**	No	**Yes**	**Yes**	**Yes**	No
	D.1) Proper part	**Yes**	**Yes**	**Yes**	No	**Yes**	**Yes**
	D.2) Direct part	**Yes**	No	No	No	No	No
	D.3) Overlap	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**
	D.4) Underlap	No	No	No	No	No	No
	D.5) Disjoint	**Yes**	No	No	No	No	No
MM = M + (P4)	A.4) Weak supplementation	**Yes**	No	No	**Yes**	**Inferred**	No
EM = M + (A5) (Let's note that (A5) implies (A4)	A.5) Strong supplementation	No	No	No	**Yes**	**Yes**	No
CM = M + (A6) + (A7)	A.6) Sum principle	No	No	No	**Yes**	**Yes**	No
	A.7) Product principle	No	No	No	**Yes**	No	No
CEM = CM + (A5)	D.6) Binary sum	No	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**
	D.7) Binary product	No	No	**Yes**	**Yes**	**Yes**	**Yes**
GM = M + (A8)	A.8) Unrestricted fusion principle	No	No	No	**Yes**	No	No
GEM = GM + (A5)	D.8) General sum	No	**Yes**	No	**Yes**	No	No
AX = (A9) + a mereology X	D.9) Atom	No	**Yes**	No	No	No	No
	A.9) Atomicity	No	No	No	No	No	No

Table 2. Competency question analysis for mereology ontology reuse

Identifier	Competency question	Competency question using the vocabulary of mereology	Extracted terms
CQ1	*What is the composition of the drug?*	Which are the parts of the drug?	- *part of*
CQ2	*Which is the main active ingredient (molecule) of the drug)?*	(It does not directly require mereotopology)	- *active ingredient* This term requires the definition of: *part of*
CQ3	*Which is the main substance of the composition?*	(It does not directly require mereotopology)	- *main substance* This term requires the definition of: *part of*
CQ4	*Does the drug have interaction with another drug?*	Are there parts of the drug that interact with parts of another drug?	- *part of*

I.a.ii) *To apply a pattern of reuse of mereologies* (inspired by [20][21]), which answers the question *¿what relations are candidate to be renamed so that mereology definitions and axioms can be reused?* The pattern involves the use of the following rules:

- *Rule 1*. If the relation establishes a (partial) order, then this relation (or its inverse) is candidate to be a kind of *part-of*. An example that adheres to this rule is *component of*, since a substance x can be a component of y, y can be a component of z, etc, in such a way that an order $x < y < z$ is established. Note that the engineer could have decided to model the relation *has component* instead of *component of*.
- *Rule 2*. If we can find a super-relation that establishes an order, then this relation is candidate to be modeled in terms of a mereology, even though it is not itself a type of *part-of*. For example, that x has as a main substance y implies that y is part of x.

I.a.iii) *To try to reformulate the competency questions in terms of mereology by following the results of the pattern rules*. Table 2 shows how these CQs require mereology terms in PPO. The competency question, *What is the drug main active ingredient (molecule)?,* has led to the formalization of the term *active ingredient* by means of the following SWRL rule:

R1) hasActiveIngredient(?x, ?y) → part-of(?y, ?x)

The competency question, *Which is the main substance of the composition?* has led to the formalization of the term *main substance* by means of the following SWRL rule:

R2) hasMainSubstance(?x, ?y) → part-of(?y, ?x)

The competency question, *Does the drug have interaction with another drug?* has led to the following SWRL rule concerning the term *interaction with*:

R3) chemicalSubstance(?chs1) ∧ chemicalSubstance(?chs2) ∧
part-of(?chs11, ?chs1) ∧ part-of(?chs21, ?chs2) ∧
hasInteractionWith(?chs11, ?chs21) →
hasInteractionWith(?chs1, ?chs2)

that is, if a part of a chemical substance *chs1* interacts with another part of a chemical substance *chs2*, then *chs1* interacts with *chs2*. In other words, the interaction is inherited from the parts to the whole [22].

The analysis of other competency questions produces rules that link chemical substance terms to mereology terms.

I.b) *Identification of the features of the mereology ontology to be reused*. Section 1 and, particularly, table 1 permits us to identify which features the reused mereology ontology should have.

Some properties (e.g. transitivity) were not clearly determined by competence questions. This fact indicates that the meaning of the CQs was not completely clear. That is, the study of the axioms and definitions shown in table 1 has helped us to identify ambiguities and, as will be seen in the further paragraphs, to precise the meaning of the CQs.

For the PPO case, the following formalization has been necessary:

A.1) *Reflexivity*. To ensure the right meaning of the ontology terms. Thus, for example, if *part of* is not reflexive, then rule (R3) may not work for a query where *chs1* is identical to *chs11* or *ch2* is identical to *chs21*.

A.2) *Antisymmetry*. To help the user to check constraints.

A.3) *Transitivity*. To be modeled if the different levels of the structure of components are provided. For example, Frenadol® is composed of paracetamol, dextrometorphan, and clorfenamine. Paracetamol, in its turn, is composed of an alcohol, an amino group and a carbonyl group. The alcohol is composed of oxygen and hydrogen, etc. Given that the inclusion of the transitivity axiom is low cost, we have opted to include all the components in the answer of CQs.

D.1) *Proper part*. The formalization of this term eases the interpretation of the competency questions. Thus, the very substance should not be a result of CQ1, but a result of CQ4 instead, since the very substance can interact with a part of another substance.

D.2) *Direct part*. This term allows answering CQ1 just in a level. Therefore, CQ1 has been split into two competency questions: (CQ1') *What is the drug composition? (considering only a level)* and (CQ1'') idem *(considering all the components)*.

A.4) *Weak supplementation principle*. This axiom helps the user to check constraints.

D.3) *Overlap*. It is necessary to formalize (A4).

D.4) *Underlap*. It is not necessary for the PPO at the moment.

D.5) *Disjoint*. It is necessary to formalize (A4).

A.5) *Strong supplementation principle* is not true if the bounds between atoms are not taken into account. We should remember that (A5) implies that two not atomic entities are identical if and only if they have the same parts. However, isomers are not ruled out in the Pharmaceutical Product ontology. An isomer is a chemical compound with the same number and kind of atoms as another but different structural arrangement. If the structure of drugs is required, then a topology ontology is needed.

D.6 and more) *Sums and product*. We do not find necessary to define sums and products.

D.10) Atom. As will be seen, the term *atom* will be used to represent the weak supplementation principle.

A.9) *Atomicity*. We do not think that atomicity is needed in the PPO.

The aforementioned features are shadowed in table 1.

II) *Selection of a mereology ontology*. This activity involves choosing the most suitable ontologies or ontology modules among those available in an ontology repository or library, for a concrete domain of interest and associated tasks. No mereology ontology completely fulfills all criteria, but KACTUS MO is the ontology that best fits the required features. This ontology has the added advantage of having been built to be easily reused in knowledge based systems.

Note however that the rest of the ontologies cover more characteristics that are not considered in the current CQs than KACTUS MO does. In this step, therefore, we have decided to reuse more the mereology ontology within the current version of the ontology, and to reuse it less in further versions.

III) *Customization of the chosen mereology ontology*. This activity involves adapting an ontology to a specific user's needs. We have carried out the following subactivities: (1) pruning the reused ontology according to the features really necessary; (2) enriching the ontology (e.g. with the *part of* reflexivity axiom); (3) translating from CML into OWL + SWRL; and (4) evaluating the ontology obtained. The result of this activity is the tree shown in Figure 2 (the concepts in italic have been added during activity (IV), and a series of OWL definitions and SWRL rules. Some of them are the following:

D.1') MereologicalIndividual $\subseteq$ owl:thing

D.2') mereologicalIndividualNotFWSPrinciple[9] $\equiv$
mereologicalIndividual $\cap$ not mereologicalIndividualFWSPrinciple

D.3') atom $\equiv$ mereologicalIndividualFWSPrinciple $\cap$
(= 0 properPart mereologicalIndividual)

Asymmetry of *proper part of*:

R.4)properPart(?x, ?y) $\wedge$ properPart(?z, ?x) $\rightarrow$ differentFrom(?y, ?z)

Transitivity of *proper part of*:

R.5)properPart(?x, ?y) $\wedge$ properPart(?y, ?z) $\rightarrow$ properPart(?x, ?z)

Weak supplementation principle:

R.6) properPart(?x, ?y) $\wedge$ properPart(?z, ?y) $\wedge$ isDisjointFrom(?z, ?x) $\rightarrow$
merologicalIndividualFWSPrinciple(?y)

This principle is completed with the *atom* definition (D.3').

IV) *Integration of the chosen mereology ontology in the PPO*. The product of the activity (III) has been included in the PPO. Besides the bridge rules (R1), (R2) and (R3), the following definitions have been incorporated to obtain the ontology to be newly evaluated:

D.4) chemicalSubstance $\subseteq$ mereologicalIndividualFWSPrinciple

D.5) drug $\equiv$
chemicalSubstance $\cap$
($\geq$ 1 hasActiveIngredient chemicalSubstance) $\cap$
($\geq$ 1 hasMainSubstance chemicalSubstance)

[9] FWS means 'fulfilling weak supplementation principle'.

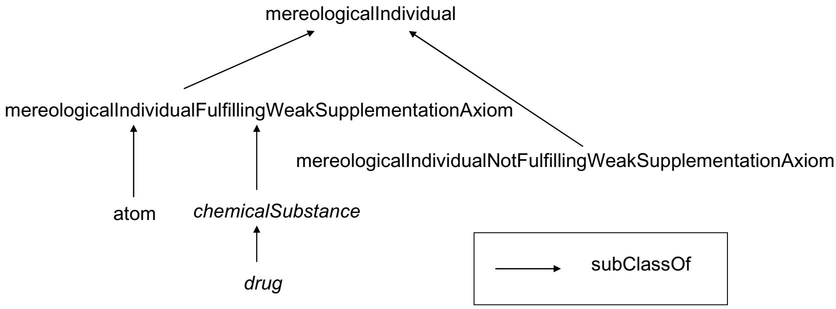


Figure 2. Resulting mereology ontology concept tree integrated in the Pharmaceutical Product ontology through the concepts *chemical substance* and *drug*.

4. Conclusions and Future Directions

In this paper, we have presented the study of different mereologies following Varzi's work [5], and have analyzed a set of mereology ontologies according to the principles and definitions previously identified. Then, we have specified the mereology modeling necessities in the Pharmaceutical Product ontology in the NeOn European project. We have observed that the formalization of the reflexivity, antisymmetry and transitivity axioms, together with the weak supplementation principle and the *proper part of* and *direct part of* definitions are useful for the development of the ontology. Moreover, in the future, it is possible to incorporate axioms that imply extensionality, if a detailed structure of drugs is modeled in the ontology. Future investigation lines should include the analysis of other types of *part-of* different to the functional one and the reuse of patterns for other types of common ontologies (e.g. topologies).

After analyzing the candidate mereology ontologies, we select KACTUS MO to model the part-whole relationship in the NeOn Pharmaceutical Product.

One of the results of this work is the procedure to reuse mereology ontologies, which refines the one presented in [23] for time ontologies. Such a procedure consists in: (I) searching the candidate mereology ontologies; (II) selecting the mereology ontology that best matches the modeling necessities; (III) adapting the chosen mereology ontology; and (IV) integrating the mereology ontology in the host ontology. Activity (I) requires analyzing the CQs of the ontology where the mereology ontology will be reused, and identifying the features required by the mereology ontology. We provide a table explaining the features of the most well-known mereology ontologies. The table also indicates which of these features have been useful in the real case presented in the paper. It must be added that the analysis of CQs may imply their redefinition. Another result of our work is the OWL+SWRL mereology ontology.

Acknowledgements

This work has been partially supported by the IST European Project NeOn (IST-2005-027595). We are grateful to the reviewers for their interesting comments, and to

Rosario Plaza for her help in checking and reviewing the grammar and spelling of the paper and improving clarity and style.

References

[1] Lesniewski S (1916) *Podstawy ogolnej teoryi mnogosci*. I, Moscow: Prace Polskiego Kola Naukowegow Moskwie (English translation due to Barnett DI, 'Foundations of the General Theory of Sets. I', in Lesniewski S, Collected Works, Surma SJ et al. (eds), Dordrecht, Boston, y Londres: Kluwer Academic Publishers, 1992, Volume 1, pp. 129 -173)

[2] Leonard HS, Goodman N (1940) *The Calculus of Individuals and Its Uses*. Journal of Symbolic Logic 5:45-55

[3] Simons P (1987) *Parts: A Study in Ontology*. Clarendon Press, Oxford, United Kingdom

[4] Varzi A (2003) *Mereology*. In Zalta EN, Nodelman U, Allen C (eds) Stanford Encyclopedia of Philosophy, Stanford: CSLI (on line publication) (*http://plato.stanford.edu/entries/mereology/*) (last access: 6PthP February 2008)

[5] Varzi A (2007) *Spatial Reasoning and Ontology: Parts, Wholes, and Locations*. In Aiello M, Pratt-Hartmann I, van Benthem J (eds) Springer-Verlag, pp 945-1038

[6] Smith B (1996) *Mereotopology: A Theory of Parts and Boundaries*. Data and Knowledge Engineering, 20, 287-303

[7] Smith B (2001) *Fiat objects*. Topoi, 20(2):131-148

[8] Lambrix P (2000) *Part-Whole Reasoning in an Object-Centered Framework*. Lecture Notes in Artificial Intelligence LNAI 1771, Springer Verlag, Berlin, Germany

[9] Studer R, Benjamins VR, Fensel D (1998) *Knowledge Engineering: Principles and Methods*. IEEE Transactions on Data and Knowledge Engineering 25(1-2):161–197

[10] Gruber TR (1993) *A translation approach to portable ontology specification*. Knowledge Acquisition 5(2):199–220

[11] Gangemi A, Guarino N, Massolo C, Oltramari A (2001) *Understanding top-level ontological distinctions*. In: Gómez-Pérez A, Grüninger M, Stuckenschmidt H, Uschold M (eds) IJCAI'01 Workshop on Ontologies and Information Sharing. Seattle, Washington, pp 26-33. CEUR Workshop Proceedings 47:26-33. Amsterdam, The Netherlands (*http:/ceur-ws.org/Vol-47/*)

[12] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A, Schneider L (2003) *The WonderWeb Library of Foundational Ontologies*, Preliminary Report, WonderWeb deliverable D1.7 (HTU*http://www.loa-cnr.it/Papers/DOLCE2.1-FOL.pdf*UTH)

[13] Borst WN (1997) *Construction of Engineering Ontologies*. Centre for Telematica and Information Technology, University of Tweenty. Enschede, The Netherlands

[14] Gómez-Pérez JM, Pariente T, Buil-Aranda C, Herrero G, Baena A (2007). *NeOn Deliverable D8.3.1. Ontologies for pharmaceutical case studies*.

[15] Schneider L (2004) *How to Build a Foundational Ontology. The Object-Centered High-level Reference Ontology OCHRE*. Saarland University forthcoming publication. (*http://www.ifomis.uni-saarland.de/Research/Publications/forthcoming/ki2003epaper.pdf*)

[16] KACTUS (1996) *The KACTUS Booklet version 1.0*. Esprit Project 8145 KACTUS. *http://www.swi.psy.uva.nl/projects/NewKACTUS/Reports.html*

[17] Pease RA, Niles I (2002) *IEEE Standard Upper Ontology: A Progress Report*. The Knowledge Engineering Review 17(1):65–70

[18] Borgo S, Guarino N, Masolo C (1996) *A Pointless Theory of Space Based on Strong Connection and Congruence*. In: Carlucci-Aiello L, Doyle J (eds) 5PthP International Conference on Principles of Knowledge Representation and Reasoning (KR'96). Morgan Kaufmann Publishers, San Francisco, California, pp 220–229

[19] Casati R, Varzi A (1995) *Holes and other superficialities*. MIT Press, Cambridge, Massachusetts

[20] Clark P, Thompson J, Porter BW (2004) Knowledge Patterns. In: Staab S, Studer R (eds) *Handbook on Ontologies*. International Handbooks on Information Systems. Springer-Verlag, pp 191-208

[21] Presutti V et al. (2008) A Library of Ontology Design Patterns: reusable solutions for collaborative design of networked ontologies. Deliverable 2.5.1 IST European Project NeOn (IST-2005-027595)

[22] Bernauer J (1996) *Analysis of part-whole relation and subsumption in the medical domain. Data & Knowledge Engineering* 20(3):405-415

[23] Fernández-López M, Gómez-Pérez A (2004) *Searching for a Time Ontology for Semantic Web Applications*. In: Varzi A, Vieu L (eds) Formal Ontology in Information Systems, Torino, Italy IOS Press, Amsterdam

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-195

An Upper-Level Ontology for Chemistry

Colin BATCHELOR [a,1]
[a] *Royal Society of Chemistry, Cambridge, UK*

Abstract. Chemical entities are the foundation of biochemistry and biology, but until now there have been few coherent attempts to produce a top-level ontology for chemistry to connect ontological descriptions of reality at the molecular level, such as ChEBI, with upper-level ontologies such as BFO, or indeed familiar laboratory-scale concepts such as mixtures. We work out relationships between chemical types that are compatible with the OBO Relation Ontology, describe macroscopic chemical systems in terms of grains and collectives, and propose a top-level ontology for chemically-relevant continuants and discuss it in relation to BFO and BioTop.

Keywords. upper-level ontology, chemistry, OBO Relation Ontology

1. Introduction

Chemoinformatics, the study of information systems that handle chemical entities, has spawned a multimillion-pound industry supplying services to the pharmaceutical and other industries. The field has been slow to adopt the methods of formal ontology. Exceptions include work by Dumontier and coworkers [1,2], who use the connectivity of atoms to infer classifications for molecules, work at NIST [3,4] on a chemical taxonomy of HIV inhibitors written in OWL, and ChEBI (Chemical Entities of Biological Interest),[5] an ontology containing around 15 000 chemical entities. Most if not all biomedical ontologies refer explicitly or implicitly to chemical entities, so the OBO Foundry project [6], a suite of interoperable biomedical ontologies, needs a soundly-constructed ontology of chemical entities to build on. However progress on aligning ChEBI with other biomedical ontologies such as the Gene Ontology (GO)[7] is slow [8,9]. One reason is that it is far from clear from the definitions in ChEBI whether the terms refer to molecules in the universe, to names of molecules, or to molecular structures.

The case for an upper-level ontology for chemistry is therefore twofold; first, defining what the objects referred to in an ontology actually are allows the curators to set its scope and determine the genera for high-level genus–differentia definitions; second, it allows the types in the ontology to be reused safely by other ontologies with overlapping scope. Schulz *et al.* [10] have developed an upper-level ontology for molecular biology by taking the GENIA ontology [11] as a start point, and performing an ontological analysis of it which is based around choosing a set of foundational relations and developing formal properties and classifying entities based on those.

[1]Corresponding Author: Royal Society of Chemistry, Thomas Graham House, Milton Road, Cambridge, UK CB4 0WF. batchelorc@rsc.org

In this paper we present an upper-level ontology for chemistry based on an analysis of the foundational and disguised foundational relations in ChEBI. The paper is structured as follows: In section 2 we analyse ChEBI and discover . In section 3 we rewrite the relations so that they are compatible with the OBO Relation Ontology [12], taking into account Rector *et al.*'s [13] distinction between determinate and granular parthood, and determine the types that belong in this upper-level ontology. In section 4 we align the terms with a top-level ontology, in this case BFO [14]. Finally in Section 5 we discuss related work and assess the implications for ChEBI, BioTOP [10] and the InChI chemical identifier [15].

We present types as follows: existing ChEBI entities will be in *italics*, proposed top-level types will be in **bold** and relationships will be in `monospaced text`. CamelCase names are taken from BioTop. BFO types will be in CamelCase preceded by "snap:".

2. Examination of ChEBI

We shall take a realist perspective[16] and think in terms of ChEBI containing types that are actually or potentially instantiated in the universe. So the instances we are seeking to describe in this paper with a chemical ontology include for example, single atoms in the interstellar medium, a water molecule interposed between a particular pair of nucleotides in a specific RNA motif, an individual cluster of atoms under a scanning electron microscope tip, a sodium ion in a glass of brine and so forth.

This clashes with the use of the word "instance" in the definition of the `is_a` relationship given by ChEBI:

> Implies that 'Entity A' is an instance of 'Entity B'. [...] *chloroform* (CHEBI:23143) is an instance of the class of *chloromethanes* (CHEBI:23148), which is itself an instance of the class of *chloroalkanes* (CHEBI:23143), and so forth.

Elsewhere ChEBI explicitly describes two superclasses: *organic functional classes* (CHEBI:33244) and *natural product classes* (CHEBI:33243), which reflect two parallel systems for classifying organic compounds, one according to the biological context where they were first discovered, the other according to their chemical composition. They will have to be renamed and have definitions written for them if ChEBI is to be made compatible with other realist ontologies. It is possible to imagine a shelf which has on it bottles containing natural products, but certainly not bottles that contain natural product classes.

For this section we shall determine what top-level classes are entailed by a chemical reading of ChEBI and its relationships and definitions, and define them properly, in the light of BFO, in a later section. We refer to the OBO format version of ChEBI Release 43 throughout.

2.1. Explicit Parthood

ChEBI has one explicit parthood relation, `is_part_of`, which is defined as follows:

> Used to indicate relationship between part and whole. [...] *tetracyanonickelate(2−)* (CHEBI:30025) is part of *potassium tetracyanonickelate(2−)* (CHEBI:30071)

This example is not compatible with the part_of relationship in the OBO Relation Ontology, because that would entail that all tetracyanonickelate (2−) ions were part of some potassium tetracyanonickelate (2−) species, but of course it is quite possible to have tetracyanonickelate ions that are free in solution or part of some other complex. Indeed, we will see later on that compounds similar to "potassium tetracyanonickelate" may have different ontological interpretations depending on context.

Inspection of ChEBI leads to at least five interpretations of is_part_of:

1. is necessarily part of a molecular entity: *carbonyl group* (CHEBI:23019) and *carbonyl compounds* (CHEBI:36586)
2. is possibly part of a molecular entity: *electron* (CHEBI:10545) is part of *muonium* (CHEBI:30213)
3. is possibly part of a mixture: *kanamycin A* (CHEBI:17630) is part of *kanamycin* (CHEBI:6104)
4. is possibly part of a salt: *lead(2+)* (CHEBI:30179) is part of *lead diacetate* (CHEBI:31767)
5. is connected in the ChEBI database structure to another entry: *biological role* (CHEBI:24432) is part of *ChEBI ontology* (CHEBI:23091)

This contradicts the principle of univocity,[17] that each relationship should only have one interpretation. This is also possible because the domain and range of the is_part_of relationship have not been specified in advance. We will defer in-depth discussion of these to a later section, but point out that the fifth interpretation can in no sense be related to instances in reality and therefore we will not consider it further.

2.2. *Disguised Parthood*

There are also three 'disguised' parthood relations that prove straightforwardly to be incompatible with a realist approach to ontology development—is_substituent_group_from, has_functional_parent and has_parent_hydride. First, is_substituent_group_from can be interpreted as:

1. is possibly part of a molecular entity:
   methyl group (CHEBI:32875) is_substituent_group_from *methane* (CHEBI:16183)

This is difficult to interpret in terms of the all–some structure of RO relations. Does it make sense to say that all methyl groups are derived formally from some other methane molecule? For compatibility with RO it is best to drop this relation altogether and think of the relation between methyl groups and methane molecules in terms of parthood as we shall see later.

Second, has_functional_parent from a realist perspective looks like this:

1. shares a substructure with a molecular entity: D-*glucoronate 1-phosphate* (CHEBI:28547) has_functional_parent D-*glucuronate* (CHEBI:15748)

The ChEBI definition is:

> Used to denote the relationship between two molecular entities (or classes of entities), one of which possesses one or more characteristic groups from which the other can be derived by functional modification.

Again, while it is true to say that all molecules of D-*glucoronate 1-phosphate* could in principle have been formed by phosphorylating some molecule of D-*glucoronate* it seems best again to drop this relationship.

Last, `has_parent_hydride` is a special case of `has_functional_parent`, though it does have a range if not a domain:

> Denotes the relationship between an entity and its parent hydride (defined by IUPAC as "an unbranched acyclic or cyclic structure or an acyclic/cyclic structure having a semisystematic or trivial name to which only hydrogen atoms are attached").

The definition of parent hydride here is in fact closest to that of CHEBI:33245, *organic fundamental parents*, but this type is problematic as it has been added as a guide to systematic nomenclature; the children of this type have little in common.

2.3. Subsumption Relations

The `is_a` relationship is used for a variety of purposes in ChEBI:

1. an amount of a compound has a biological role:
   tris (CHEBI:9754) is a *buffer* (CHEBI:35225)
2. an amount of a compound has an application:
   sodium dodecyl sulfate (CHEBI:8984) is a *detergent* (CHEBI:27780)
3. connecting a less-abstract class with a more-abstract class:
   propane (CHEBI:32879) is a *alkanes* (CHEBI:18310)
4. connecting macroscopic entities with atoms:
   metals (CHEBI:33521) is a *atoms* (CHEBI:33250).
5. connecting elements with atoms:
   main group elements (CHEBI:33318) is a *atoms* (CHEBI:33250)

The clearest problems with the first two cases are that the relationships would be better represented as `has_role` and `has_application`, and secondly that they can only be understood in terms of amounts of compounds, inconsistently with the rest of the ontology, which is mainly described in terms of individual molecules. It is samples of tris that act as buffers, rather than individual tris molecules, but every propane molecule is an alkane molecule. The third case is absolutely standard in ontologies and it is possible to imagine genus–differentia definitions for almost all of the structurally-based examples in ChEBI. However, ChEBI has not followed this route, preferring a multiple-genus approach. We shall consider the case of *polypodine B* (CHEBI:28485), which has no fewer than nine `is_a` parents, which we list in Table 1. The easiest parent to discuss is *phytoecdysteroids* (CHEBI:26118), which says something about the role the compound plays in the context of plant metabolism but is inessential to its structural description. The other parents have genus–differentia names but not definitions, so a *3β-hydroxy steroid* is a compound with the steroid skeleton and a hydroxy group bound to the skeleton at the position numbered 3 with a particular orientation (β) relative to the rest of the molecule. The definition of the steroid skeleton and the canonical numbering are given by Moss.[18]

This demonstrates that simply stating the determinate parts of a molecule is not enough to specify it unambiguously. We must also specify their relative locations. This requires a new parthood relationship which we will define in detail in the next section.

is_a parent	Specifies Role	Parent	Substituent	Location	Orientation
phytoecdysteroids (CHEBI:26118)	+	–	–	–	–
3β-hydroxy steroids (CHEBI:36836)	–	+	+	+	+
26-hydroxy steroids (CHEBI:36852)	–	+	+	+	–
20-hydroxy steroids (CHEBI:36854)	–	+	+	+	–
2β-hydroxy steroids (CHEBI:36859)	–	+	+	+	+
14α-hydroxy steroids (CHEBI:36861)	–	+	+	+	+
22-hydroxy steroids (CHEBI:36863)	–	+	+	+	–
6-oxo steroids (CHEBI:36883)	–	+	+	+	–
5β-hydroxy steroids (CHEBI:38195)	–	+	+	+	+

Table 1. ChEBI's genus–genus–genus–genus–genus–genus–genus–genus–genus description of polypodine B. Here "Location" means the location of the substituent on the parent skeleton and "Orientation" means the substituent's orientation relative to

The fourth and fifth cases are tricky to disentangle: *metals* and *main group elements* themselves are undefined within ChEBI, but the definition of *atom* reads:

An atom is the smallest particle still characterizing a chemical element.

This definition relies on the notion of a chemical element, which is defined in the Gold Book [19] as

1. A species of atoms; all atoms with the same number of protons in the atomic nucleus.
2. A pure chemical substance composed of atoms with the same number of protons in the atomic nucleus. Something this concept is called the elementary substance as distinct from the chemical element as defined under 1, but mostly the term chemical element is used for both concepts.

There is a straightforward fix for the oddity of elements being `is_a` children of atoms. If the descendants are renamed with names ending in "atom"—"main group atom", "s-block atom" and so forth, then the problem goes away. The existence, however, of the terms *metals* (CHEBI:33521) and *nonmetals* (CHEBI:25585) is pathological. Metallicity and nonmetallicity are not properties of atoms but rather of large assemblies of atoms, so would belong elsewhere in ChEBI if at all.

3. Relation Definitions

We need to distinguish between the parthood relations that hold both (a) on the level of molecules and a sample in a test-tube, and (b) only on the level of the sample in the test-tube. Grenon *et al.* [14] propose that each material application of BFO should be restricted to a given level of granularity. For this reason we seek to distinguish the relations that hold within a level of granularity from those that hold between levels.

Rector *et al.* [13] propose a distinction between granular and determinate parts which is useful to us here. A determinate part, for example a finger on a hand, is one in which the part is directly part of the whole and in which on being removed necessarily damages or diminishes the whole. A granular part is one, like a cell within a finger, where the grains are parts of the whole by virtue of being grains in a collective and in which removing one granular part does not necessarily damage or diminish the whole.

Another way of looking at the distinction is that determinate parts are often named (ring, middle, index, pinky in the case of fingers), or numbered as in the case of atoms in a molecule (the InChI identifier is written in terms of a canonical numbering for each atom) or amino acid residues in a peptide chain. No such naming or numbering happens to hairs on one's head or the molecules in a macroscopic sample.

The determinacy of the determinate parthood relation is stronger for chemical entities than is generally the case in biomedicine. While my hand can lose a finger while remaining identifiably a hand, a molecule of C_{60} cannot lose carbon atoms without ceasing to be C_{60}. The same applies to nearly all of the entities in ChEBI defined in terms of a specific formula. But the granular parthood relation is exactly the same—a lump of gold can lose gold atoms without ceasing to be gold. A lump of rocksalt can lose both Na^+ and Cl^- ions without ceasing to be rocksalt.

We need to distinguish salts (and their molten phases, ionic liquids) from mixtures. It is not clear from [13] whether salts count as mixtures. As a mixture is understood in chemistry, however, it is an amount of matter that has components, or in the language of [13], "ingredients", which are collectives in which the grains retain their identity. The ingredients of a mixture can in practice be recovered by mild techniques such as chromatography. Absent from the definition in [13] of a mixture being an amount of matter that has ingredients, is this notion of the grains retaining their identity. A macroscopic sample of a component of a mixture can generally exist independently but a macroscopic sample of lead (2+) would undergo a spectacular Coulomb explosion. Other examples of mixtures include alloys and solutions, gels, foams and soft matter in general.

If we examine our four surviving `is_part_of` cases from above for determinate and granular parts we find:

1. carbonyl compound molecules have as determinate parts carbonyl groups
2. muonium atoms have as determinate parts electrons (subatomic particles)

3a. kanamycin mixtures have as an ingredient kanamycin A

3b. kanamycin mixtures have as granular parts kanamycin A molecules

3c. kanamycin mixtures have as granular parts molecules of kanamycins, where *kanamycins* (CHEBI:24951) is a class that has subclasses kanamycins A, B and C, which are compounds, *kanamycin* (CHEBI:6104), which is the mixture we have already discussed, and the kanamycin derivatives *2″-nucleotidylkanamycins* (CHEBI:27557) and *acetylkanamycins* (CHEBI:22201).

4. lead diacetate salts have as granular parts lead(2+) ions.

We might also add, though these relations are not explicit in ChEBI:

5. carbon dioxide molecules have as determinate parts oxygen atoms
6. [bmim][PF_6] ionic liquids have as determinate parts $bmim^+$ ions.
7. pure gold has as granular parts gold atoms

So our macroscopic entities, salts, solutions, mixtures, and pure substances all have granular parts. In addition, mixtures and solutions have ingredients. Molecular entities, on the other hand, have only determinate parts. Note, however, that the chemical formula may be insufficient to tell you whether it refers to a molecular or macroscopic entity. Take NaCl, for example. In both liquid and solid NaCl, the sodium ions have no particular affinity for a single chloride ion, but in the solid state are surrounded by six

Relationship	Domain	Range
has_determinate_part	**molecule**	**molecular part**, **atom**
	molecular part	**molecular part**, **atom**
	atom	**subatomic particle**
has_substituent	**molecule**	**molecular part**, **atom**
	molecular part	**molecular part**, **atom**
has_granular_part	**pure substance**	**molecule**
	pure substance	**atom**
	salt	**molecule**
	salt	**atom**
	mixture	**molecule**
	mixture	**atom**
has_ingredient	**mixture**	**pure substance**, **salt**

Table 2. Domains and ranges of relationships for upper-level chemical classes.

nearest neighbours. There are no identifiable molecules and they can only be understood as macroscopic entities. Contrast this with a sample of gaseous NaCl, which consists of diatomic molecules. Loss of a sodium or chlorine atom from a given molecule would result in it no longer being an NaCl molecule. This suggests very strongly that ChEBI should be restricted to describing molecular-level entities.

The relationships in RO are all defined in the all–some direction, which makes it a trivial task to determine an RO-compatible mapping. Currently is_part_of means in about a dozen cases 'necessarily part of', and in all others 'possibly part of'. The solution is to reverse them all and replace them with the appropriate specialization of the RO has_part relation. All but one of our parthood relations have the same formal definition as the RO has_part relation, except that the domains and ranges, which are listed in Table 2, are different for each one.

The exception is has_substituent. The domain of has_substituent, C in the definition, is **molecules** and **molecular parts**. Allowable substituents S are **molecular parts** and **atoms**.

> *C has_substituent S at_position P* $\equiv$ for all c, s, t, if c instance_of C at t and s instance_of S at t then there is some s that is part_of c at position P.

It is useful here to introduce the distinction between an *open-world* name and a *closed-world* name. Many chemical names are polysemous and have both an open-world and a closed-world reading. This has been discussed at length from a natural-language-processing perspective by Corbett *et al.* [20]. Systematic and semi-systematic chemical names specify a genus and differentiae. Often these differentiae themselves have genus–differentia form, so the name has to be interpreted recursively. In the closed-world reading, which corresponds to an EXACT name in [20], any locations with unspecified substituents have a hydrogen atom attached to them. In the open-world reading, which corresponds to a CLASS name in [20], the substituents which are not mentioned in the name are left unspecified. Thus a type that has a name in the closed-world reading is an is_a descendant of the type that has the same name in the open-world reading.

In the above case a propane molecule (in the closed-world sense) is an alkane molecule which `has_part` carbon atom with a cardinality of exactly 3 and `has_part` hydrogen atom with a cardinality of exactly 8.

polypodine B would be the intersection of *steroids* (CHEBI:35341) with eight substituent–position pairs as follows:

> A steroid that has an oxo substituent at position 6, a beta-hydroxy substituent at position 2, a beta-hydroxy substituent at position 3, a beta-hydroxy substituent at position 5, an alpha-hydroxy substituent at position 14, a hydroxy substituent at position 20 and a hydroxy substituent at position 26.

If we assume that all positions left unspecified in that definition are occupied by hydrogen atoms, then we have the type for the molecule with the closed-world name "polypodine B". If we leave them unspecified, then we have a type for the molecules with the open-world name "polypodine B", which chemists might refer to as "the polypodine B", "a polypodine B", or, although these are not attested, "polypodine Bs" or "polypodines B".

4. Alignment with a Top-Level Ontology

There is no shortage of top-level ontologies to choose from, of which DOLCE [21], GFO [22] and BFO [14] are recent examples. We choose BFO for this alignment because it is the top-level ontology for the OBO ontologies of which ChEBI is one.

In the last section we identified top-level classes **molecule**, **molecular part**, **atom**, **subatomic particle**, **pure substance** (for want of a better name), **salt** and **mixture**. Our careful insistence on a realise perspective has ensured that we haven't interpreted them as molecular structures, which would in terms of BFO be generically-dependent continuants, or names. They are all, in BFO terms, snap:IndependentContinuants. They cannot inhere in anything else. **molecules** and **pure substances** are all snap:Objects—they are spatially extended, self-connected and self-contained. **salts** and **mixtures** are clearly not snap:ObjectAggregates, because all of the boundaries within them (between, for example, solutes and solvents, or positively-charged and negatively-charged ions) are connected to some other boundary within the system. There are, so to speak, no gaps.

molecular parts, for example the methyl group in a toluene molecule are snap:FiatObjectParts, in that there is no physical discontinuity between the methyl group and the benzene ring. The methyl group is a good example of 'carving nature at the joints', as the processes of adding a methyl group (methylation) and removing one (demethylation) are carried out by both enzymes and small molecules in nature and in the laboratory.

atoms and **subatomic particles** are somewhat trickier. While it is true that within an atom there are nodal points and planes where the probability of finding an electron is mathematically zero, this is not the sense of a gap in a snap:ObjectAggregate, not least because it is the same electron on either side of the nodal plane. One cannot somehow convert the nodal plane into a bigger, three-dimensional, gap and obtain two fractional electrons. Hence they are either, according to whether they are bound, snap:Objects or snap:FiatObjectParts. We therefore call them snap:IndependentContinuants.[2]

Now that we have the alignment with BFO we can write definitions for our top-level classes.

[2] I am grateful to Barry Smith for this suggestion [23], which is also a recommendation of Schulz *et al.* [24].

1. **atom** A snap:IndependentContinuant which has as determinate parts a single atomic nucleus and one or more electrons.
2. **subatomic particle** A snap:IndependentContinuant which does not have as granular parts atomic nuclei or electrons.
3. **molecule** A snap:Object which has as determinate parts two or more **atoms**.
4. **molecular part** A snap:FiatObjectPart which consists of at least one **atom** which is part of a **molecule**.
5. **collective** A snap:Object that has granular parts.
6. **pure substance** A **collective** of many **atoms** or **molecules** of a single type.
7. **salt** A **collective** of positively-charged **atoms** or **molecules** and negatively-charged **atoms** or **molecules**.
8. **mixture** A **collective** that has as ingredients **pure substances**.

5. Related Work

In addition to ChEBI, there are three serious chemical ontologies of which we are aware. The first, CO, is by Dumontier and co-workers [1,2], who have been using OWL to represent how atoms are connected to each other and have worked out necessary and sufficient conditions for defining particular molecular parts, such as a carbonyl group or a nitro group. The top-level term in [1] is OrganicGroup, but no alignment of this to a higher-level ontology is attempted. The second is ChemTop, which is simply the chemical subset of BioTop.[25] The third is the chemical ontology implicit in ChemBLAST [3], which, like CO, has no explicit commitment to a top-level ontology.

5.1. Implications for ChEBI

The current *molecular structure* tree, therefore, needs to be replaced by five trees rooted in **atom**, **molecule**, **molecular part**, **subatomic particle** and **collective**. It would be safest, in order to prevent the sort of confusion seen above, if this last were moved to a separate ontology.

The main implication of this work for ChEBI is that the current `is_part_of` relation should be replaced by appropriate `has_part` relations.[3]

The first thing to note is that the high-level classes within ChEBI do not map neatly to the high-level classes we have found here. Secondly, ChEBI is not `is_a`-complete. To take an eye-catching example, *amino-acid residues* (CHEBI:33708) has no `is_a` parents, even though it `is_substituent_group_from` *amino acids* (CHEBI:33709), so there is no inferrable connection between it and *groups* (CHEBI:24433).

The full alignment task, which we do not attempt here, is to determine the upper-level classification of each type in the ontology by inspection. Before we can use a reasoner to work with genus–differentia definitions we must first have a genus for each type. Often there are cues in the term name. We list some of these cues in Table 3.

It is also worth noting in passing some other problems with ChEBI as it is presently constituted, notably that many entity names with a closed-world reading are plural. This implies that ChEBI has been constructed in terms of instances rather than types. It

[3]Note added after acceptance: A similar changeover has now been scheduled for the October 2008 release of ChEBI.

Top-level class	Cue
atom	name is a chemical element or ends "elements"
mixture	definition includes "mixture" or "racemate"
molecule	default
molecular part	name ends in "group(s)" or "residue(s)"
pure substance	difficult to determine automatically. `is_a` descendant of *native element minerals* (CHEBI:46730)
salt	`is_a` descendant of *salts* (CHEBI:24866) or *minerals* (CHEBI:46662) but not of *native element minerals* (CHEBI:46730)
subatomic particle	`is_a` descendant of *subatomic particle* (CHEBI:36342)

Table 3. Some cues for classifying entities in ChEBI

also leads to ungrammatical-looking statements such as *metals* `is_a` *atoms*. It would be clearer to replace plural names, for example *pyrroles*, with something like "pyrrole compounds"—an alternative, "pyrrole molecule", could be interpreted as either a molecule of pyrrole (closed-world) or a molecule of a pyrrole (open-world).

5.2. Implications for the Use of InChI

Prasanna and co-workers[4] use InChIs to represent fully-specified **molecules**, under-specified **molecules** and **molecular parts**. This is intrinsically ambiguous because the bonding between atoms is only implied in the InChI by the presence or otherwise of hydrogen atoms. Thus the InChI

```
InChI=1/C6/c1-2-4-3-5-6
```

which represents either an implausible molecule which is only likely to last long in interstellar space or a fragment of a molecule, could, given the **molecular part** interpretation, indicate either a singly-bonded cyclohexane ring or an aromatically-bonded benzene ring.

This means that even the combination of this upper-level ontology for chemistry and the InChI identifier is insufficient to fully describe objects of chemical discourse without a domain ontology such as ChEBI. However, it would be useful to indicate whether a given InChI is to be understood as a grain or a collective.

5.3. Implications for BioTop

We list some example mappings in Table 4. BioTop's classes Atom and SubAtomicParticle map straightforwardly to ours.

Some of them, however, we list as undefined. The reason is that they have `is_a` descendants which belong to different upper-level types. MonoMolecularEntity and PolyMolecularCompositeEntity, for example, both have some descendants that map to **molecule** and some that map to **molecular part**. The same applies to NucleicAcid and Peptide.

Other terms are undefined because their definitions refer to different upper-level types, for example PortionOfHeterogeneousSolid, where the examples given in the

Current work	BioTop
atom	Atom
subatomic particle	SubAtomicParticle
molecule	EntireMolecularEntity
	WaterMolecule
	Lipid
	Eicosanoid
	Steroid
	EntireNucleicAcidMolecule
	FattyAcid
	EntireProteinMolecule
mixture	PortionOfHeterogeneousLiquid
molecular part	Monomer, HeterocyclicBase
undefined	MonoMolecularEntity
	NucleicAcid
	Carbohydrate
	ChainOfCarbohydrateMonomers
	PortionOfHeterogeneousSolid
	MoleculeComplex
	Peptide
	AminoAcidMonomer

Table 4. Some example mappings from BioTop to current work

BioTop definition are a NaCl crystal, which we identify as a **salt**, and graphite, which is a **pure substance**, and AminoAcidMonomer, whose definition is "Amino Acids molecules or residues (residues as in peptide bonds)".

In some cases there is not enough information to decide.

6. Conclusions and future work

We have examined ChEBI in the light of the OBO Relations Ontology (RO), Rector *et al.*'s notion of granular and collective parts and BFO, and used it to create an upper-level ontology of independent chemical continuants and propose a thorough overhaul of the structure of ChEBI with RO-compatible relationships. We have also looked briefly at how the upper-level types relate to ChEBI, BioTop and the InChI identifier. However, this is still essentially an armchair ontology. We have not attempted to test it with automated reasoning, as has been done as part of BioTop[10] or by Keet and Artale for generic parthood relations.[26] The other thing that is missing is a proper consideration (and alignment to BFO) of the dependent continuants in ChEBI. We have shown that a number of them only make sense if applied to collectives rather than grains, but we have not determined whether this is the case for all of them.

Acknowledgements

I thank Peter Corbett, Chris Mungall and Barry Smith for useful preliminary discussions. I would also like to thank the participants in a workshop on the future of ChEBI held on the 9th of July 2008 at the EBI in Hinxton, especially Kirill Degtyarenko, Janna Hastings, Barry Smith and Alan Ruttenberg.

References

[1] N. Villanueva-Rosales and M. Dumontier. Describing chemical functional groups in OWL-DL for the classification of chemical compounds. in *OWL: Experiences and Directions (OWLED 2007)*, Innsbruck, Austria, 2004.

[2] H. J. Feldman, M. Dumontier, S. Ling and C. W. V. Hogue. CO: A Chemical Ontology for Identification of Functional Groups and Semantic Comparison of Small Molecules. *FEBS Lett.*, **579**, 4685–4691.

[3] M. D. Prasanna, J. Vondrasek, A. Wlodawer, H. Rodriguez and T. N. Bhat, Chemical compound navigator: A web-based chem-BLAST, chemical taxonomy-based search engine for browsing compounds. *Proteins: Struct. Funct. Bioinformatics*, **63**, 907–917, 2006.

[4] M. D. Prasanna, J. Vondrasek, A. Wlodawer and T. N. Bhat, Application of InChI to curate, index and query 3-d structures, *Proteins: Struct. Funct. Bioinformatics*, **60**, 1–4, 2005.

[5] K. Degtyarenko, P. de Matos, M. Ennis, J. Hastings, M. Zbinden, A. McNaught, R. Alcantara, M. Darsow, M. Guedj and M. Ashburner, *Nucleic Acids Res.*, **36**, D344–D350, 2008.

[6] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall, the OBI Consortium, N. Leontis, P. Rocca-Serra, A. Ruttenberg, S.-A. Sansone, R. H. Scheuermann, N. Shah, P. L. Whetzel and S. Lewis, *Nat. Biotechnol.*, **25**, 1251–1255, 2007.

[7] Gene Ontology Consortium, *Nucleic Acids Res.*, **34**, D322-D326, 2006.

[8] H. L. Johnson, K. Bretonnel Cohen and L. Hunter, A Fault Model for Ontology Mapping, Alignment and Linking Systems. *Pacific Symposium on Biocomputing*, 2007.

[9] M. Bada and L. Hunter, Identification of OBO nonalignments and its implications for OBO enrichment. *Bioinformatics*, 2004, **24**, 1448–1455.

[10] S. Schulz, E. Beisswanger, U. Hahn, J. Wermter, A. Kumar and H. Stenzhorn, From Genia to BioTop: Towards a Top-Level Ontology for Biology, in *Proceedings of FOIS 2006. International Conference on Formal Ontology and Information Systems*, Baltimore, USA, 2006.

[11] T. Ohta, Y. Tateisi and J.-D. Kim, The GENIA corpus; An annotated research abstract corpus in molecular-biology domain, in *HLT 2002 – Proceedings of the 2nd International Conference on Human Language Technology Research*, pp. 82–86

[12] B. Smith, W. Ceusters, B. Klagges, J. Köhler, A. Kumar, J. Lomax, C. Mungall, F. Neuhaus, A. L. Rector and C. Rosse, Relations in biomedical ontologies, *Genome Biol.*, **6**, R46, 2005.

[13] A. Rector, J. Rogers and T. Bittner, Granularity, scale and collectivity: When size does and does not matter, *J. Biomed. Informat.*, **39**, 333–349, 2006.

[14] P. Grenon, B. Smith and L. Goldberg, Biodynamic Ontology: Applying BFO in the Biomedical Domain. *Ontologies in Medicine*, ed. D. M. Pisanelli, IOS Press, Amsterdam, 2004, pp. 20–38.

[15] See `http://www.iupac.org/inchi`, accessed 2008-07-18.

[16] O. Bodenreider, B. Smith and A. Burgun, The Ontology-Epistemology Divide: A Case Study in Medical Terminology. in *Proceedings of FOIS 2006. International Conference on Formal Ontology and Information Systems*, Turin, Italy, 4–6 November 2004.

[17] B. Smith, J. Köhler and A. Kumar, On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology. in *Data Integration in the Life Sciences (DILS 2004)*, Leipzig, March 2004.

[18] G. P. Moss, Nomenclature of steroids (recommendations 1989), *Pure Appl. Chem.*, **61**, 1783–1822, 1989.

[19] *IUPAC Compendium of Chemical Technology*, Electronic version, `http://goldbook.iupac.org/C01022.html`, accessed 2008-07-15.

[20] P. Corbett, C. Batchelor and A. Copestake, Pyridines, pyridine and pyridine rings: disambiguating chemical name entities, in *Proceedings of BERBMTM-08 at LREC-2008*, Marrakech, Morocco, 2008.

[21] C. Masolo, S. Borgo, A. Gangemi, N. Guarino and A. Oltramari, "Ontology Library", WonderWeb Deliverable 18, WonderWeb consortium, 2003.

[22] H. Herre, B. Heller, P. Burek, R. Hoehndorf, F. Loebe, and H. Michalek. General Formal Ontology (GFO): A Foundational Ontology Integrating Objects and Processes. Part I: Basic Principles. Research Group Ontologies in Medicine (Onto-Med), University of Leipzig.

[23] B. Smith, personal communication.

[24] S. Schulz, M. Boeker and H. Stenzhorn, How Granularity Issues Concern Biomedical Ontology Integration. in *Proceedings of the International Congress of the European Federation for Medical Informatics (MIE 2008)*, Gothenburg, Sweden, May 2008.

[25] See `http://purl.org/chemtop/dev`. Accessed 2008-08-17.

[26] C. M. Keet and A. Artale, Representing and Reasoning over a Taxonomy of Part-Whole Relations, *Applied Ontology*, in press.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-208

SKI*ing* with DOLCE: toward an e-Science Knowledge Infrastructure

Boyan Brodaric[1], Femke Reitsma[2], Yi Qiang[2]
[1]Geological Survey of Canada, Ottawa, brodaric@nrcan.gc.ca
[2]UK National e-Science Center, University of Edinburgh, femkereitsma@gmail.com, yi_qiang@hotmail.com

Abstract: An ontology of general science knowledge (SKIo) is developed to enhance machine representation and use of scientific theories in emerging e-Science Knowledge Infrastructures. SKIo specializes the DOLCE foundational ontology with science knowledge primitives, such as science theory, model, data, prediction, and induction. These are arranged to reflect the complex knowledge structures used in science, such as scientific ideas playing different roles within and between theories. SKIo is encoded with OWL-DL, uses the DOLCE Descriptions and Situations module, and provides defining conditions for its primitives to enable an extensible bridge between DOLCE and domain science ontologies. An application to environmental theories is demonstrated, and its utility to other natural sciences is promising.

Keywords: science ontology, scientific reasoning, DOLCE, environmental model

1. Introduction

Ontology-enabled infrastructures in support of cyber-based scientific activity, or e-Science, are being developed and used in many science domains [1]. This is leading to significant scientific and societal benefits, in that faster computations are occurring over more data and the resultant predictive models are providing larger and more accurate scenarios about situations affecting humans and the environment. Although these early e-Science achievements are laudable and significant, they do fall short of a broader e-Science vision in which scientists not only operate over more observed data to make better predictive models, but also directly use e-Science infrastructures to find, generate and test science theories. This broader vision of e-Science requires a Science Knowledge Infrastructure (SKI [2]) that enables the capture, representation and use of the full spectrum of science knowledge. Using SKIs, scientists should be able to annotate existing resources, such as observed data and predicted models, with respect to potentially competing science theories, in order to enable knowledge search and evaluation, as well as facilitate reproduction of simulations, experiments and results.

The present focus on a fragment of science knowledge has some negative consequences as it limits full scientific discovery and reproducibility in e-Science infrastructures. This occurs because only some aspects are explicitly represented in the infrastructure (e.g. data, models, concepts), while other knowledge (e.g. science theories) is largely implicit as it is buried in scientists' heads and in ancillary resources such as textbooks, papers, reports and maps. An initial challenge then is the development of a computable representation of a wide suite of science knowledge primitives. Foundational ontologies are a good candidate for such representation not only due to their formality, rigor and commitment to internal coherence, but also due to their generality in that, like science knowledge primitives, the contents of foundational ontologies are intended to be re-used across science domains. This contrasts with the numerous ontologies being developed for specific science domains, and is aligned with the few that are being developed as a general science superstructure, but these latter are narrowly focused and do not often utilize a foundational ontology.

In this paper we specialize the DOLCE foundational ontology [3] with a modest number of science knowledge primitives, such as science model, science theory, data, prediction and induction, and test the resultant ontology (SKIo) by representing environmental theories. Section 2 describes a typical use-case scenario in the environmental sciences; Section 3 discusses related work; Section 4 explains our general approach based on computationally inspired renditions of the science knowledge cycle; Section 5 presents SKIo; Section 6 outlines some results from using SKIo to represent environmental theories; and Section 7 concludes with a brief summary.

2. An Environmental Modeling Use-Case Scenario

Problem Scenario: Jane is a scientist who wants to integrate into a global climate scenario some model of Net Primary Production (NPP), which involves the conversion of solar energy, carbon dioxide and water, into biomass. She begins searching using a few keywords in Google as well as in her University library database. She finds a huge number of results, and after much sifting has a large collection of papers that seem relevant but cannot be easily differentiated, largely because they use polysemous terminology. For example the key term “model” is used in several senses in the environmental modeling literature, but only the first two senses are relevant to her:

1. model = a system of equations to support calculations and simulations [4];
2. model = a theory with equations and broader scientific implications [5];
3. model = a simulation software with equations and implications [6];
4. model = the results of a simulation run, or other process, in which some geographically located climatic situation is represented [6].

Proposed Solution: Jane logs on to a web-based SKI and begins searching for a relevant “model” by using a number of concepts she is familiar with in NPP modeling; she expects these concepts to be used as variables in equations. Because the different senses of “model” are well demarcated in the SKIo ontology, and because the SKI’s contents are annotated by this ontology, she is able to find candidate “models” corresponding to senses 1 and 2 linked to digital resources. After integrating a newly found model and running her experiment, she creates a web page documenting the process, appends a draft publication, and annotates the resource in the SKI.

Additional Requirements: SKIs should also help scientists resolve questions such as: who else has solved problem p, or a similar problem in another domain? Who is working in the same research field? What results when existing theory x_1 is replaced by a new theory x_2 and tested against data y as originally reported in journal paper z?

What other data satisfy x_2, and to what degree? What other theories are satisfied by y, in which papers, and how do these differ from x_1? How was x_1 derived—what observed data, reasoning, and verification procedures were used, as reported in which papers? What other theories is x_1 part of, and what is its role in those theories? What theories have been derived from x_1? What theories could be derived from x_1 that satisfy y?

3. Related Work

Although ongoing work on science ontologies is vast, and growing, at present a machine-readable ontology for general science knowledge does not exist. Existing initiatives emphasize the computational representation of the science knowledge cycle, the development of ontologies that span aspects of all sciences, are limited to one science domain, or which incorporate foundational ontologies:

- **The Science Knowledge Cycle:** Several accounts of the science knowledge cycle begin to distill the numerous and complex philosophical approaches into representations amenable to computation. These focus on identifying key primitives in the cycle [7, 8, 9, 10] for incorporation into schemas [2], formal reasoning systems [11], or machine-readable science theories [12], but without explicitly representing the primitives in a machine-readable ontology.
- **Ontologies of Science:** by science ontologies we mean a conceptualization of general science knowledge primitives that can be applied across a wide breadth of science domains and which are well defined and represented in a formal language. Existing science ontologies meet this definition partially because they focus on a fragment of key primitives such as science experiments [13] or publications [14].
- **Domain Science Ontologies**: though ontologies are being developed in numerous science domains [e.g. 15, 16], they cannot serve as a superstructure for science knowledge because the abstractions are not sufficiently general. They are also used primarily for engineering purposes to facilitate data interoperability and workflow operation [17, 18] rather than to annotate and test new science ideas. Many are built bottom-up from existing vocabularies [16] and not around systematic ontological principles such as those utilized by foundational ontologies, resulting in diverse ontological assumptions that are not easily recognized or reconciled.
- **Foundational Ontologies and Science Knowledge:** foundational ontologies provide a superstructure containing the most general abstractions that can be extended to both general science ontologies and domain science ontologies, e.g. DOLCE, BFO, GFO, SUMO, Sowa's [8, 19, 20, 21, 22]. An ideal arrangement would then position a general science ontology as a layer between a foundational ontology and domain science ontologies. With one exception [16], this intermediary layer is at present missing: domain science ontologies directly specialize existing foundational ontologies or related logical theories [23, 24, 25].

4. Approach

SKIo is first represented in UML [26], and then in OWL-DL (http://www.nesc.ac.uk/technical_papers/skio3.owl). Following OWL terminology conventions, the science knowledge primitives are denoted as classes and properties: classes refer to abstractions that can be instantiated in one or more individuals, and properties refer to relations between two classes; individuals are single entities that instantiate a class, i.e. instances. Class and property names are shown in italics herein.

Several principles are followed in the design of SKIo (after [27]): (1) **Modularity**—general science classes and properties are to be added as leaves to the DOLCE hierarchy of classes and properties, such that the original hierarchical structure remains unchanged; (2) **Semantic Grounding**—SKIo is to be founded on recognized accounts of the science knowledge cycle; (3) **Semantic Coverage—**sufficient breadth of the science knowledge cycle is to be encompassed such that SKIo could be specialized by general and domain science ontologies; (4) **Semantic Precision—**sufficient depth is to be attained to enable annotation of science documents through instantiation of SKIo primitives; (5) **Coherence:** these principles are to be formalized to enhance definition and understanding of SKIo components.

5. The SKI ontology

The DOLCE 2.1 (OWL 397) ontology consists of four core classes that categorize particulars (singular entities in the world): *endurant*, *perdurant*, *quality*, and *abstract*. An *endurant* is an object-like entity that is wholly present at any point in time it exists, but whose characteristics can change over time (rock body, building, country); a *perdurant* is a process-like entity that is not wholly present at any point in time it exists, such as a *process*, *event*, or *state* (San Andreas faulting, San Francisco earthquake, being seismically active); a *quality* is a dependent characteristic inherent in an *endurant*, *perdurant*, or *abstract*, such that an *endurant* inheres physical qualities (geospatial position, size, shape, color), a *perdurant* inheres temporal qualities (duration, age), and an *abstract* or *non-physical-endurant* inheres abstract qualities (the value of the Canadian dollar); an *abstract* is an entity that does not possess physical or temporal qualities, and is often the value of a *quality*, or a space containing those values (the number 2, the munsell color space, blue).

SKIo specializes the DOLCE foundational ontology with a relatively small number of general science knowledge primitives synthesized from computational accounts of the science knowledge cycle (mainly [9, 11]). In this cycle, shown in Figure 1, scientific artifacts are produced by scientific activities: empirical regularities are induced from observed data, hypothetical propositions are abduced from all prior knowledge, predictions about the real world are deduced from empirical regularities or hypothetical propositions, and predictions are verified through further interaction with the world, which involves activities such as data collection, problem finding, and building models of the world. SKIo specializes the *activity* subclass of *perdurant* to include such science activities and it specializes some physical and social subclasses of *endurant* (*description*, *situation*, *concept*, *information-object*, *physical-endurant*) to include various science artifacts. To foster reproducibility and enhance explanation, each science artifact is defined at least in part by the science activity that produces it.

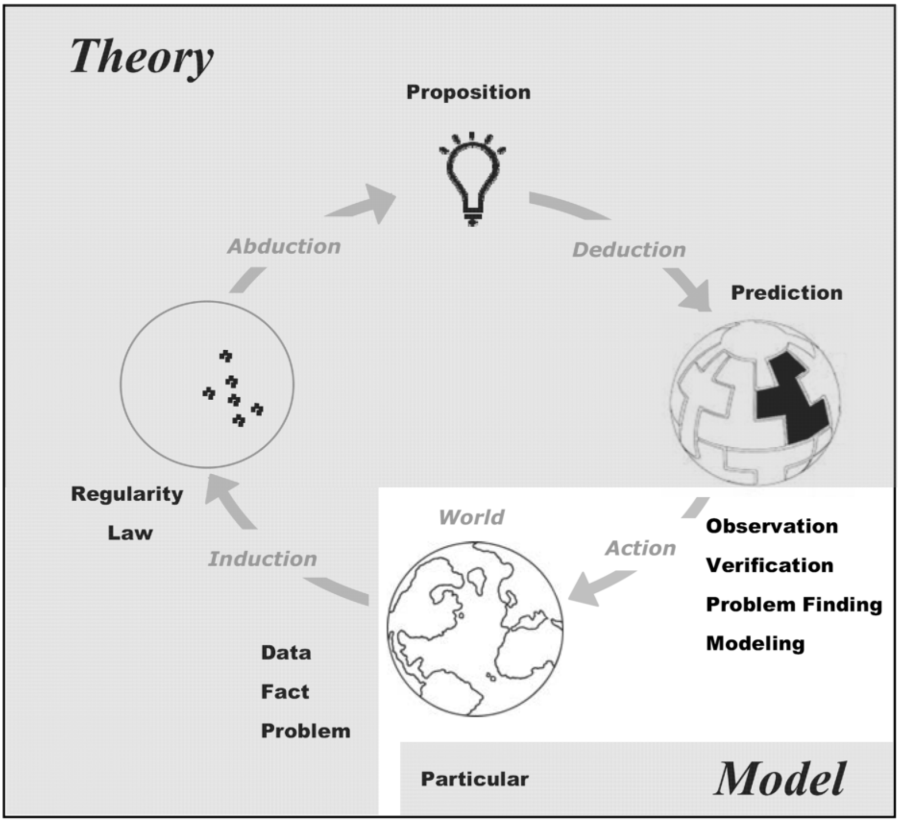


Figure 1: the knowledge cycle in SKI [after 8, 9]; grey areas represent some theory and model contents.

5.1. Descriptions and Situations

DOLCE's descriptions and situations are designed to represent socially constructed contexts and the states-of-affairs interpreted by those contexts, respectively [28]. Although descriptions and situations have been applied to the biomedical domain [24], they are specialized by SKIo into general science classes to provide scientific constraints on their meaning. In SKIo, a description is a science idea that is syntactically expressed as a *ScienceStatement* (a DOLCE *information-object*), such as a text body, figure, or web site, and which is contained in a *SciencePublication* as well as physically manifest in multiple forms such as in hardcopy or computer memory. A science theory is then a science idea comprised of one or more coherent descriptions that characterize the structure or behavior of some aspect of reality in sufficient generality to satisfy a wide number of science models and be used to predict the particulars in the models [2]. Specifically, a *ScienceTheory* contains descriptions which are satisfied by (can be used to scientifically predict) the particulars in the science models, and conversely a *ScienceModel* contains scientifically discovered particulars which satisfy (are scientifically predictable by) some *ScienceTheory*. Specializations include *GeoScienceTheory* and *GeoScienceModel* consisting of geoscience particulars.

For example, if a geoscience theory consists of equations then satisfaction implies that model members can be calculated from the application of data to the equations. A1 states this in a first order logic reference statement via the *predictable-by* property, which takes as its range a *Prediction* (the result of a *Deduction*), indicating a particular is forecast by the prediction. Formally, given *ScienceModel* (M), *ScienceTheory* (T), *Prediction* (p), *particular* (m) a member of model (M), and a *particular* (y) then:

(A1) $\forall (m \in M)\, \exists T\, [\textit{satisfies}(M,T) \leftrightarrow \exists p\, [\textit{predictable-by}(m, p, T) \rightarrow \exists y\, \textit{Deduction}(T, y, p)]]$

When the description itself is a concept definition, then any satisfying model is part of the extension of the concept, because each particular in the model can be deductively classified from the concept definition. This includes the case where some extension members might be designated as prototypes for the concept. In SKIo, a *Definition* is a canonical idea included mainly for explicitness, as DOLCE descriptions can be definitions implicitly. *Data* is also a science idea, one that results from the observation or inference of some quality. Specifically, *ObservedData* results from the observation of physical or temporal qualities, as it is assumed that abstract qualities are not observable but are inferred. The syntactic expression of some data is a *DataSet*.

5.2. Concepts

The DOLCE *concept* class is a contextualized socio-cognitive artifact, one used to classify a particular within a situation to enable it to satisfy a description [29]. DOLCE provides three types of concepts: *role*, *course*, and *parameter*, for classifying endurants, perdurants and quality regions, respectively. Concepts are related to descriptions in four ways in SKIo: (1) a concept can be defined by a description; (2) a description (e.g. a theory part) uses concepts in its body to describe an idea; (3) the *role* concept is specialized to *ScienceRole* to represent the science function performed by a theory or its part, because a science idea can be expressed by many statements and can play different roles within and between theories; and (4) a science theory can maintain an index of its component concepts such as parameters or science roles. For example, the theory of special relativity has as a part the idea $e = mc^2$ that: defines the *parameter* concept *energy*; *d-uses parameter* concepts *energy* (*e*), *mass* (*m*), and *constant speed of light* (*c*); *plays* the *ScienceRole* of *Proposition* within the theory; and is indexed by the theory for its *ScienceRole* and *parameter* concepts which become components of the theory. In subsequent theories the same idea plays the role of *Assumption* or *ScienceProblem* [30]. Key science roles in SKIo include:

- *Assumption*: is defined by an originating *Assertion* and is considered primitive such that the related idea is not necessarily empirically supported or inferred.
- *ScienceProblem*: is defined as the product of *ProblemIdentification* in which a theory part is identified as problematic because it exhibits inconsistencies (initially obtained via *Verification*) with observed data, or inferred theory or models.
- *Fact*: is the incorporation of some *Data* into some theory. Because of this dependence facts are always ‘theory-laden’. Facts also support specification of the scientific discovery of a particular, in that a fact can indicate that data leads to the identification of a particular, e.g. a magnetic measurement indicates a rock body.

- *EmpiricalRegularity*: is defined as a situational empirical pattern produced by *Induction*. Situational refers to the case where the regularity is satisfied by some but not all of the possibly valid science models. For example, if the regularity is expressed as a relation amongst concepts, then the relation is present only in some but not all situations in which particulars classified by the concepts are jointly present. The regularity might not be present universally because of insufficient verification or because the pattern is dependent on certain historical conditions that are temporary and change in time. This is implemented in SKIo by requiring the regularity's existence to be dependent on one or more endurants or perdurants, likely some subset of those involved in its original induction.
- *ScienceLaw*: is defined as an universal empirical pattern produced by *Induction*, i.e. its existence is not dependent on any specific endurants or perdurants. Empirical regularities and science laws can evolve toward each other by the logic of induction: additional data might suggest that a situational pattern is universal, and conversely more data might contradict a science law and reveal it to be situational.
- *Prediction*: is defined as a conjecture about individuals produced via *Deduction* that can be empirically verified. Because only physical and temporal qualities are observable it follows only these qualities are predictable, but SKIo does not impose this constraint as it is convenient to also predict individuals.
- *Proposition*: is a best-guess conjecture produced via *Abduction*, which can be situational or universal, and about individuals or theories.

5.3. Activities

DOLCE activities are non-atomic perdurants that follow some plan, sequence some tasks, can produce some endurants, and are performed by some agents. In SKIo the plan is likely some research project containing tasks performed by scientists, or some computational methods containing procedural tasks syntactically expressed in SKIo *Software* and performed by computers. SKIo activities include observation, inference, assertion, verification, problem finding, science modeling, and doing research. These produce key science artifacts such as science models, science roles, and information objects. Of particular importance are the *Inference* activities as they bind together much of the knowledge cycle (after [8, 9, 11]):

- *Induction*: involves finding a pattern in data (logical induction), or dis/confirming a pattern, or instance, via data collection and evaluation (pragmatic induction). In logical induction: given data $\{(a_1, b_1), (a_2, b_2), (a_3, b_3)\}$ then infer T (A, B), where T is some theoretical relation over classes A, B, and a_i and b_i are their respective instances. In pragmatic induction: given T (A, B), note in newly observed data that $\{(a_1, b_1), (a_2, b_2), (a_3, b_3)\}$ and infer T (A, B) $\models$ TRUE (or FALSE if disconfirmed), or given a_1 note in data $\{a_1, a_1,\ldots\}$ and infer a_1 = TRUE (or FALSE if disconfirmed). SKIo *Induction* refers to logical induction performed on *Data* where the induced T is an *EmpiricalRegularity* or *ScienceLaw*, while *Verification* encompasses pragmatic induction resulting in dis/confirmation of a science role when tested against data, or in/coherency when tested against theoretical relations.
- *Deduction*: involves generating a prediction about the world using existing theory and data. Logically, given theory T:A→B (A and B are classes) and instance a_1 (of

A), then b_1 (of B) is deduced: $T \wedge a_1 \vdash b_1$. In SKIo *Deduction*, T is realized as a theoretical science role (*Proposition*, *EmpicalRegularity*, *ScienceLaw*), a_1 as a science role about an observed or inferred instance (*Fact*, *Prediction*, *Proposition*), and b_1 as the resultant *Prediction*.

- *Abduction*: involves guessing a *Proposition* to enable coherence of discordant science knowledge. Logical abduction is reverse deduction: in a deduction of the form $T \wedge a_1 \vdash b_1$, as above, where T and b_1 are known, then guess the missing instance a_1. Pragmatic abduction, on the other hand, is more concerned with guessing theory T via transformation from radically different prior knowledge (often via analogy ~): e.g. given T_1:C→D, and ~(A,C) ∧ ~(B,D), then T_2:A→B. SKIo *Abduction* encompasses both logical and pragmatic abduction, such that the resultant *Proposition* refers to either a_1 (an instance) or T_2 (a theoretical relation).

6. Application of SKIo to Environmental Modeling

To exemplify SKIo, we represent NPP theories (i.e. "models" [4]) such as those sought by Jane in Section 2. The theories are expressed primarily as systems of equations containing (1) input variables and (2) fixed values segregated into general constants (e.g. atmospheric pressure) and specific parameters (e.g. temperature). The theories can be categorized into biogeographic or biochemical, where the former are empirically inferred from data and the latter are derived from existing theories and express biochemical processes more explicitly. Some of these theories share a few ancillary theories which are obtained from other sources such as books or web pages. SKIo representation of all these elements involves the following SKIo classes, which are applied to the BIOME3 Model [31] in Figure 2:

- Each paper, book, and web site that expresses the theories is represented as an instance of *SciencePublication* containing *ScienceStatement* instances.
- Each equation, or other theory part such as a table or figure, is expressed as a distinct instance of *ScienceStatement*.
- The description of each theory is represented as a whole *GeoScienceTheory* instance, and is expressed by the relevant *SciencePublication* instances.
- The description of each equation, or other theory part, is represented as a *GeoScienceTheory* part instance, and is expressed by *ScienceStatement* instances.
- The variables (e.g. temperature) and fixed values in each equation are represented as DOLCE *parameter* classes such that an instance of each parameter is *d-used* by each equation, and each parameter instance classifies (is *valued-by*) a region instance containing the value for the parameter. The variables are also realized as qualities of geoscience particulars in associated models.
- Each part of each empirically inferred theory is represented as an *EmpiricalRegularity* instance, because these are largely induced from empirical data and are local to Earth situations (the fixed values are calibrated to the Earth).
- The parts of the theories that were derived from other theories play a *Proposition* science role, because these are theoretically postulated (hence via abduction).
- Some theories share common parts which play different roles amongst the theories, e.g. a theory developed by [32] adopts a *Proposition* from [33] as an *Assumption.*

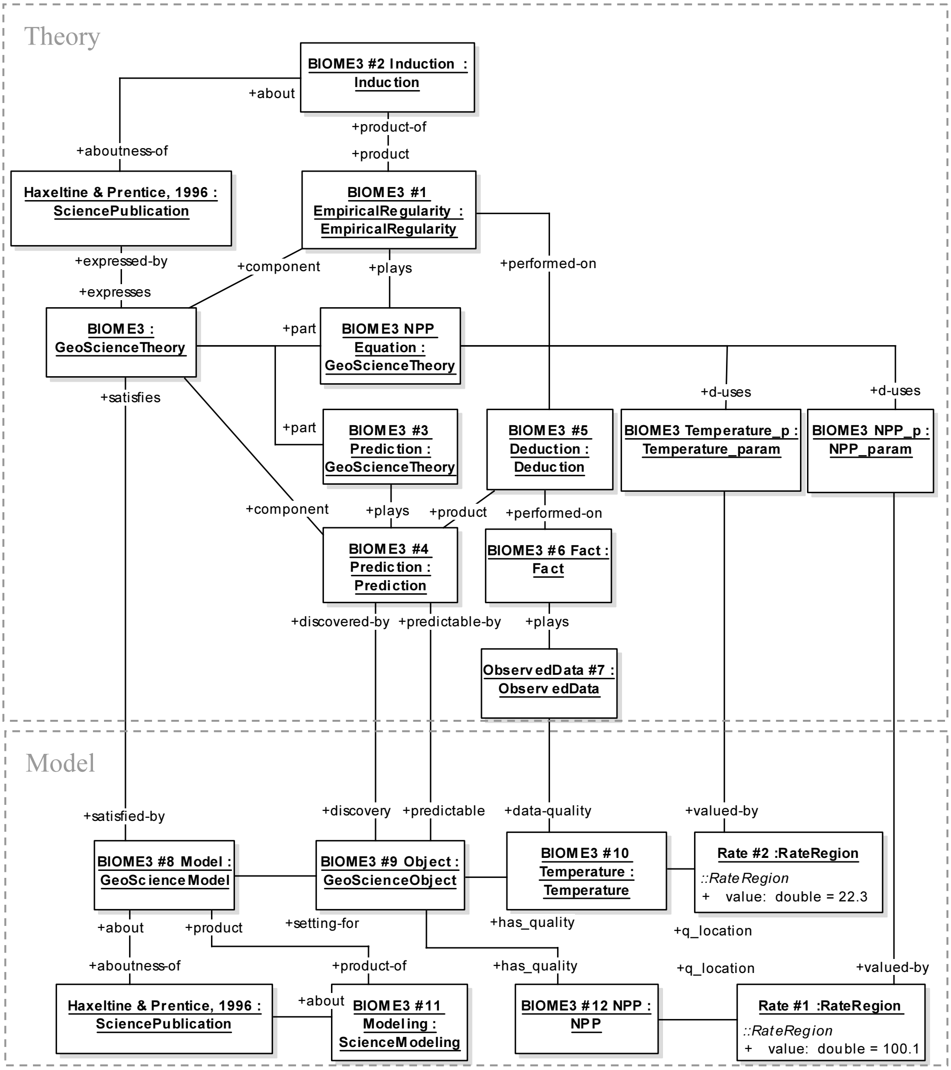


Figure 2: partial SKIo representation of the BIOME3 NPP theory [31] and a related model; in the boxes, the text before the colon denotes the instance name, and the text after the colon denotes the SKIo class name.

- Many of the theories, such as the Miami Model [34] or BIOME3 Model [31], are satisfied by well known *GeoScienceModel* instances which are either based on empirical facts (contain particulars generated from observed data) or predictions (contain particulars generated from inferences).

7. Conclusions

The SKIo ontology meets the representation and evaluation requirements outlined in Sections 2 and 4. The four senses of “model” from Section 2 are disambiguated as

ScienceTheory (senses 1 and 2), *ScienceModeling* (sense 3), and *ScienceModel* (sense 4). The evaluation criteria from Section 4 are satisfied via: (1) modular specialization of DOLCE; (2) grounding in primitives adopted from the science knowledge cycle; (3) specialization of domain ontologies via representation of geoscience classes such as *GeoscienceModel* and *GeoScienceTheory* as well as *parameter* classes such as NPP and temperature; (4) representation of environmental knowledge from several peer reviewed papers; and (5) definition and OWL formalization of the science knowledge primitives. Future coupling of SKIo to an operational SKI should then facilitate the search, retrieval and use of basic science knowledge. For example, Jane could obtain knowledge about NPP by searching for geoscience theories that use the NPP concept. For the correlation between annual average temperature and NPP, Jane could retrieve the Miami Model [34] and the theory of [32], including a predicted model of world NPP. Consequently, we assume SKIo can be applied in other sciences and are encouraged that this will advance next-generation e-Science by facilitating not only computing operations in existing infrastructures, but also the discovery and testing of science artifacts in emerging SKIs. Limitations of SKIo include incompleteness in representing elements such as science methods and instruments, representing science knowledge change, and guides for specializing social versus physical DOLCE classes.

Acknowledgements

The authors gratefully acknowledge support of the U.K. e-Science Institute under theme 4 “Spatial Semantics for Automating Geographical Information Processes”, and the valuable comments of the three reviewers as well as F. Probst and M. Gahegan.

References

[1] Hendler, J. (2004). Science and the Semantic Web. Science 299(5606):520-521.
[2] Hars, A. (2001). Designing Scientific Knowledge Infrastructures: The Contribution of Epistemology. Information Systems Frontiers, 3(1):63-73.
[3] Gangemi, A., Guarino, N., Masolo C., Oltramari, A. (2003). Sweetening WordNet with DOLCE, AI Magazine, 24(3):13-24.
[4] Adams, B., A. White, and T. M. Lenton. (2004). An analysis of some diverse approaches to modelling terrestrial net primary productivity. *Ecological Modelling* 177 (3-4):353-391.
[5] Baldocchi, D., Finnigan, J., Wilson K., Paw U K.T., Falge E. (2000). On Measuring Net Ecosystem Carbon Exchange Over Tall Vegetation on Complex Terrain. Boundary-Layer Meteorology, 96(1-2), August 2000, pp. 257-291(35).
[6] Phillips, T. J., AchutaRao, K., Bader, D., Covey, C., Doutriaux, C. M., Fiorino, M., Gleckler, P. J., Sperber, K. R., Taylor, K. E. (2006). Coupled climate model appraisal, a benchmark for future studies. EOS Trans. AGU 87(19): 185.
[7] Shrager, J., Langley, P. (1990). Computational Models of Scientific Discovery and Theory Formation. Morgan Kaufman, San Mateo.
[8] Sowa, J. F. (2000). Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks/Cole: New York, 594 pp.
[9] Sowa, J. F. (2006). The Challenge of Knowledge Soup. In: J. Ramadas, S. Chunawala, (Eds.), Research Trends in Science, Technology and Mathematics Education, Homi Bhabha Centre, Mumbai, 2006, pp. 55-90.
[10] Thagard, P. (1988). Computational philosophy of science. MIT Press, Cambridge, MA.

[11] Ray, O. (2005). Hybrid Abductive Inductive Learning. PhD Thesis, Imperial College, University of London, http://www.bcs.org/upload/pdf/oray.pdf.

[12] Poole, D., Smyth, C., Sharma, R. (2008). Semantic Science and Machine-Accessible Scientific Theories. In: McGuinness, D.L., Fox, P., Brodaric, B. (Eds.), Semantic Scientific Knowledge Integration, AAAI Spring Symposium, Technical Report SS-08-05, AAAI Press, Menlo Park, CA, 81-86.

[13] Soldatova, L.N., King, R.D. (2006). An ontology of scientific experiments. Journal of the Royal Society Interface, 3(100):795-803.

[14] Benjamins, V. R. Fensel, D., Gómez Pérez, A. (1998). Knowledge Management through Ontologies. In Proceedings, Knowledge Engineering: Principles and Methods, PAKM-98, pp. 5.1-5.12, 1998.

[15] McCray, A. (2003). An upper-level ontology for the biomedical domain. Comparative and Functional Genomics, 4:80–84.

[16] Raskin, R.G., Pan, M.J. (2005). Knowledge representation in the Semantic Web for Earth and Environmental Terminology (SWEET). Computers and Geosciences, 31(9):1119-1125.

[17] Ludäscher, B., K. Lin, S. Bowers, E. Jaeger-Frank, B. Brodaric, C. Baru, C. (2006). Managing scientific data: From data integration to scientific workflows. In: Sinha, A.K. (Ed.) Geoinformatics: Data to Knowledge, GSA Special Paper 397, 109-130.

[18] McGuinness, D. L., Fox, P., Cinquini, L., West, P., Garcia, J., Benedict, J. L., Middleton, D. (2006). Enabling Scientific Research using an Interdisciplinary Virtual Observatory: The Virtual Solar-Terrestrial Observatory Example. AI Magazine, 29(1):65-76.

[19] Grenon, P., Smith, B. (2004). SNAP and SPAN: Towards Dynamic Spatial Ontology. Spatial Cognition and Computation, 4(1).

[20] Herre, H., Heller, B., Burek, P., Hoehndorf, R.Loebe, F., Michalek, H. (2006). General formal ontology (GFO) – part I: Basic principles. Technical Report 8, Onto-Med, University of Leipzig, 2006.

[21] Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A. (2003). WonderWeb DeliverableD18, Ontology Library (final). Laboratory For Applied Ontology, http://www.loa-cnr.it/Papers/D18.pdf.

[22] Pease, A., (2006). Formal representation of concepts: The Suggested Upper Merged Ontology and its use in linguistics. In: Schalley, A. C., Zaefferer, D. (Eds.), Ontolinguistics. How Ontological Status Shapes the Linguistic Coding of Concepts. Vorbereitung Berlin:New York (Mouton de Gruyter).

[23] Bittner, T. (2007). From top-level to domain ontologies: Ecosystem classifications as a case study. In M. Duckham, B. Kuipers, L. Kulik, and S. Winter, editors, Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science. International Conference, COSIT 2007, 61-77.

[24] Gangemi, A., Catenacci, C., Battaglia, M. (2004). Inflammation Ontology Design Pattern: an Exercise in Building a Core Biomedical Ontology with Descriptions and Situations. In D.M. Pisanelli (ed.) Ontolgies in Medicine, IOS Press, Amsterdam, Berlin, 2004, pp. 64-80.

[25] Grenon, G., Smith, B., Goldberg, L. (2004). Biodynamic Ontology:Applying BFO in the Biomedical Domain. In: Pisanelli, D.M. (ed.), Ontologies in Medicine, Amsterdam: IOS Press, 2004, 20–38.

[26] Brodaric, B. (2008). Science Knowledge Infrastructure Ontology 3.0. U.K. e-Science Technical Report Series, Report UKeS-2008-03, 40 pp., http://www.nesc.ac.uk/technical_papers/UKeS-2008-03.pdf.

[27] Gruber, T.R. (1995). Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal Human-Computer Studies. 43(5-6): 907-928.

[28] Gangemi, A., Mika, P. (2003). Understanding the Semantic Web through Descriptions and Situations. In: Meersman R, *et al*. (eds.), International Conference on Ontologies, Databases and Applications of Semantics (ODBASE 2003), Catania, (Italy), November 3-7, 2003.

[29] Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., Guarino, N. (2004). Social Roles and their Descriptions. In: D. Dubois, C. Welty, M.A. Williams (eds.), Proceedings of the Ninth International Conference on the Principles of Knowledge Representation and Reasoning (KR2004), Whistler, Canada, June 2-5, 2004, pp. 267-277.

[30] Kaku, M. (2004). Einstein's Cosmos: How Albert Einstein's Vision Transformed Our Understanding of Space and Time. Norton, New York.

[31] Haxeltine, A., Prentice, I.C. (1996). BIOME3: an equilibrium terrestrial biosphere model based on ecophysiological constraints, resource availability and competition among plant functional types. Global Biogeochem. Cycles 10:4, 693–70.

[32] King, A. W., W. R. William, and W. M. Post. (1995). In search of missing carbon sink: a model of terrestrial biospheric response to land-use charge and atmosphere CO2. *Tellus. Series B* 47:501-519.

[33] Polglase, P. J., and Y. P. Wang. (1992). Potential CO2-enhanced carbon storage by the terrestrial biosphere. *Australian Journal of Botany* 40 (641-656).

[34] Leith, H. (1975) Modeling the primary productivity of the world. In *Primary Productivity of the Biosphere*, ed. H. Leith, Whittaker, R.H. New York: Springer-Verlag.

This page intentionally left blank

6. Processes and Events

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-223

On the Syntax and Semantics of Effect Axioms

Haythem O. ISMAIL
German University in Cairo
Department of Computer Science
haythem.ismail@guc.edu.eg

Abstract. Effect axioms constitute the cornerstone of formal theories of action in AI. They drive standard reasoning tasks, especially prediction. These tasks need not be coupled with actual acting; the reasoning agent is, thus, typically given an *ex post acto* narrative of what actions took place. An *acting* agent, however, has no access to such knowledge; it needs to face what we call the event categorization problem, and figure out what actions it did. Until this is achieved, effect axioms will be useless. A careful review of the literature on effect axioms reveals that their syntax, semantics, and ontological commitments are so deeply entrenched in the armchair reasoning about action paradigm, that they cannot be used in resolving the event categorization problem. By enriching the ontology of action theories, we propose a different approach for representing effects of actions that unifies the two views. The enriched ontology is independently motivated by linguistic concerns.

Keywords. Knowledge representation, actions, events, states.

1. Introduction

Reasoning about action has kept the AI agenda busy for several decades. It has been the origin of such important and nagging problems as the frame problem, the ramification problem, and the qualification problem [1, for instance]. Investigating these problems has culminated in a number of mature action theories that allow reasoning about action to go mostly unhindered [1,2,3, for example].

It is important, however, to distinguish two modes of reasoning about action:

Mode 1. Given knowledge of the executability and effects of some actions, we need to answer questions pertaining to the outcome of executing action instances in a given situation. Note that this is pure armchair reasoning; no *acting* needs to be actually taking place. This reasoning mode is the one with which the theories of action indicated above are mostly concerned.

Mode 2. An agent is acting, and needs to reason about its own actions as it executes them. In this paper, I am not concerned with aspects of such reasoning related to execution monitoring and error recovery in the context of planning. Rather, I focus on the less widely considered aspect related to beliefs of agents about what they are doing, and *have* done, as their own actions unfold in time. Such concerns mainly emerge in the philosophy of action [4,5,6, for example], though AI has occasionally been interested in these questions [7,8].

I will argue that taking the Mode-2 perspective raises problems that are different from those addressed by current Mode-1 theories of action. In particular, armchair reasoning about action naturally assumes that the reasoning agent knows what actions took place (they are given as part of a "narrative"), and the task is to predict outcomes. (Not that it is a simple task.) For an acting agent, however, knowledge of what actions took place is not readily available. Even in the simple, single-agent world usually investigated in AI, the lonely agent needs to reason in order to figure out what it did; it needs to be able to *categorize* its activity before it can figure out the ramifications or even report (possibly in a natural language) on what it did [9]. The reasoning problem of an agent's categorizing its own activities will be referred to as the *event categorization problem*.

Perhaps the central ingredient of theories of action are effect axioms: axioms capturing the causal laws of the domain. A close examination of the literature shows that current treatments of effect axioms are so deeply entrenched in the Mode-1 paradigm that their syntax, semantics, and ontological commitments are not suitable for Mode-2 reasoning. By enriching the ontology of action theories, we propose a slightly different approach to representing the effects of actions that can account both for Mode-1 and Mode-2 reasoning. Incidently, the enriched ontology is independently motivated by linguistic concerns.[1]

Effect axioms are examined in Section 2. Section 3 motivates the event categorization problem. Sections 4 and 5 analyze the inadequacy of effect axioms, in their current form, to account for event categorization. An approach to effect axioms that does this is presented in Section 6.

2. Effect Axioms: Round I

In formal AI action theories, effect axioms are logical formulas which specify, for each action, its *direct* effects. These are effects that cannot be specified by general domain constraints. Table 1 lists some of the common approaches to representing effects of actions. Unbound variables are universally quantified with widest scope. The examples are drawn from the extensively-studied Yale shooting scenario [10].

The main difference between the theories presented in [11], [1], [2], and [3] (other than the logics they employ) is their treatment of major problems, such as the frame problem. However, we shall only concentrate on those differences relevant to the Mode-1/Mode-2 distinction. A generic effect axiom has the following form (see the exemplary axioms of [1] above):

$$[Q_i \wedge Q] \supset R(\eta_i, a)$$

a is an action term, η_i denotes an effect of (the denotation of) a, and $R(\eta_i, a)$ is a formula involving a and η_i. R may explicitly allude to an element of causality ([1] and [3]) or not ([11] and [2]). Shanahan's "Initiates" and "Terminates" indicate what Talmy refers to as *onset-causation* [12]—causation where the cause only *triggers* p; p persists

[1]The reasoning task of *explaining* observations is one in which the reasoner does not know all actions that took place. This is related, but not identical, to the issue that concerns us. Explanation is usually achieved by abduction. As we shall see below, we can adhere to mostly-monotonic deduction, without the need for the specialized machinery required by abduction. Alternatively, our approach may provide a way to *focus* abductive reasoning in the special case we are considering.

Reference	**Syntax**	**Examples**
[11]	$C(\mathsf{a}) = \{\eta_i\}$	$C(\mathsf{load}) = \{\mathsf{Loaded}\}$ $C(\mathsf{shoot}) = \{\neg\mathsf{Alive}\}$
[1]	$[Q_i \wedge Q] \supset \mathrm{Initiates}(a, \eta_i, s)$ $[Q_i \wedge Q] \supset$ $\quad \mathrm{Terminates}(a, \eta_i, s)$	$\mathrm{Initiates}(\mathrm{Load}, \mathrm{Loaded}, s)$ $\mathrm{HoldsIn}(\mathrm{Loaded}, s) \supset$ $\quad \mathrm{Terminates}(\mathrm{Shoot}, \mathrm{Alive}, s)$
[2]	$Q_i \supset \eta_i(do(a, s))$ $Q_i \supset \neg\eta_i(do(a, s))$	$\mathrm{Loaded}(do(\mathrm{Load}, s))$ $\mathrm{Loaded}(s) \supset \neg\mathrm{Alive}(do(\mathrm{Shoot}, s))$
[3]	$t + 1 : \eta_i \Leftarrow t : a \wedge Q_i$	$t + 1 : \mathit{Loaded} \Leftarrow t : \mathit{Load}$ $t + 1 : \neg\mathit{Alive} \Leftarrow t : \mathit{Shoot} \wedge \mathit{Loaded}$

Table 1. Examples of effect axioms.

by inertia. Giunchiglia et al's modal $\Leftarrow$ refers to Talmy's *extenet-durational causation* [12]—causation where the persistence of the cause is necessary for the persistence of the effect.[2]

The antecedent of our generic effect axioms, in general, contains two expressions: Q_i and Q. Both qualify the effect axiom but, given the subscript, Q_i somehow depends on η_i and Q depends only on a. The distinction is roughly the following: Q is a conjunction of *executability conditions*, required for a to be executable; Q_i is a conjunction of *effectiveness conditions*, required for a to yield the particular effect η_i.

Interestingly, this distinction does not seem to get the attention it deserves in many theories of action. In [11], for example, *all* qualifications are decoupled from effect axioms, enumerated (akin to the set $C(a)$ of consequences) as part of the action description. As a result, Ginsberg and Smith cannot specify that a loaded gun is needed for the shooting to be lethal, without requiring a loaded gun for the shooting to be possible in the first place.[3] On the other hand, both types of condition appear undistinguished in Shanahan's effect axioms [1]. Thus, (i) having a gun and (ii) the gun's being loaded would appear undistinguished in an effect axiom relating pulling the trigger to death. This makes one wonder about the semantics of action terms; for the absence of either condition will merely block the lethal effect of trigger-pulling. Now, whereas the non-lethal pulling of the trigger of an unloaded gun is a successful pulling of the trigger, no action qualifies as a pulling of the trigger in the absence of a gun.

Reiter [2] (but not Reiter in [13]) and Giunchiglia *et al.* [3] consistently recognize this distinction by providing separate "precondition axioms" specifying executability conditions; effectiveness conditions are conjoined in the antecedent of effect axioms. Now, even if the formalism allows for a distinction between executability and effectiveness (as in [2,3]), it is not clear how this feature will be exploited. In fact, it seems that the distinction between what should go into the effect axioms and what should be separately asserted is, to a big extent, arbitrary. If they do give definitions of qualification, authors typically give vague definitions under which several fine-grained notions are conflated. The following examples come from [13], in the context of database update.

[2]This interpretation of $\Leftarrow$ is only revealed by its use in domain constraints in solving the ramification problem.

[3]This might be Hanks and McDermott's original sin, but, for the Yale shooting scenario [10] to make the point it was intended for, everyone assumes that the gun could be successfully shot without being loaded (which is quite revealing as we shall see below). Readers who find this counter-intuitive may replace the act of shooting by that of pulling the trigger.

Example 1. It is possible to register a student in a course only if they have passed all pre-requisites.

Example 2. You can only change the grade of a student in a course to grade g if g is different from their current grade.

Example 3. A student may drop a course only if they are registered in it.

It is interesting to note that each of these examples illustrate a different type of qualification. The first is an example of *normative* qualifications; there is nothing physically impossible about registering a student in any course, but one *ought to* only register in courses for which they have all pre-requisites. The last is akin to an executability condition (but see Section 5.2); one cannot remove a student's name from a list if it is not already there. Each of these notions may be further divided into even more fine-grained ones, giving rise to an ontology of qualifications.[4]

The second example is the really interesting one. Why is it necessary for the new grade to be different from the old one? This is not a matter of executability: one can always delete the old grade and then insert it again. Nor is this a matter of normative conventions: there is nothing inappropriate about performing this vacuous update. Nor is it a matter of effectiveness for that matter; for what effect will be blocked if the two grades are identical? The only problem is that we cannot *categorize* the action as a *change* unless the new grade is different from the old one. This being said, we should now turn to event categorization.

3. The Event Categorization Problem

It is always said that there is a striking similarity between the ways we conceive of time and those in which we conceive of space, at least as revealed by language [14,15,16,12, for instance]. In particular, objects (denoted by count terms) correspond to events, and matter (denoted by mass terms) corresponds to states. I will take this fairly acceptable view as a basis for my distinction between what I call "events" and what I call "states".

One can view the conceptual difference between objects and matter as topological: objects are conceived of as topologically-closed and matter as topologically-open. What the distinction amounts to is that objects have their boundaries as parts; matter on the other hand, though always exists in the constitution of bounded objects, is not conceived of as having any boundaries as parts. On this view, any bounded amount of matter constitutes an object. Thus, a pile of sand, a lake, and a beam of light are objects, but sand, water, and light are only matter.

Similarly, events are (temporally) closed situations, ones that have their boundaries as parts, and states are open situations (also see [17]). This loosely corresponds to the linguistic distinction between bounded and unbounded sentences. For example, the imperfective (unbounded) sentence (1a) describes the street-crossing situation as a state, since the temporal boundary of the situation (i.e., its end) is not part of the description; as far as we can tell, the speaker might still be crossing the street. On the other hand, the perfective (bounded) (1b) describes the situation as an event, a bounded whole. Because

[4]For example, it is probably physically impossible for me to do a double somersault, given my fitness and lack of training. But maybe I can learn. However, I do not think I can ever learn to shoot a gun if I do not have one.

its temporal boundary is part of an event, event-sentences always imply that the reported situation has come to an end.[5]

(1) (a) I was crossing the street.
  (b) I crossed the street.

Now, as far as language is concerned, the space-time analogy is almost perfect. However, there is a certain respect in which it seems not to hold. In particular, the analogy fails in the way we actually *experience* time and space. In our everyday experience, we encounter objects; we see them, touch them, and (possibly) manipulate them in a, more or less, direct way. However, we rarely encounter matter per se; matter *typically* comes packaged as objects. Thus, we do not see "wood", "glass", or "paper"; we see chairs, bottles, and books.[6]

Our temporal experience, on the other hand, follows the exact opposite pattern. We never experience an event, a *whole* situation; no sooner have we reached the end of a situation, than its beginning has already moved into the past, beyond the reach of our conscious experience. Instead, the world continuously unfolds, presenting us with a continuous flux of states. Evidently, whatever is "now" the case is a state, never an event, for an event has its boundary as an essential part and, thus, can only exist in retrospect, when it has reached an end.[7]

But, if experience consists of only a cascade of states, where do events come from? Events are purely conceptual beasts; we conjure them up by conceptualizing a whole out of some state's starting to hold, holding for a while, and then ceasing. Logically, we must infer event occurrences from patterns of states. One might propose the following: an event has occurred if some state started to hold, held for a while, and then ceased. This is fine and good; the problem is that it only describes the occurrence of some *uncategorized* event token. An uncategorized event token is not very interesting, since one cannot derive any consequences of its occurrence, nor can one report its occurrence in any natural, informative way. The problem then is to infer, not only the occurrence of an event token, but also a categorization thereof. I shall call this *the event categorization problem.*

An acting agent inevitably faces the event categorization problem. It does so in the need to categorize its own acts—the primary example of Mode-2 reasoning.

4. Effect Axioms: Round II

The theories of action presented in Section 2 can happily accommodate agents doing Mode-1 reasoning. However, as they stand, the same theories are not prepared for Mode-2 reasoners. In particular, all effect axioms presented in Table 1 presume that we know what action took place. That is, in order for these axioms—which drive all inferences in an action theory—to be at all useful, the agent has to have categorized its own acts. Otherwise, the agent will not know which effect axioms are applicable.

[5]Note that by speaking of boundedness here I am not referring to *telicity*. The two notions are often conflated [18,19, for example], but several authors have distinguished them [20,21, for example]. Boundedness is a purely topological notion; telicity involves goal-directedness or, in general, some notion of a natural boundary. The role of telicity in my proposal will be discussed in detail in Section 5.2.

[6]Or we see *chunks* of wood, glass, and paper; but these are also objects, given their boundaries.

[7]This point has been made by Ismail [8, Ch. 3] and independently by Galton [22].

One might suggest that we can still proceed with existing Mode-1-oriented action theories if we are to give up Mode-2 reasoning all together, and tackle the event categorization problem without resorting to reasoning. In what follows, let $\mathbb{M}(c)$ be the motor program the agent executes whenever it decides to perform an instance of the act category denoted by the term c.[8] We can readily dismiss one obvious loophole:

> *If the agent starts to perform $\mathbb{M}(c)$, then, when it is done, it should believe that its action is of category c.*

The problem, of course, is that $\mathbb{M}(c)$ may fail. For example, in an attempt to shoot a gun, the agent's finger slips in the process of pulling the trigger. While someone might propose that this is indeed a shooting, albeit a failed one, it should be clear that this is just a linguistic trick: a failed shooting is as good a shooting as an alleged murderer is a murderer. Also clearly, if such an act is assumed to be a shooting, then we are committed to deriving all consequences of shooting.

A more realistic suggestion will be the following:

> *Each motor program should end with a sensing step, checking for certain conditions signalling success or failure. Whenever $\mathbb{M}(c)$ succeeds, then an instance of c has been performed. This information may be added to the agent's knowledge base by the motor program itself.*

While this may work for *primitive* act categories, which are directly grounded in motor programs through meta-theoretical association [9], it does not scale up to *high-level* act categories. The latter are associated with motor programs through reductions to primitive acts, expressed in statements of the object language. For example, suppose that pulling the trigger is primitive. It is not hard to construct a corresponding motor program that would sense whether the trigger indeed moved. However, if one advises the agent that it can shoot by pulling the trigger, then either

1. one will risk the agent's having misconceptions about what it did, since a successful pulling of the trigger need not be a successful shooting; or
2. one will have to change the motor program so that it also listens for a bang. (But then a pulling of the trigger would *only* be successful if a bang is heard.)

Both options are clearly unsatisfactory, and any other option will have to involve some sort of reasoning.[9] Further, this suggestion amounts to assuming that the agent can acquire beliefs about event occurrence through proprioception, which goes against our discussion in Section 3.

5. Two Important Distinctions

5.1. Necessary and Contingent Effects

Action theories in AI distinguish between direct and indirect effects of actions. Among the direct effects, we propose a distinction between those that are *necessary* and those

[8] I am, thus, assuming a first-order theory in which act categories are denoted by terms (similar to "Shoot" and "Load" in Table 1).

[9] Not to mention that sometimes it is not clear what condition to check for within the motor program.

that are *contingent*. For the agent to categorize an action as c, it has to check for all, and only, necessary effects of c. We can identify necessary effects with effects that are not qualified, and contingent effects with those that are. Qualified effects of actions are only achieved under some conditions. For example, shooting is only lethal if the gun is loaded (see Table 1). Unqualified effects, on the other hand, *necessarily* ensue: loading the gun unconditionally results in its being loaded.

Existing theories of action would, however, have counter-intuitive consequences should we apply the above definition of necessary effects without any provisions. Recall Reiter's Example 1 from Section 2. In Reiter's theory, the student's being registered in a course is not a necessary effect of registering, since the effect axiom is qualified. This is counter-intuitive; it seems that you can only categorize an action as one of *registering* if it results in the student's being *registered*. The trouble with this is probably rooted in the conflation of different types of qualification (see Section 2). In particular, passing all its pre-requisites is a normative qualification for registering in a course. Normative conditions should not be stated as qualifications in effect axioms; they should appear elsewhere in the theory (with special syntax) to indicate when an action is *permissible*.

Similarly, in Reiter's Example 2, it seems that once you've changed a grade to g, then the current grade's being g is an unconditional, necessary effect. The problem is that the old grade's being different is not really a qualification; rather, it has to be different for the performed action to be a *change* of grade.

Thus, we shall define necessary effects of an act category to be unqualified effects, provided that we do not include normative conditions, executability conditions, and the negation of the effect as qualifiers in the effect axiom. In what follows, let $\mathbb{NE}(c)$ denote the set of necessary effects of act category c.

5.2. Telic and Atelic Acts

Consider the following linguistic reports.

(2) (a) I ran.
(b) I ran to the store.
(c) I ran toward the store.
(d) I ran past the store.

For our acting agent to be able to honestly report (2a), it only needs to have been running for a while. Thus, if the state of running ever held, an event of running did occur. This property is characteristic of event categories commonly referred to in the literature as *atelic* [15, for instance]. For the *telic* category reported by (2b), (i) the agent has to have run, (ii) the running has to have stopped at the store, and (iii) the running has to have caused the agent to be at the store. Thus, in order for the agent to form a belief along the lines of (2b), it needs to monitor what it is doing, making sure that it has culminated in a certain way, and to involve some sort of causal reasoning. Clearly, a final state in which a telic event naturally culminates is a necessary effect.

Beyond the standard telic/atelic distinction, [8] points out event categories with different occurrence conditions. Sentence (2c) reports an occurrence of a *left-atelic* event category: similar to (2b), there *is* a natural final state (being at the store), but it should not be reached. (2d) is a report of a *right-atelic* event category: a necessary effect has to be achieved, but the agent's activity may continue beyond that achievement.[10] For limitations of space, I cannot embark on a detailed analysis of this ontology of event categories. A formal treatment may be found in [8]. I will henceforth stick to the tradition, and consider only telic and atelic categories.

Now, let us take some time to discuss what we mean by telicity.[11] The above discussion, like traditional discussions of telicity [23,24, for example], gives the impression that telicity is merely about a natural *end* point. However, several authors [15, for example] have pointed out that telicity is not just a matter of states that hold in the aftermath of an event. This is particularly true for examples like Example 2 of Section 2, where changing a grade (a telic event) can only occur if it effects a transition from an initial, pre-action state to a final, post-action state.

A detailed analysis of telicity may be found in [15]. However, for the sake of the relatively modest objective of this paper, identifying occurrence conditions of telic event categories need not require such an ambitious analysis. In particular, I will take telic events to be effecting a transition from an initial state s_i, that holds at the onset of the event, to a terminal state s_t that is caused by it. In typical examples of telicity, s_i and s_t are contradictory states. For example, in (2b), s_i is the state of not being at the store and s_t is being at the store. Thus, not only should s_t hold at the end of the event, but it should also *not* hold immediately before and, crucially, throughout the event occurrence.

In other cases, however, s_i and s_t need only be contrary states. In this case, s_i implies, but is not necessarily identical to, the negation of s_t. For example, consider (3), where it is not sufficient for me end up at the store. In addition, I have to have started running at the park, which is contrary, but not contradictory, to my initially being at the store.

(3) I ran from the park to the store

In general, however, s_i and s_t could be any two states. In fact, the two states may be identical, as in the example of running around a track. But note that in all cases, through out the event occurrence, s_t does not hold. If it does, then we have a case, not of telicity, but of right-atelicity. Whenever s_i is not indicated by the event description, it seems that it is always taken to be the negation of s_t—the weakest contrary-to-s_t state.

Thus, similar to the set $\mathbb{NE}(c)$ of necessary effects of event category c, there is a set $\mathbb{NI}(c)$ of *necessary initial conditions*, containing all the s_is. In most cases, the conditions in $\mathbb{NI}(c)$ are the negations of those in $\mathbb{NE}(c)$. Conditions in $\mathbb{NI}(c)$ are somewhat similar to executability conditions in that they (i) are implied by the occurrence of an instance of c and (ii) need to be achieved by the agent before attempting an execution of c. The two types of condition are different in that necessary effects *cannot* be achieved if executability conditionas are not satisfied; they could be achieved, however, even if initial conditions do not hold (witness (3)). One could say that executability conditions are conditions on motor programs, whereas necessary initial conditions are conditions on event descriptions.

[10]We can further categorize telic acts into those with immediate effects and those with delayed effects (see [8]).

[11]This discussion was motivated by the criticisms levelled by an anonymous reviewer of FOIS-08 at a draft of this paper.

6. A Unified Framework

In this section, a framework for mode-2 reasoning is presented. I should stress that what will be presented is just a *framework*, not a complete axiomatic system. A complete system needs more space for presentation, and more time for working out all the details. The purpose of the framework is to provide a starting point, and to illustrate the rich epistemological ontology needed for a mode-2 reasoner (as opposed to the relatively coarse ontology of mode-1 reasoning systems).

6.1. Formal Machinery

We shall need a formal language to talk about states, events, and event categories. Only informal semantics will be provided, however. In my so doing, I trust that the reader will not end up in confusion and that nothing is at stake, regarding the point I am trying to make. In addition to a formal semantics, axioms ruling out unintended models will also be needed should this language be actually used for reasoning.

The purpose of the language should also be clear. Statements of the language represent beliefs of a reasoning and acting agent; they do not represent a third-person perspective of what the agent is doing, was doing, or has done. Thus, no terms denoting agents necessarily appear in the language; only a constant I is necessary, denoting the reasoning agent's self-concept [9]. Moreover, statements representing beliefs about what the agent *is doing* are not derived as a result of, say, plan recognition. Beliefs about what it is doing come form the agent's first-person access to its own intentions and bodily feedback (see [5,6] and particularly [4, p. 23]).

The language is a first-order language akin to that of [25], which is a revised version of Allen's interval-based theory [7], where instants are independently included in the ontology. The ontology comprises ordinary objects and individuals, time instants, time intervals, states, event categories, and event tokens. The following predicate symbols are part of our language. (Superscripts indicate adicity.)

- $\sqsubset^2$: $[\![t_1 \sqsubset t_2]\!]$ is true whenever interval $[\![t_1]\!]$ is a proper subinterval of interval $[\![t_2]\!]$
- $\prec^2$: $[\![t_1 \prec t_2]\!]$ is true whenever time $[\![t_1]\!]$ wholly precedes time $[\![t_2]\!]$
- Begins^2: $[\![\mathsf{Begins}(i, t)]\!]$ is true whenever instant $[\![i]\!]$ limits interval $[\![t]\!]$ at its beginning.
- Ends^2: $[\![\mathsf{Ends}(i, t)]\!]$ is true whenever instant $[\![i]\!]$ limits interval $[\![t]\!]$ at its end.
- Within^2: $[\![\mathsf{Ends}(i, t)]\!]$ is true whenever instant $[\![i]\!]$ is within interval $[\![t]\!]$.[12]
- $\mathsf{HoldsAt}^2$: $[\![\mathsf{HoldsAt}(s, i)]\!]$ is true whenever state $[\![s]\!]$ holds at instant $[\![i]\!]$.
- Occurs^2: $[\![\mathsf{Occurs}(e, t)]\!]$ is true whenever event token $[\![e]\!]$ occurs on time $[\![t]\!]$.[13]
- Cat^2: $[\![\mathsf{Cat}(e, c)]\!]$ is true whenever event token $[\![e]\!]$ is an instance of event category $[\![c]\!]$
- Caused^2: $[\![\mathsf{Caused}(e_1, e_2)]\!]$ is true whenever event token $[\![e_1]\!]$ caused event token $[\![e_2]\!]$.

For the sake of brevity, we define the following convenient predicates:

[12] An axiomatization of Within may be found in [25].

[13] For simplicity, I am overloading Occurs to work for both durative and instantaneous events; [25] uses two different predicate symbols.

- $t_1 \supset\!\subset t_2 =_{\mathrm{def}} t_1 \prec t_2 \land \neg\exists t_3[t_1 \prec t_3 \land t_3 \prec t_2]$[14]
- $\mathsf{Holds}(s,t) =_{\mathrm{def}} \forall i[\mathsf{Within}(i,t) \supset \mathsf{HoldsAt}(s,i)]$
- $\mathsf{MHolds}(s,t_1) =_{\mathrm{def}} \mathsf{Holds}(s,t_1) \land \neg\exists t_2[\mathsf{Holds}(s,t_2) \land t_1 \sqsubset t_2]$

In addition, we have five function symbols:

- $\uparrow^1$: $[\![\uparrow s]\!]$ is the event category of onsets of state $[\![s]\!]$.
- $\downarrow^1$: $[\![s \downarrow]\!]$ is the event category of cessations of state $[\![s]\!]$.
- $\dot{\neg}^1$: $[\![\dot{\neg}\, s]\!]$ is the unique state that holds at every instant at which $[\![s]\!]$ does not hold.
- Prog^1: $[\![\mathsf{Prog}(c)]\!]$ is the unique state that holds whenever event category $[\![c]\!]$ is in progress
- Clos^2: $[\![\mathsf{Clos}(s,t)]\!]$ is the event token of state $[\![s]\!]$ maximally holding throughout time $[\![t]\!]$

Clos is closely related to MHolds:

AMC. $\mathsf{Occurs}(\mathsf{Clos}(s,t),t') \;\equiv\; (t' = t) \land \mathsf{MHolds}(s,t)$

Thus, every time a state maximally holds, the agent may easily infer the occurrence of some event, namely the closure of that state at the said time. As pointed out in Section 3, such events are not very useful because they do not fall under any natural category.[15]

The semantics of the progressive operator is, no doubt, mysterious. However, I will content myself with the informal gloss given above, and refer the reader to [27]. Nevertheless, note that I am mainly concerned with acting agents, for which progressive states are primarily experienced while acting. Knowledge of such states is a first-person privilege that is investigated in depth in the philosophy of action [4,5,6]. For how such knowledge may be induced in the case of a robot, see [9]. Essentially, $\mathsf{Prog}(c)$ holds whenever the agent is executing $\mathbb{M}(c)$, for primitive acts, or a plan for performing c, for high-level acts. I will, hence, take it as unproblematic that agents can form beliefs such as $\mathsf{Holds}(\mathsf{Prog}(\mathsf{Run}(\mathsf{I})),t)$ or $\mathsf{Holds}(\mathsf{Prog}(\mathsf{RunTo}(\mathsf{I},\mathsf{Store})),t)$. I shall also restrict Prog by the following axiom.

AP. $[\mathsf{Occurs}(e,t) \land \mathsf{Cat}(e,c)] \supset \mathsf{MHolds}(\mathsf{Prog}(c),t)$

This axiom licences inference from statements like (1b) to statements like (1a). Thus, I am not assuming that a notion of intention is an essential ingredient in the retrospective use of Prog.

It should be pointed out, however, that **AP** is not appropriate for all classes of event categories. In particular, it does not hold for categories that are *indefinitely-specified* [28]. Indefinitely-specified event categories are ones that involve an indefinite entity. For example, the category of (bomb) explosions seems to involve an indefinite bomb, the category of concerts involves an indefinite performer, and the category of my picking up *a* block involves an indefinite block. Indefinitely-specified event categories do not conform to **AP** since multiple tokens of them may overlap in time. Consider the category pick-up-a-block of picking up an indefinite block.[16] Suppose there are two blocks: A

[14]The reader will note that $\supset\!\subset$ is Allen's [7] `meets`. The symbol "$\supset\!\subset$" is the one used for the same relation in discourse representation theory [26].

[15]If we adopt Galton's Po operator [27], then $\mathsf{Clos}(s,t)$ is of category $Po(s)$.

[16]See [28] for a more elaborate characterization of the syntactic structure of terms denoting indefinitely-specified event categories.

and B. I start picking up A with my right hand at time instant i_1. While still in the process of picking up A, I start picking up B with my left hand at time instant i_2. Picking up A ends at i_3, and picking up B ends, later, at i_4. Clearly, the categories of picking up A and picking up B satisfy **AP**. However, pick-up-a-block does not. The reason is that, whereas a token of pick-up-a-block does occur over the interval extending from i_1 to i_3, Prog(pick-up-a-block) does not *maximally* hold over this interval. Rather, it maximally holds over the interval extending from i_1 to i_4.

The above notwithstanding, for the purpose of this paper, I will consider only definitely-specified event categories. In banishing indefinitely-specified categories, I am adopting the common tacit (and possibly sub-conscious) policy of most authors. Nevertheless, a careful examination of indefinitely-specified event categories is doubtlessly called for. This is particularly pressing since this class of event categories seems to challenge many of our intuitions about events.[17]

6.2. *Occurrence Conditions*

The occurrence conditions of an event category will be represented by statements of the following form, where ϕ is the condition.

- $\exists e[\mathsf{Occurs}(e,t) \wedge \mathsf{Cat}(e,c)] \equiv \phi$

The exact form of ϕ depends on the type of category we are considering. First, events are either instantaneous or durative. I will take instantaneous events to be onsets or cessations of states. To infer the occurrence of an onset, the agent should experience a state not holding, followed by an experience of the same state holding. Similarly for cessations.[18] [19]

AO. $\exists e[\mathsf{Occurs}(e,i) \wedge \mathsf{Cat}(e, \uparrow s)] \equiv$
$$\exists t_1, t_2[\mathsf{Holds}(\dot\neg\, s, t_1) \wedge \mathsf{Holds}(s, t_2) \wedge \mathsf{Ends}(i, t_1) \wedge \mathsf{Begins}(i, t_2)]$$
AC. $\exists e[\mathsf{Occurs}(e,i) \wedge \mathsf{Cat}(e, s \downarrow)] \equiv$
$$\exists t_1, t_2[\mathsf{Holds}(s, t_1) \wedge \mathsf{Holds}(\dot\neg\, s, t_2) \wedge \mathsf{Ends}(i, t_1) \wedge \mathsf{Begins}(i, t_2)]$$

Durative events are those described in Section 3 as comprising a state starting to hold, holding for a while, and then ceasing. Inferring the occurrence of such events is not as simple as in the case of instantaneous events; it depends on whether the event is telic or atelic.

For atelic categories, we have the following general condition:[20]

AA. $\mathsf{Cat}(\mathsf{Clos}(\mathsf{Prog}(c), t), c)$

Thus, the closure of the progressive state of an atelic event category is always an instance of it. Using **(AMC)**, (**AP**), and **AA**, it is easy to show that

---

[17] See [28] for examples of such intuitions.

[18] Events such as winking or hiccupping, which are usually considered punctual ([24]), are not instantaneous on my view.

[19] Since states need to be *experienced* in order for an agent to infer the occurrence of their onsets and cessations, such states cannot hold at isolated instants. This assumption is made explicit in axioms **A0** and **A1** below. In the terminology of [25], these states are "states of motion". States that can hold at isolated instants (Galton's "states of position") cannot be experienced; similar to events, their holding at an instant has to be inferred.

[20] Once again, this condition does not apply to indefinitely-specified event categories.

TA. $\exists e[\mathsf{Occurs}(e,t) \wedge \mathsf{Cat}(e,c)] \equiv \mathsf{MHolds}(\mathsf{Prog}(c),t)$

For telic events, the situation is more complex:

AT1. $$[\mathsf{MHolds}(\mathsf{Prog}(c),t_1) \ \wedge \bigwedge_{s\in\mathbb{NE}(c)} \mathsf{Holds}(\dot{\neg}\ s,t_1) \wedge \\ \bigwedge_{s\in\mathbb{NI}(c)} \exists t_2[t_2 \supset\!\subset t_1 \ \wedge\ \mathsf{Holds}(s,t_2)] \wedge \\ \bigwedge_{s\in\mathbb{NE}(c)} \exists t_3[t_1 \supset\!\subset t_3 \ \wedge\ \mathsf{Holds}(s,t_3) \wedge \ \mathsf{Caused}(\mathsf{Clos}(\mathsf{Prog}(c),t_1),\mathsf{Clos}(s,t_3))]] \\ \supset \mathsf{Cat}(\mathsf{Clos}(\mathsf{Prog}(c),t_1),c)$$

That is, whatever the agent did on t_1 is an instance of c if, as far as it knows, the necessary initial conditions held as it started c-ing, it was c-ing throughout t_1, the necessary effects of c did not hold throughout t_1 but started as the agent's activity halted, and the performance of that activity is what caused the necessary effects. I believe that the biggest challenge in all this is the final causal link; ultimately, a full analysis of causation along the line of [29,30, for example] is needed. However, this might be a situation in which defeasibly inferring this causal link would be appropriate.[21]

We can now state the following stronger condition on telic event occurrence.

AT2. $$\exists e[\mathsf{Occurs}(e,t_1) \wedge \mathsf{Cat}(e,c)] \equiv \\ [\mathsf{MHolds}(\mathsf{Prog}(c),t_1) \ \wedge \bigwedge_{s\in\mathbb{NE}(c)} \mathsf{Holds}(\dot{\neg}\ s,t_1) \wedge \\ \bigwedge_{s\in\mathbb{NI}(c)} \exists t_2[t_2 \supset\!\subset t_1 \ \wedge\ \mathsf{Holds}(s,t_2)] \wedge \\ \bigwedge_{s\in\mathbb{NE}(c)} \exists t_3[t_1 \supset\!\subset t_3 \ \wedge\ \mathsf{Holds}(s,t_3) \wedge \ \mathsf{Caused}(\mathsf{Clos}(\mathsf{Prog}(c),t_1),\mathsf{Clos}(s,t_3))]]$$

The right-to-left direction follows from (**AT1**) and (**AMC**). The left-to-right direction could be replaced by a set of conditionals. The first, with $\mathsf{MHolds}(\mathsf{Prog}(c),t_1)$ as a consequent, is just (**AP**). The rest correspond to unqualified effect axioms and axioms for necessary initial conditions.[22]

The occurrence condition for the act of changing student a's grade to g (see Example 2 in Section 2) may be stated as follows (assuming pretend-it-is-English semantics).

$$\exists e[\mathsf{Occurs}(e,t_1) \wedge \mathsf{Cat}(e,\mathsf{Change}(\mathsf{grade}(a),g))] \equiv \\ [\mathsf{MHolds}(\mathsf{Prog}(\mathsf{Change}(\mathsf{grade}(a),g)),t_1) \ \wedge \ \mathsf{Holds}(\dot{\neg}\ (\mathsf{grade}(a)=g),t_1) \wedge \\ \exists t_2[t_2 \supset\!\subset t_1 \ \wedge\ \mathsf{Holds}(\dot{\neg}\ (\mathsf{grade}(a)=g),t_2)] \wedge \\ \exists t_3[t_1 \supset\!\subset t_3 \ \wedge\ \mathsf{Holds}(\mathsf{grade}(a)=g,t_1),t_3) \wedge \\ \mathsf{Caused}(\mathsf{Clos}(\mathsf{Prog}(c),t_1),\mathsf{Clos}(\mathsf{grade}(a)=g,t_1),t_3))]]$$

---

[21] On another note, we can assert the causal relation, not between the closures of states, but from the closure of the progressive state to the *onset* of necessary effects. This would be closer to the spirit of Shanahan's "Initiates" [1]. This move will actually be required if we are to explicitly dismiss unbounded intervals from our ontology. For, in that case, necessary effects could not possibly be permanent.

[22] If these axioms are part of our theory, (**AT2**) would be a theorem.

Using the above occurrence conditions, an agent can categorize its own acts. It only needs to experience the necessary effects, but all contingent effects and ramifications may be inferred through the standard axioms. Note that whether **(AT2)** or **(TA)** is applicable depends on the teleological features of c. Knowledge of telicity may be explicitly given to the agent, by asserting of durative event categories whether they are telic or atelic. However, it may also be derived through a fine-grained analysis of telicity in the spirit of [15].

7. Conclusions

By distinguishing necessary and contingent effects, and telic and atelic act categories, we presented a language which seems to be expressive enough to serve as a tool for both Mode-1 and Mode-2 reasoning. The underlying ontology makes distinctions that have long been recognized as necessary for natural language semantics [27,24,15].

What has been presented is by no means a theory, it is a framework within which more investigations should follow. The following are some of the issues of which this paper could only scratch the surface.

1. What is a precondition? We distinguished executabiltiy conditions, normative qualifications, and effectiveness conditions. However, we also noted a strong relation between executability conditions and the necessary initial conditions appearing in general occurrence axioms. Where do these fit? Are there other types of preconditions?
2. How rich is the notion of telicity? We distinguished telic and atelic events, but we noted the existence of other classes (left-atelic and right-atelic, telic with immediate or delayed effects, etc.). A careful examination of telicity as a fundamental notion for event occurrence is called for.
3. What is the effect, on the presented system, of admitting indefinitely-specified events? As noted in the paper, at least two of our fundamental axioms will no longer be valid.

To conclude, here are some remarks on constructing formal theories that accommodate both Mode-1 and Mode-2 reasoners:

1. Executability conditions and normative qualifications should be separated from effect axioms.
2. The negation of an effect should not appear as a qualification on the same effect. Rather, it is a *necessary* initial condition, akin to executability conditions.
3. Necessary effects of telic categories should appear only in occurrence conditions; the corresponding effect axioms logically follow.

Alternatively, given a standard Mode-1 action theory that observes the first two remarks, we can always generate occurrence conditions by realizing that telic acts are those with unqualified effects. Such effects are the necessary effects needed for occurrence conditions.

References

[1] Murray Shanahan. *Solving the Frame Problem: A Mathematical Investigation of the Common Sense Law of Inertia*. MIT Press, Cambridge, MA, 1997.
[2] Raymond Reiter. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems.* MIT Press, Cambridge, MA, 2001.
[3] E. Giunchiglia, J. Lee, V. Lifschitz, N. McCain, and H. Turner. Nonmonotonic causal theories. *Artificial Intelligence*, 153:49–104, 2004.
[4] Kevin Falvey. Knowledge in intention. *Philosophical Studies*, 99:21–44, 2000.
[5] George Wilson. Proximal practical foresight. *Philosophical Studies*, 99:3–19, 2000.
[6] David Velleman. *The Possibility of Practical Reason*. Oxford University Press, Oxford, 2000.
[7] James Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.
[8] Haythem O. Ismail. *Reasoning and Acting in Time*. PhD thesis, University at Buffalo, The State University of New York, 2001.
[9] Stuart C. Shapiro and Haythem O. Ismail. Anchoring in a grounded layered architecture with integrated reasoning. *Robotics and Autonomous Systems*, 43(2-3):97–108, 2003.
[10] Steve Hanks and Drew McDermott. Nonmonotonic logic and temporal projection. *Artificial Intelligence*, 33:379–412, 1987.
[11] Matthew Ginsberg and David Smith. Reasoning about action II: The qualification problem. *Artificial Intelligence*, 35(3):311–342, 1988.
[12] Leonard Talmy. *Toward a Cognitive Semantics*. MIT Press, 2000.
[13] Raymond Reiter. Proving properties of states in the situation calculus. *Artificial Intelligence*, 64(2):337–351, 1993.
[14] Emmon Bach. The algebra of events. *Linguistics and Philosophy*, 9:5–16, 1986.
[15] Manfred Krifka. The origins of telicity. In Susan Rothstein, editor, *Events and Grammar*, pages 197–235. Kluwer Academic Publishers, Dordrecht, 1998.
[16] Ray Jackendoff. Parts and boundaries. *Cognition*, 41:9–45, 1991.
[17] Fabio Pianesi and Achille Varzi. Events, topology, and temporal relations. *The Monist*, 79(1):89–116, 1996.
[18] Renaat Declerck. Aspect and the bounded/unbounded (telic/atelic) distinction. *Linguistics*, 17:761–794, 1979.
[19] Östen Dahl. On the definition of the telic-atelic (bounded-nonbounded) distinction. In Philip Tedeschi and Annie Zaenen, editors, *Tense and Aspect*, volume 14 of *Syntax and Semantics*, pages 79–90. Academic Press, New York, 1981.
[20] Ilse Depraetere. On the necessity of distinguishing between (un)boundedness and (a)telicity. *Linguistics and Philosophy*, 18:1–19, 1995.
[21] Jean-Pierre Koenig and Nuttanart Muansuwan. How to end without ever finishing: Thai semi-perfective markers. *Journal of Semantics*, 17:147–184, 2000.
[22] Antony Galton. On what goes on: The ontology of processes and events. In Brandon Bennett and Christiane Fellbaum, editors, *Formal Ontology in Information Systems: Proceedings of the Fourth International Conference (FOIS 2006)*, pages 4–11, Baltimore, 2006. IOS Press.
[23] Zeno Vendler. Verbs and times. *The Philosophical Review*, 66(2):143–160, 1957.
[24] M. Moens and M. Steedman. Temporal ontology and temporal reference. *Computational Linguistics*, 14:15–28, 1988.
[25] Antony Galton. A critical examination of Allen's theory of action and time. *Artificial Intelligence*, 42:159–188, 1990.
[26] Hans Kamp and Uwe Reyle. *From Discourse to Logic*. Kluwer Academic, Dordrecht, 1993.
[27] Antony Galton. *The Logic of Aspect*. Clarendon Press, Oxford, 1984.
[28] Haythem O. Ismail. Simultaneous events and the "once-only" effect. In Brandon Bennett and Christiane Fellbaum, editors, *Formal Ontology in Information Systems: Proceedings of the Fourth International Conference (FOIS 2006)*, pages 143–154, Baltimore, 2006. IOS Press.
[29] David Lewis. Causation. *The Journal of Philosophy*, 70(17):556–567, 1973.
[30] Charles Ortiz. Explanatory update theory: Applications of counterfactual reasoning to causation. *Artificial Intelligence*, 108(1–2):125–178, 1999.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-237

Introducing Realist Ontology for the Representation of Adverse Events

Werner CEUSTERS [a,1], Maria CAPOLUPO [b], Georges DE MOOR [c], Jos DEVLIES [c]
[a] *New York State Center of Excellence in Bioinformatics & Life Sciences, University at Buffalo, NY, USA*
[b] *Independent consultant, Buffalo NY, USA*
[c] *Research in Advanced Medical Informatics and Telematics VZW, Ghent, Belgium.*

Abstract. The goal of the REMINE project is to build a high performance prediction, detection and monitoring platform for managing Risks against Patient Safety (RAPS). Part of the work involves developing an ontology enabling computer-assisted RAPS decision support on the basis of the disease history of a patient as documented in a hospital information system. A requirement of the ontology is to contain a representation for what is commonly referred to by the term '*adverse event*', one challenge being that distinct authoritative sources define this term in different and context-dependent ways. The presence of some common ground in all definitions is, however, obvious. Using the analytical principles underlying Basic Formal Ontology and Referent Tracking, both developed in the tradition of philosophical realism, we propose a formal representation of this common ground which combines a reference ontology consisting exclusively of representations of universals and an application ontology which consists representations of defined classes. We argue that what in most cases is referred to by means of the term '*adverse event*' – when used generically – is a *defined class* rather than a *universal*. In favour of the conception of adverse events as forming a defined class are the arguments that (1) there is no definition for '*adverse event*' that carves out a collection of particulars which constitutes the extension of a universal, and (2) the majority of definitions require adverse events to be (variably) the result of some observation, assessment or (absence of) expectation, thereby giving these entities a nominal or epistemological flavour.

Keywords. Basic Formal Ontology, Referent Tracking, adverse events, patient safety.

1. Introduction

'*High performance prediction, detection and monitoring platform for patient safety risk management (REMINE)*' is the name of a European Large Scale Integrating Project (IP) which has been funded by the European Commission since January 1, 2008 [1]. The main objective is to develop a technological platform and identify best practice business processes allowing automated management and prevention of Risks against Patient Safety (RAPS).

Part of the work to be carried out consists of the development of an ontology that will support several functionalities offered by the envisioned technological platform. In this

[1] Corresponding author: Werner Ceusters, Ontology Research Group, NYS Center of Excellence in Bioinformatics & Life Sciences, 701 Ellicott street, suite B2-160, Buffalo NY 14203, USA.

Table 1. Adverse event related definitions from authoritative sources

ID	Term	Definition	Source	Ref.
D1	***adverse drug event*** (adverse drug error)	Any incident in which the use of a medication (drug or biologic) at any dose, a medical device, or a special nutritional product (for example, dietary supplement, infant formula, medical food) may have resulted in an adverse outcome in a patient.	JTC	[2]
D2	***adverse drug experience***	any adverse event associated with the use of a drug in humans, whether or not considered drug related, including the following: • an adverse event occurring in the course of the use of a drug product in professional practice; • an adverse event occurring from drug overdose whether accidental or intentional; • an adverse event occurring from drug abuse; • an adverse event occurring from drug withdrawal; and • any failure of expected pharmacological action.	FDA	[3]
D3	***adverse drug reaction***	an undesirable response associated with use of a drug that either compromises therapeutic efficacy, enhances toxicity, or both.	JTC	[2]
D4	***adverse event***	an ***observation*** of a change in the state of a subject ***assessed*** as being untoward by one or more ***interested parties*** within the ***context*** of a protocol-driven research or public health.	BRIDG	[4]
D5	***adverse event***	an event that results in unintended harm to the patient by an act of commission or omission rather than by the underlying disease or condition of the patient	IOM	[5]
D6	***adverse event***	any unfavourable and unintended sign (including an abnormal laboratory finding), symptom, or disease temporally associated with the use of a medical treatment or procedure that may or may not be considered related to the medical treatment or procedure	NCI	[6]
D7	***adverse event***	any untoward medical occurrence in a patient or clinical investigation subject administered a pharmaceutical product and which does not necessarily have to have a causal relationship with this treatment	CDISC	[7]
D8	***adverse event***	an untoward, undesirable, and usually unanticipated event, such as death of a patient, an employee, or a visitor in a health care organization. Incidents such as patient falls or improper administration of medications are also considered adverse events even if there is no permanent effect on the patient.	JTC	[2]
D9	***adverse event***	an injury that was caused by medical management and that results in measurable disability.	QUIC	[8]
D10	***error of omission***	An error which occurs as a result of an action not taken. Errors of omission may or may not lead to adverse outcomes.	JTC	[2]
D11	***observation***	an act of recognizing and noting a fact or an occurrence of an event of interest. An observation may involve examination, interviews, or measurement with devices. Observations are not intended to alter the state of the subject.	BRIDG	[4]
D12	***serious adverse drug experience***	Any adverse drug experience occurring at any dose that results in any of the following outcomes: • death • a life-threatening adverse drug experience • inpatient hospitalization • prolongation of existing hospitalization • a persistent or significant disability/incapacity • a congenital anomaly/birth defect	FDA	[3] [9]

paper, we focus on one particular RAPS issue, namely the ontological treatment of *adverse events*.

The term '*adverse event*' is defined in the literature in a variety of ways, superordinate terms frequently used being '*reaction'*, '*effect'*, '*event'*, '*problem'*, '*experience'*, '*injury'*, '*symptom'*, '*illness'*, '*occurrence'*, '*change'*, and even '*something'*, '*act'*, '*observation'* and '*term*', the latter four being the result of applying flawed terminological theories which rest on a confusion between an entity and an observation or record thereof [10]. This multitude of definitions is brought about by the many organisations and initiatives that have set themselves the noble goal of reducing the occurrence of adverse events, especially since the year 2000, when the Institute of Medicine published its report *To Err is Human: Building a Safer Health System* [11]. **Table 1** contains a small selection of adverse event definitions by authoritative sources, drawn from a larger collection that we composed for our work in [12].

Research aimed at bringing about some order in this domain falls into three categories. One is classification, as witnessed by the work of Chang *et al.* who developed – on the basis of a set of criteria specified in [13] – a classification schema consisting of five root nodes which they found to be the '*homogeneous elements*' encountered in relevant sources: ***Impact***, ***Type, Domain, Cause*** and ***Prevention and Mitigation*** [14]. Others, such as the BRIDG consortium, have tried to resolve the multitude of definitions by reaching consensus on just one [4], with the result of being extremely reductionist. A third group of researchers has focused on building ontologies. Unfortunately, this latter group has typically employed the rather weak principles underlying the '*concept*'-orientation [15] in ontology development, so that, for example, '*age*' and '*gender*' become a subclass of '*patient'* [16]. We nonetheless believe that ontology is indeed the right approach to take in addressing this difficult and important problem, but, in contrast to what is still the majority view among ontologists, we believe that ontology will bring benefits only when rigorous principles are applied, principles that go far beyond the basic requirement of computational soundness.

2. Objective and design

Our goal is to bring clarity to the terminological wilderness that grew out of all the efforts documented in [12]. Problems arise not only because of differences amongst initiatives in terms of scope, health care settings involved, jurisdictions, and objectives – the consequence being that definitions resulting from such efforts are not applicable outside the original boundaries – but also because of a widespread failure to adopt sound ontological and terminological principles in analysing and conveying what is relevant. As an example, a definition such as " *'Adverse outcome' should be understood to mean not only a non-trivial adverse outcome* […] *but also an incident* […] *which results in a recognized potential risk of a non-trivial adverse outcome* […]" [17] (irrelevant detail omitted), is of the form '*an X is an X or a Y which leads to an X*' and is thus at best uninformative.

To obtain our goal, we analysed the literature and collected all relevant definitions and descriptions that we found. We modified some of these definitions slightly in order to have them convey better what we judged to be the intended message, thereby still keeping track of the original versions in order to identify general principles for improved definition construction in the domain of patient safety.

We also studied a variety of classification systems, taxonomies, terminologies and concept-based ontologies – we use the term '*concept-based ontologies*' to differentiate representational artifacts created on the basis of the concept orientation from the realism-based ontology that is being developed under REMINE – in order to obtain a comprehensive list of entity types whose nature and interrelationships are to be studied and formally represented to satisfy requirements of the project.

3. Methodology

We performed our analysis following the principles advocated in Basic Formal Ontology and Referent Tracking.

3.1. Basic Formal Ontology

Basic Formal Ontology (BFO) is a framework encapsulating best practices in ontology development that is designed to serve as basis for the creation of high-quality shared ontologies in the biomedical domain [18, 19].

BFO has a *realist* orientation based on the view that terminologies and ontologies are to be aligned not on '*concepts*' but rather on entities in reality [15]. Central to this view are three assumptions.

The first is that biological reality exists objectively in and of itself, i.e. independently of the perceptions or beliefs or theories of cognitive beings. Thus not only do a wide variety of entities exist in reality (human beings, hearts, bacteria, disorders, ...), but also how these entities relate to each other – that certain hearts are parts of human beings, that certain bacteria cause disorders in human beings – is not a matter of agreements made by scientists but rather of objective fact.

The second assumption is that reality is accessible to us and that its structure can be discovered: it is scientific research that allows human beings to find out what entities exist and what relationships obtain between them.

The third assumption is that an ontology should mirror its corresponding domain of reality. Thus an important aspect of the quality of an ontology is determined by the degree to which not merely its individual representational units correspond to entities in reality but also the structure according to which these units are organized mimics the corresponding structure of reality. Realism-based ontology development was introduced into biomedical informatics some fifteen years ago as a means of detecting and avoiding the systematic mistakes characteristic of concept-based terminologies [20-24], mistakes which are not eliminated through the use of description logics or similar computational devices [25].

BFO acknowledges only those entities which exist in reality, and rejects all types of putative entities postulated merely as artifacts of specific logical or computational frameworks. The corresponding logical and computational artifacts themselves, however, are indeed accepted as part of reality. BFO captures a small number of basic categories into which the entities in reality are divided, thereby distinguishing, at the highest level of its organisation:

(1) *particulars* such as ***Werner Ceusters*** from *universals* such as HUMAN BEING,[2]

(2) *continuants* such as ***Werner Ceusters' heart*** from *occurrents* such as ***the beating of Werner Ceusters' heart***,

(3) *independent entities* such as ***Werner Ceusters' heart*** from *dependent entities* such as ***the function of Werner Ceusters' heart***.

Dependent entities are such that they cannot exist – in the ontological rather than biological sense – without some instance of the category independent entity. BFO also distinguishes three major families of relations between entities in the categories just distinguished:

(1) *<p, p>-relations*: from particular to particular (for example: ***Werner Ceusters' s brain*** being part of ***Werner Ceusters***);

(2) *<p, u>-relations*: from particular to universal (for example: ***Werner Ceusters*** being an instance of HUMAN BEING);

(3) *<u, u>-relations*: from universal to universal (for example: HUMAN BEING being a subkind of ORGANISM) [26].

3.2. Referent Tracking

Referent tracking has been introduced as a new paradigm for entry and retrieval of data in the Electronic Health Record (EHR) to avoid the multiple ambiguities that arise when statements in an EHR refer to disorders, lesions and other entities on the side of the patient exclusively by means of generic terms from a terminology or ontology [27]. Referent tracking avoids such ambiguities by introducing *IUIs* – Instance Unique Identifiers – for each numerically distinct entity that exists in reality and that is referred to in statements in a record. As ontologies serve integration of information at the level of the types (universals and defined classes) which particulars instantiate, so referent tracking serves integration of information at the level of these particulars themselves, which, if they are catered for at all in current record systems, are represented in heterogeneous and unstable ways.

Drawing on this framework, we have proposed a calculus for use in quality assurance of the complex representations created for clinical or research purposes, for example in coding of clinical trial data [28]. This calculus is based on a distinction between three levels [29]:

(1) the level of reality (for example, in the medical domain, the reality on the side of the patient);

(2) the cognitive representations of this reality for example as embodied in observations and interpretations on the part of clinicians and others;

(3) the publicly accessible concretizations of these cognitive representations in artifacts of various sorts, of which ontologies and terminologies and Electronic Health Records are examples.

---

[2] For clarity, we will from here on represent particulars in ***bold italic*** and universals in SMALL CAPS. Terms (or other representational units) denoting either universals or particulars will be written in italics between single quotes. For additional clarity, we will sometimes use the words '*particular*', '*universal*' and '*term*' explicitly to denote entities of the corresponding type.

4. Results

4.1. Terminological conventions

In line with the terminology proposed in [29], we will henceforth use the term '*class'* to denote a collection of all and only those particulars to which a given general term applies. A class can be either: (1) the extension of a universal, thus comprehending all and only those particulars which instantiate the corresponding universal (at that time); or (2) a subset of the extension of a universal defined as being such that the *members* of this class exhibit an additional property which is (a) not shared by all instances of the universal, and (b) also (can be) exhibited by particulars which are not instances of that universal. For such a class, we reserve the term '*defined class*'. Examples are: the class of influenza patients in Leipzig; the class of rabbits with congenitally absent nipples.

We will further use the term '*property'* to denote the combination formed by a relation that this particular enjoys with some other entity and this entity itself. For example, it is a property of my brain that it is part_of me; the property here is the combination formed by the part_of relation and my body. Similarly, it is a property of my brain that it is an instance of BRAIN; the property here is the combination formed by the instance_of relation and the universal BRAIN. Note that such combinations are not extra entities which exist in addition to the entities or relations through which they are formed. Rather, to talk of properties is to parse the reality already existing in the context of a given particular in a new way, reflecting the subject-predicate structure of languages such as English and the 'F(a)' structure of predicate logic based languages.

By '*portion of reality*' we mean any combination of particulars (including classes and defined classes), universals and properties. The use of this expression, too, does not reflect any extra entities which would exist in addition to the entities or relations which already exist and are classified under other headings.

We use the term '*representational unit*' (RU) for any symbolic representation (a code, a character string, an icon, …) which *denotes* a portion of reality and which is not constructed out of smaller parts which play a similar denoting role.**Table 2** gives an overview of the type of representational artifacts that are useful for representing portions of reality and of the sorts of entities that should be represented in each type of artifact. The latter is inspired by the view that *reference ontologies* should be the equivalent of scientific theories and therefore should represent what is generic in the world – whether or not in a specific domain – in a way that maximizes faithfulness and comprehensiveness with respect to reality.

Table 2: representational artifacts and their suggested representational units

Representational artifact	Contains representational units for …
Reference Ontology	• universals • relationships between universals (following the principles of the Relation Ontology [26])
Application Ontology	• universals • defined classes • relationships between universals and defined classes (following the principles of the Relation Ontology [26]) • particulars required for defining defined classes
Inventory	• particulars • properties

Application ontologies, in contrast represent matters in a local, purpose-driven way and in a format that is more suitable for computation [30]. Examples of *inventories* are databases which store information about particulars, examples being Electronic Health Records or Adverse Event Registries.

4.2. Core representational units

Table 3 shows the minimal collection of classes related to entities in reality that must be taken into consideration if we are to be in a position to represent the portion of reality around a particular patient on whose side an adverse event might have occurred under any of the definitions for adverse event analyzed thus far. Under the label '*denotation'* we propose a generic term applicable to a member of the corresponding class. The '*Class type*' column indicates whether the class is the extension of a universal (U) or a defined class (DC). The '*Particular type*' column indicates to what category of particulars, in terms of Basic Formal Ontology, the members of the corresponding class belong.

The descriptions provided in the right-most column are, be it noted, not to be interpreted as definitions for the terms that we choose to use in our ontology to denote the corresponding entities. Rather, they serve only to illustrate the sorts of roles played by different sorts of entities in a scenario in which an adverse event might have occurred. It is important, too, that the terms listed under the denotation-column should be seen as pertaining to the domain of adverse events. Thus for example we do not claim that anything which would be referred to by third parties by means of the term '*observation*' falls under the description provided. The conditionals that are used in most of these descriptions reflect the fact that a particular portion of reality might be such that a phenomenon which is considered to be an adverse event under one definition, is not an adverse event in terms of another definition. The conditionals should not be interpreted as having in every case to do with probabilities or uncertainty.

4.3. The place of 'adverse events'

The representational units for the core classes identified above can be used to represent all possible portions of reality which feature entities that can be referred to by means of the term '*adverse event*' under any of the definitions listed in [12]. As an example, **Table 4** lists the particulars and associated properties involved in a case in which

a patient born at time t_0

undergoing anti-inflammatory treatment and physiotherapy since t_2

for an arthrosis present since t_1

develops a stomach ulcer at t_3.

This table thereby provides an example of an adverse event case analysis of the sort that is made possible by the framework here presented.

The relationships employed in composing representations of properties in this Table are drawn from [26, 31]. We preserve the formatting conventions proposed in [26], except that we pick out particulars using ***bold italic***. We introduce the primitive **is_about** relation holding between a representational unit and the entity in reality about which this unit contains information at a certain time. We further take certain shortcuts in our representation of the temporal relationships involved in such an analysis, by simply stating for example that t_0 **earlier** t_1 **earlier** t_2 **earlier** t_3.

Table 3: Universals and Defined Classes for the adverse events domain.

	Denotation	Class Type	Particular Type	Description (role in adverse event scenario)
				Level 1
C 1	subject of care	DC	independent continuant	person to whom ***harm*** might have been done through an ***act under scrutiny***
C 2	act under scrutiny	DC	act of care	***act of care*** that might have caused ***harm*** to the ***subject of care***
C 3	act of care	U	process	activity carried out by a ***care giver*** to a ***subject of care***, motivated by an ***underlying disease*** and a ***care intention***
C 4	care giver	DC	independent continuant	person that performed an ***act of care*** directed to the ***subject of care***
C 5	underlying disease	DC	dependent continuant	the disease in the ***subject of care*** which is part of what serves to motivate performance of the ***act of care***
C 6	involved structure	DC	independent continuant	anatomical structure (of the ***subject of care***) involved in an ***act of care***
C 7	structure change	U	process	change in an anatomical structure of a person
C 8	structure integrity	U	dependent continuant	aspect of an anatomical structure deviation from which would bring it about that the anatomical structure would either (1) itself become dysfunctional or (2) cause dysfunction in another anatomical structure
C 9	integrity change	U	structure change	change in the ***structure integrity*** bringing about a change in the range of circumstances under which the anatomical structure would become dysfunctional or cause dysfunction in another structure
C 10	harm	U	integrity change	***integrity change*** bringing about an expansion in the range of circumstances of the sort typically occurring in the life of the ***subject of care*** under which the anatomical structure would become dysfunctional or cause dysfunction in another structure
C 11	care effect	DC	integrity change	***integrity change*** brought about by an ***act of care***
C 12	subject investigation	DC	process	looking for a ***structure change*** in the ***subject of care***
C 13	harm assessment	U	process	determining whether an ***observation*** is faithful to reality, and if so, whether the ***structure change*** which is the target of the ***observation*** is a ***harm***
C 14	care intention	DC	dependent continuant	intention of a ***care giver*** that motivates him towards an ***act of care***
				Level 2
C 15	observation	DC	dependent continuant	cognitive representation of a ***structure change*** resulting from an act of perception within a ***subject investigation***
C 16	harm diagnosis	DC	dependent continuant	cognitive representation, resulting from a ***harm assessment***, and involving an assertion to the effect that a ***structure change*** is or is not a ***harm***
C 17	care effect belief	DC	dependent continuant	belief on the side of the ***care giver*** concerning the ***care effect*** that he ascribes to the ***act of care***
				Level 3
C 18	care reference	DC	information entity	concretized (through text, diagram, …) piece of knowledge drawn from state of the art principles that can be used to support the appropriateness of (or correctness with which) processes are performed involving a ***subject of care***

Table 4: Example of an adverse event case analysis

IUI	Particular description	Properties
#1	the patient who is treated	*#1* **member** C1 **since** t_2
#2	*#1*'s treatment	*#2* **instance_of** C3 *#2* **has_participant** *#1* **since** t_2 *#2* **has_agent** *#3* **since** t_2
#3	the physician responsible for #2	*#3* **member** C4 **since** t_2
#4	*#1*'s arthrosis	*#4* **member** C5 **since** t_1
#5	*#1*'s anti-inflammatory treatment	*#5* **part_of** *#2* *#5* **member** C2 **since** t_3
#6	*#1*'s physiotherapy	*#6* **part_of** *#2*
#7	*#1*'s stomach	*#7* **member** C6 **since** t_2
#8	*#7*'s structure integrity	*#8* **instance_of** C8 **since** t_0 *#8* **inheres_in** *#7* **since** t_0
#9	*#1*'s stomach ulcer	*#9* **part_of** *#7* **since** t_3
#10	coming into existence of ***#9***	***#10*** **has_participant** ***#9*** **at** t_3
#11	change brought about by ***#9***	***#11*** **has_agent** ***#9*** **since** t_3 ***#11*** **has_participant** ***#8*** **since** t_3 ***#11*** **instance_of** C10 **at** t_3
#12	noticing the presence of ***#9***	***#12*** **has_participant** ***#9*** **at** t_{3+x} ***#12*** **has_agent** ***#3*** **at** t_{3+x}
#13	cognitive representation in ***#3*** about ***#9***	***#13*** **is_about** ***#9*** **since** t_{3+x}

We also allow for temporal annotations additional to those described in [26], at the same time remaining faithful to EN 12388:2005: Health Informatics – Time Standards for Healthcare Specific Problems [32].

Under the proposed scenario, ***#10***, i.e. the appearance of ***#9***, would (modulo the wide variation in interpretations that can be given to the majority of the definitions found) qualify as an adverse event as defined by the Institute of Medicine (definition D5).

However, for definition D9, it would rather be ***#9*** itself that would so qualify, while for D4, the definition of 'adverse event' proposed by the BRIDG consortium [4], it would be either ***#12*** or ***#13***. The counterintuitive nature of the latter case has its roots in certain conflations in the HL7 RIM [33], by which BRIDG is heavily inspired.

Because of the various sorts of entities that qualify as adverse events depending on which definition is used, at least two adverse event classes need to be defined: one for adverse events under views that see adverse events as processes, and one for adverse events that see them as continuants. A further distinction has to be made between adverse events as entities in first order reality, and phenomena in first order reality qualified as adverse events by relating to certain cognitive representations, records or theories.

5. Discussion

Already a very superficial analysis of the definitions in **Table 1** applying the analytical principles just sketched demonstrates that the question "*What are adverse events?*" cannot be answered directly, but needs to be reformulated as "*What might the author of a particular sentence containing the phrase 'adverse event' be referring to when he uses that phrase?*". Indeed, the authors of the listed definitions must have had very distinct entities in mind: we cannot imagine even one single example of an entity which would be such that, were it placed before these authors, they would each in turn be able to point to it while the first would say – faithfully and honestly – "*that is an observation*" (definition D4), the second: "*that is an injury*" (definition D9), the third: "*that is a laboratory finding*" (definition D6), and so on. Clearly, nothing which **is** an injury can **be** a laboratory finding, although, of course, laboratory findings can aid in diagnosing an injury or in monitoring its evolution. Similarly, nothing which **is** a laboratory finding, can **be** an observation, although, of course, some observation must have been made (by either a human being or a device) if we are to arrive at a laboratory finding.

One could argue, perhaps, that the authors of some of these definitions resort to metonymy, i.e. linguistic formulations in which a term denoting some entity is replaced by a term that denotes a related entity as in 'The White House decided that …', rather than 'The President of the United States decided that …'. If that would be the case, we would still have to qualify such usage as bad practice, specifically because we are convinced that definitions should be constructed to avoid ambiguity, rather than to contribute to confusion. This is all the more the case where the definitions in question are to serve as the basis for reasoning systems developed for use by computers.

However, because all the authors of the mentioned systems use the term '*adverse event*' in some context for a variety of distinct entities, and because these contexts look quite similar – in each of them, more or less the same sort of entities seem to be involved – there is some common ground (some portion of reality) which is such that the entities within it can be used as referents for the various meanings of '*adverse event*'.

5.1. Classifying adverse event related entities in terms of the three levels of reality

The definitions for the term '*adverse event*' and for other closely related terms differ amongst themselves in that they require a representation which resorts to one, two or all three levels of reality as described above. The first part of D12 (from the Food and Drug Administration) is an example in which all terms refer to level 1 entities: *drugs*, *drug doses*, *deaths*, *hospitalizations*, *disabilities*, and so forth, are all entities that exist in first-order reality. Another example is D9: the terms '*injury*', '*medical management*', '*measurement*' and '*disability*', when used in the context of a specific patient that may or may not have experienced an adverse event, all denote existing entities on the side of that particular patient and his environment, and are not about something else: these terms thus denote level 1 entities. D2, in contrast, requires bringing level 2 and perhaps even level 3 entities into the picture, and this because of the clause '*any failure of <u>expected</u> pharmacological action*'. Expectations can only be raised by a cognitive being and are part of the cognitive representation this cognitive being has constructed ***about*** the first order reality which forms his environment. Thus, in this interpretation of D2, i.e. if the expectation concerning the pharmacological action is 'in the mind' of the particular clinician assessing whether the patient has an adverse drug experience, D2 involves a level 2 entity. However, if this expectation is something which is part of '*general knowledge*' or belongs to the '*state of the art*', then we are dealing with an additional level 3 entity: in order for the clinician assessing the case to have access to that '*general knowledge*', it must have been concretized in some enduring fashion, for example in a manual or textbook.

5.2. Lack of clarity in definitions

D2 exhibits a characteristic which, unfortunately, is shared by the majority of the definitions encountered: they lack sufficient clarity of phrasing to allow an analysis to be conducted unproblematically in realist terms. Often multiple interpretations can be given to one or more terms used within such a definition, whereby each interpretation suggests a denotation at a distinct level of reality. An example is definition D3, in which the response that is described as being *undesirable* can be understood in three different ways:

(1) as denoting something on level 1, namely a *realizable entity* (a *disposition* or *tendency* [34]), which exists objectively as an increased health risk; in this sense any event '*that either compromises therapeutic efficacy, enhances toxicity, or both*' is undesirable;
(2) as denoting something on level 2, so that, amongst all of those events which influence therapeutic efficacy or toxicity, only some are considered *undesirable* (for whatever reason) by either the patient, the caregiver or both; or
(3) as denoting something relating to level 3, so a particular event occurring on level 1 is *undesirable* only when it is an instance of a type of event that is listed in some guideline, good practice management handbook, i.e. in some published statement of the state of the art in relevant matters.

In other cases, this sort of analysis results in detecting hidden assumptions, conflations or even serious inconsistencies either within one definition or in the combination of several definitions offered by the same source.

An example of an inconsistency within one single definition when the latter is analyzed in realist terms is provided by the attempt at a literal interpretation of D5, and more precisely of the use, there, of the term '*act of omission*', especially if, as suggested by D10, that term is taken in such a way that it does not denote anything which exists either now or in the past. In Referent Tracking terms, there would thus be nothing to which a IUI could be assigned. Indeed, while we believe that the phrase 'action not taken' is a linguistic description (level 3 entity) that can be used adequately and meaningfully in reporting some feature of a complex portion of reality (level 1 entity), such a use does not yet signify that the term denotes directly some entity in that portion of reality. While terms of the form 'doing something' do have referents in first order reality, there are no such referents denoted by terms like 'doing nothing'.

Consider the example given in [5], in which '*not testing a diabetic patient for HbA_{1c}*' is stated to be an '*act of omission*'. This is because, in result of the work of the Diabetes Quality Improvement Project [35, 36], it is considered bad practice not to do such a test at regular intervals [37]. But clearly, if all that exist as relevant first order entities are a patient's disease (here, the diabetes) and some adverse event , then it is not possible that some '*act of omission*' – i.e. some *not doing something* that one is supposed to do according to the state of the art – could be the cause of the adverse event. The only such cause would here be the underlying disease. Events, so we believe, can only be caused by what exists. And it is in the given case indeed clear that it is precisely the diabetes on the side of the patient that causes the adverse event, although it is true that, if the test had been taken, along with further appropriate actions in line with the results of that test, then it could be expected that no adverse event would have occurred. Therefore, a better definition for what D5 is trying to express would be: '*an event that results in unintended harm to the patient (1) through an act of commission* *rather than through some underlying disease or condition of the patient, or (2) through an underlying disease or condition of the patient in the absence of appropriate actions which should have been taken in line with the state of the art in dealing with the disease*'. This rephrased definition accounts better for something else that the Institute of Medicine almost certainly had in mind when producing D5, namely that many acts of commission are part of a procedure which, in order to be conducted *lege artis*, must include taking actions of a sort which, if they would not be taken, would lead to harm to the patient because of the act of commission. An example is incising an artery during some surgical procedure in a way that inevitably leads to bleeding. It would be inappropriate, in such case, not to take actions to reduce the bleeding. Here it is not the underlying disease which leads to harm to the patient, and nor is it the 'not stopping the bleeding' which leads to the harm. Rather it is the bleeding caused by the incision.

6. Conclusion

We have used the principles of Basic Formal Ontology (BFO), including the Relation Ontology (RO), and Referent Tracking (RT) as an analytical framework to study the ontological nature of what is denoted by the term '*adverse event*'. Our research indicates that this framework is adequate to serve a number of important purposes, and that, when used appropriately, it avoids the inconsistencies and incompatibilities inherent in other approaches. Nevertheless, some further developments, especially in RO are required if we are to be able to deal more formally with some extensions that we proposed here: (1) a family of relations to deal with various aspects of *aboutness* and *denotation* to relate level 2 and level 3 entities to level 1 entities, (2) a *membership* relation to link particulars to defined classes, and (3) the capacity to refer to (open-ended) time periods in addition to time instants.

7. Acknowledgments

We wish to thank Barry Smith for useful comments on our ontological analysis.

8. References

[1] REMINE Consortium. REMINE: Supporting Hospitals in Risk Management. 2008 [cited 2008 April 9]; Available from: http://www.remine-project.eu/

[2] The Joint Commission. Sentinel Event Glossary of Terms. 2008 2008 [cited 2008 April 14]; Available from: http://www.jointcommission.org/SentinelEvents/se_glossary.htm

[3] U.S. Food and Drug Administration. IND Safety Reports. In: Code of Federal Regulations Title 21 Volume 5 § 312.32, ed. 2007.

[4] Weng C, Becich M, Fridsma D. Collective Domain Modeling across Clinical Trials Standards: Needs, Challenges, and Design Implications. *Information Technology and Communications in Health (ITCH)*. Victoria, BC, Canada 2007.

[5] Committee for Data Standards for Patient Safety - Institute of Medicine. Patient Safety: Achieving a New Standard of Care. Washington DC: The National Academies Press 2004.

[6] National Cancer Institute. Common Terminology Criteria for Adverse Events (CTCAE) Version 3.0. 2006.

[7] CDISC Submission Data Standards Team. CDISC SDTM Implementation Guide (Version 3.1.2) - Study Data Tabulation Model Implementation Guide: Human Clinical Trials; 2007 July 25.

[8] Quality Interagency Coordination Task Force. Glossary of Terms. 2008 [cited 2008 April 14]; Available from: http://www.quic.gov/report/mederr8.htm

[9] U.S. Food and Drug Administration. Code of Federal Regulations; Title 21--Food And Drugs; Chapter I--Food And Drug Administration; Department Of Health And Human Services ; Subchapter D--Drugs For Human Use; Part 310 -- New Drugs; Subpart D--Records and Reports. 2007.

[10] Smith B, Ceusters W, Temmerman R. Wüsteria. In: Engelbrecht R, Geissbuhler A, Lovis C, Mihalas G, eds. *Connecting Medical Informatics and Bio-Informatics Medical Informatics Europe 2005*. Amsterdam: IOS Press 2005:647-52.

[11] Institute of Medicine. To Err is Human: Building a Safer Health System. Washington DC: National Academy Press 2000.

[12] Ceusters W, Capolupo M, Moor GD, Devlies J. Various Views on Adverse Events: a collection of definitions. 2008 April 20, 2008 [cited 2008 April 29]; Available from: http://www.org.buffalo.edu/RTU/papers/AdverseEventDefs.pdf

[13] The WHO World Health Organization Alliance for Patient Safety. Project to Develop the International Patient Safety Event Taxonomy: Updated Review of the Literature 2003-2005; Final Report. Geneva: World Health Organization; 2005 September.

[14] Chang A, Schyve PM, Croteau RJ, O'leary DS, Loeb JM. The JCAHO patient safety event taxonomy: a standardized terminology and classification schema for near misses and adverse events. International Journal for Quality in Health Care Advance Access. 2005;17(2):95-105.

[15] Smith B. From Concepts to Clinical Reality: An Essay on the Benchmarking of Biomedical Terminologies. Journal of Biomedical Informatics. 2006;39(3):288-98.

[16] Jeong S, Kim H-G. Design of Semantically Interoperable Adverse Event Reporting Framework. *The Semantic Web – ASWC 2006*. Berlin / Heidelberg: Springer 2006:588-94.

[17] College of physicians and surgeons of Newfoundland and Labrador. Disclosure of an Adverse Outcome. 2008 [cited 2008 April 16]; Available from: http://www.cpsnl.ca/PolicyDocument.asp?ID=21

[18] Spear AD. Ontology for the Twenty First Century: An Introduction with Recommendations. Saarbrucken, Germany: Institute for Formal Ontology and Medical Information Science 2006.

[19] Smith B, Ashburner M, Ceusters W, Goldberg L, Mungall C, Shah N, et al. The OBO Foundry: Remolding Biomedical Ontologies to Support Data Integration. Nature Biotechnology. 2007;25:1251-5.

[20] Johansson I. Bioinformatics and Biological Reality. Journal of Biomedical Informatics. 2006;39(3):274-87.

[21] Ceusters W, Smith B, Goldberg L. A terminological and ontological analysis of the NCI Thesaurus. Methods of Information in Medicine. 2005;44:498-507.

[22] Ceusters W, Smith B, Kumar A, Dhaen C. Ontology-based error detection in SNOMED-CT®. In: Fieschi M, Coiera E, Li Y-CJ, eds. *MEDINFO 2004*. Amsterdam, The Netherlands: IOS Press 2004:482-6.

[23] Ceusters W, Smith B, Kumar A, Dhaen C. Mistakes in medical ontologies: Where do they come from and how can they be detected? In: Pisanelli DM, ed. *Ontologies in Medicine Studies in Health Technology and Informatics*. Amsterdam, The Netherlands: IOS Press 2004:145-64.

[24] Smith B. Ontology and the Logistic Analysis of Reality. In: Guarino N, Poli R, editors. Proceedings of the International Workshop on Formal Ontology in Conceptual Analysis and Knowledge Representation; 1993; Padova: Institute for Systems Theory and Biomedical Engineering of the Italian National Research Council; 1993. p. 51-68.

[25] Ceusters W, Smith B. Ontology and Medical Terminology: why Descriptions Logics are not enough. *Towards an Electronic Patient Record (TEPR 2003)*. San Antonio 2003.

[26] Smith B, Ceusters W, Klagges B, Köhler J, Kumar A, Lomax J, et al. Relations in biomedical ontologies. Genome Biology. 2005;6(5):R46.

[27] Ceusters W, Smith B. Strategies for Referent Tracking in Electronic Health Records. Journal of Biomedical Informatics. 2006 June;39(3):362-78.

[28] Ceusters W, Smith B. A Realism-Based Approach to the Evolution of Biomedical Ontologies. *Proceedings of AMIA 2006* 2006:121-5.

[29] Smith B, Kusnierczyk W, Schober D, Ceusters W. Towards a Reference Terminology for Ontology Research and Development in the Biomedical Domain. *KR-MED 2006, Biomedical Ontology in Action*. Baltimore MD, USA 2006.

[30] Menzel C. Reference Ontologies — Application Ontologies: Either/Or or Both/And? 2007 [cited 2008 April 28]; Available from: http://bioontology.org/wiki/images/d/d9/MenzelOntology.pdf

[31] Smith B, Grenon P. The Cornucopia of Formal-Ontological Relations. Dialectica. 2004;58(3):279-96.

[32] European Committee for Standardization. EN 12388:2005. Health informatics - Time standards for healthcare specific problems. 2005.

[33] Smith B, Ceusters W. HL7 RIM: An Incoherent Standard. In: Hasman A, Haux R, Lei Jvd, Clercq ED, Roger-France F, eds. *Studies in Health Technology and Informatics Ubiquity: Technologies for Better Health in Aging Societies - Proceedings of MIE2006*. Amsterdam: IOS Press 2006:133-8.
[34] Jansen L. Tendencies and other Realizables in Medical Information Sciences. The Monist. 2007 October 2007:(in press).
[35] McLaughlin S. The Diabetes Quality Improvement Project. Diabetes Spectrum. 2000;13(1):5-10.
[36] American Diabetes Association. Standards of Medical Care in Diabetes. Diabetes Care. 2005;28:S4-S36.
[37] National Committee for Quality Assurance. HEDIS 2008 Technical Specifications for Physician Measurement National Committee for Quality Assurance 2007.

7. Vagueness

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-253

Granular Models for Vague Predicates

Silvia GAIO [a,1]
[a] *Department of Philosophy, University of Padua, Italy*

Abstract. This paper presents an account for vague predicates based on granular partitions, alternative to Bittner and Smith's one (e.g. T. Bittner and B. Smith [1], [2], [3].).

I consider one kind of vague linguistic expressions: adjectives like *tall*, *short*, *big*, *small*, etc. They are used in problematic and unproblematic cases. Consider two examples. (a) Marc is 160 cm tall and John 190 cm tall. English speakers do not hesitate to assign the adjective *short* to Marc and *tall* to John. (b) A hundred men differ for 0.5 cm with respect to their height. That is a borderline case: speakers can be indecisive about what men are *tall* and what *short*. This paper presents a model that accounts for the computational operations that underlie speakers' application of vague adjectives both to problematic and unproblematic cases. The model can then be used in a theory of formal ontology based on the notion of granularity as the one developed by Bittner and Smith.

The model is built on two basic ingredients: (i) comparison classes and (ii) granular partitions. (i) Comparison classes are introduced to account for the context-sensitivity of vague adjectives. The extension of the predicate *being tall* in the comparison class of men is different from its extension in the comparison class of children. (ii) We can look at the elements of a context under different standards of precision, each of them corresponding to a granular level of observation. Finer the level is, more differences between the individuals are detected. Granular partitions as equivalence classes are used to represent indistinguishability relations between objects with respect to the properties expressed by vague adjectives. The elements of each comparison class turn out to be weakly ordered with respect to each vague predicate. Such an algebraic treatment makes the model computational.

Keywords. Vagueness, Granularity, Gradable Adjectives, Contextualism

Introduction

You can observe the world under several perspectives that differ one from the other for the standard of precision adopted in each of them. For instance, if you want to draw the map of Italy, you can take a coarse point of view and draw Italy as boot-shaped. But taking a less coarse perspective, you can add some further details and draw the shore line more precise, and finer the perspective is, more Inlets are detected and drawn.

Those informal consideration about more or less refined points of view towards the world are captured by the notion of granularity. The idea of connecting granularity to vagueness has been developed in Bittner and Smith's jointed works, such as [1], [2], [3]. Bittner and Smith worked mainly on the problem of vagueness of proper names.

[1]Corresponding Author: Silvia Gaio, Department of Philosophy, University of Padua, Piazza Capitaniato 3, 35139 Padova - Italy. E-mail: silvia.gaio@unipd.it.

In this paper, I do not consider the problems of vague singular reference that Smith and Bittner are concerned with. However, I take into account their idea of connecting granularity to vagueness to present a formal model and account for a kind of vague linguistic expressions that Bittner and Smith do not consider: adjectives like *tall*, *short*, *big*, *small*, etc.

It is well known that vague expressions present borderline cases, that is, cases where the application of predicates is questionable. But when English native speakers have to judge on non-borderline cases, the use of adjectives such as *tall/short* is not problematic. Consider the following example. There are three men, John, Bill and Marc. John is 185 cm tall, Bill 183 cm, Marc 165 cm. Speakers that see the three men do not know their precise height, but can observe that John and Bill do not relevantly differ in height, as well as they can detect a big difference between John-Bill and Marc. So, if the agents are asked to describe the height of the three guys using the adjectives *tall* and *short*, they will naturally say that John and Bill are tall, Marc is short. It can also happen that, given such a context, speakers refer to Marc by using the definite description "the short man", and to John and Bill by "the tall men". The intuition behind this phenomenon seems to be the following: when there is not a big difference between (at least) two objects with respect to some property, we can correctly attribute the same adjective to both the objects, but if there is a relevant difference between them, then we describe the objects using a pair of polar adjectives, like *tall/short*[2].

This paper presents a model that describes the computational operations underlying speakers' decision. It can turn out to be an useful tool for formal ontology, since one of its goals is to investigate the logical features of predication.

The paper is divided into five parts: in section 1 I describe the features of the kind of adjectives I want to account for; in section 2 I present the intuitions that set up the theoretical background for the model; in section 3 I define a language $\mathcal{L}$ and its interpretation; in section 4 the model is built up; finally, in section 5 I try to briefly compare my proposal to Bittner and Smith's one.

1. Gradable adjectives

Adjectives such as *tall*, *short* or *wet*, *dry* are called *gradable* and they are characterised by the following features:

1. they can occur in a predicative position, that is, after verbs such as 'be', 'become', 'seem', etc.;
2. they can be preceded by degree modifiers such as 'very', 'clearly', etc.;
3. they can be made into comparative and superlative.

Each gradable adjective can be classified as positive or negative. We can then pair adjectives in such a way that each pair contains a positive and a negative adjective. Such a classification is based on some empirical characteristics demonstrated by the adjectives themselves[3]. For instance, measure phrases can be associated with positive adjectives, but not with negative ones (you can say "John is 178 cm tall" but not "John is 178

[2] The idea that there must be a sort of gap in order to distinguish objects with respect to a property is expressed also in Halpern [4].

[3] For the following considerations and examples I refer to Kennedy [5].

cm short”). Negative adjectives allow downward entailments, while positive ones allow upward entailments. Consider, for example, the pair *safe/dangerous*, such that *safe* is negative, *dangerous* positive. From “It is dangerous to drive in Paris” you infer “It is dangerous to drive fast in Paris” (and also “It is dangerous to drive slow in Paris”) but not the reverse, and from “It is safe to drive fast in Des Moines” (or from “It is safe to drive slowly in Des Moines”) you infer “It is safe to drive in Des Moines”, but not the reverse.

Examples of polar pairs are: *tall/short*, *expensive/cheap*, *big/small*, *clever/stupid*.

Gradable adjectives, moreover, can be distinguished between relative and absolute[4]:

Absolute Adjectives like *wet*, *closed*, *flat*, have positive forms that relate objects to maximal or minimal degrees; they are not affected by the Sorites paradox, nor have borderline cases. Consider some examples that can explain the relation of absolute adjectives with maximal or minimal degrees. *Wet* and *open* require their arguments to have a *minimal* degree of the property they describes (adjectives such as *wet* or *open* are called *minimum standard* absolute adjectives). The polar counterparts of *wet* and *open* are respectively *dry* and *closed*. As they require their argument to possess a *maximal* degree of the property in question, they are called *maximum standard* absolute adjectives.

Consider now some examples to verify the acceptability of comparatives raised by absolute adjectives, and their sensitivity to degree-modifiers:
“The platinum is less impure than the gold”;
“The desk is wetter than the floor”;
“The window is closed enough to keep the cold out”.

Relative Adjectives like *tall*, *big*, *expensive*, have the following features:

1. Context-sensitivity: the extension of the predicates generated by relative adjectives changes from context to context. This means also that a sentence containing a relative gradable adjective can get different truth value depending on the context of utterance. For example, a sentence like “John is tall” can be true in the comparison class of *men*, but false in the comparison class of *basketball players*.
2. Borderline cases: there are cases where it is difficult to determine whether an adjective can be attributed to some object. And moreover, there is no clear sharp boundary between a positive and a negative polar relative adjectives.
3. Sorites-sensitivity: every relative gradable adjective can give rise to a Sorites paradox.

According to Keefe and Smith’s characterisation of vagueness, what makes an expression vague is the combination of three features: lack of sharp boundaries, presence of borderline cases, Sorites-sensitivity[5]. Relative gradable adjectives have all those three features, so they are vague expressions. I will hereafter focus on this class of adjectives and the predicates generated by them.

---

[4]For the considerations relative to such a distinction I refer to Kennedy [6].
[5]See Keefe-Smith [8] and Keefe [9].

2. Theoretical Background

I consider vagueness of relative gradable adjectives related to context-dependency and granularity.

Context-dependency is a feature of relative gradable adjectives, as mentioned in the previous section. For instance, the predicate *tall* applied to the domain of human beings has a different extension than the same predicate applied to the domain of equatorial trees. But also within the domain of human beings, there might be sub-contexts that influence the interpretation of the predicate. For example, in the context where we consider only Dutch women the predicate has a different extension than in the context where we consider Italian women. Moreover, suppose an individual named Sue is four years old and 130 cm tall. We can say that she is tall as a child, but short as a human being. In fact, if we compare Sue with the individuals of the domain of all human beings, she results to be under the height-average. But considered as a four-years-old child, she is quite tall. So, the extension of vague predicates depends on the evaluations that are made each time in a specific *comparison class*.

But within a comparison class, like the class of *children*, we can be interested in a more restricted context. For instance, it can happen that a primary school teacher wants to divide the children of his class into two groups, one containing all the tall children and the other all the short ones. In such a case, in order to judge about the height of his pupils the class of *children* is not relevant as comparison class. Instead, the teacher will compare each pupil among the set of children restricted to his class. Such a set is a subset of the comparison class of *children* and is called *context*. In the model that will be presented below both comparison classes and contexts are used.

Nevertheless, context-dependency is not sufficient for explaining the vagueness of predicates. Given a context, that is, a set of individuals, and a vague predicate P, there are several ways to consider the differences between the individuals in the context. We might look at them under different standards of precision. The standard we choose often depends on our interest or our actual purpose[6]. As the standard changes, we may cover different things under the same label or split meanings in a more refined way. We call this phenomenon *granularity*. Let us try to understand what it means.

We can think of standards of precision as observational levels that differ one from the other for the grain size assumed to observe the world. Each grain size provides a degree of specification of the meaning of vague predicates. The line between the extensions of two polar adjectives that apply to the same domain can change according to different granular levels. Consider, for instance, the couple of adjectives *tall* and *short* and the example about John, Bill and Marc stated above. Formally, say we have a context $o = \{j, b, m\}$, with j, b, m standing for John, Bill, Marc, respectively. If we look at o from a point of view with a coarse grain size, no difference among the objects is detected: Bill, John and Marc are all equally tall. The coarse point of view can be given, for instance, by a distant point of observation or by some specific purpose (for example, if we have to enlist men shorter than 160 cm, we do not discriminate between the men we have to rule out: they are all equally tall as they equally exceed the cut-off point). From a less coarse point of view, that is, using a finer grain size to discriminate differences, we might say that John and Bill are equally tall, while Marc is short. We can then establish a comparative

[6]See Hobbs [11] and Mani [12].

relation: John (Bill) is taller than Marc. But with an even finer grain size we can well distinguish the height of all the three men and say that John is taller than Bill and Marc, and Bill taller than Marc. Consider this very last ordering, given by the finest granular level. The same ordering can be provided by two models, that differ for the extension of *tall* and *short*. According to model 1 both John and Bill are tall and Marc short, according to model 2 only John is tall and both Bill and Marc short. But our intuition is that only model 1 is correct. Some constraints are then needed to rule out models like 2, that do not respect our intuitions. The model presented in this paper shows such constraints.

3. Language and Interpretation

To give a model to account for gradable adjectives, let us fix first a language and then an interpretation for it.

3.1. Language $\mathcal{L}$

Let $\mathcal{L}$ be a formal language through which some English expressions can be represented. $\mathcal{L}$ consists of:

1. individual constant symbols (that represent proper names: *John, Mary, Sue,...*): j,m,s, ...
2. individual variable symbols: x,y,z, ...
3. monadic predicates (representing common count nouns like *pig, man, winner*): A, B, C, ...
4. functions (representing relative gradable adjectives): P, Q, R, ..., standing for *tall*, *big*, *fat*, ... We will later define the polar counterparts of such functions: $\overline{P}$, $\overline{Q}$, $\overline{R}$, ..., standing for *short*, *small*, *thin*, ...
5. usual logical connectives with identity, quantifiers.

3.2. Interpretation of $\mathcal{L}$

Let $\mathcal{D}$ be the domain of objects.

First, I want to select the comparison classes within which the individuals are compared. Then, the values of the functions representing gradable adjectives are given within each comparison class. I do that in order to make possible for any individual x to be P in some comparison class, $\overline{P}$ in other comparison classes. In such a way, we can account for the example mentioned above: Sue, who is four years old and 130 cm tall, is tall if considered in the comparison class of children, short if considered in the comparison class of human beings. At the same time, the attribution of an adjective is relativised to a comparison class represented by a count noun. For instance, to evaluate 'John is tall', first a comparison class is fixed, like the class of *men*, and then the application of *tall* to that class is evaluated with respect to John. In such a case, then, 'John is tall' is equivalent to 'John is a tall man'. What I want to do is very similar to what Bennett [13] proposes. His theory does not treat adjectives and count names as predicates, but combines adjectives with count names, that is, adjectives are considered as "operators modifying the meaning of count nouns"[7].

So, first of all I assume that monadic predicates select some objects of $\mathcal{D}$. I am assuming here that it is always possible to give a precise extension for each monadic predicate. For example, I do not consider the problem concerning the extension of *child*, that is known to be a vague predicate too. I ignore this kind of problems because I am concerned with polar vague adjectives. So, I assume the domains to which the functions representing adjectives apply to be precise.

Each interpretation $I(A)$ is a comparison class, for A a monadic predicate. For short I call $I(A)$ s. Let CC be the set of comparison classes s.

I have already mentioned that comparison classes do not suffice to evaluate the assignment of an adjective to an individual (e.g. when we want to consider only the children in a primary school). We need subsets of comparison classes, i.e. *contexts*. A context o is a subset of a comparison class.

Definition 1 *Let O_s be the set of all contexts in some comparison class s: $O_s = \wp(s)$.*

4. Context Structures and Weak Orderings

In this section contexts structures are defined. Then, given some cross-contextual constraints, the comparative relation is also defined.

As Luce [14] himself highlights, the non-transitivity of indifference relations reflects human inability to discriminate with precision among things that do not differ much one from the other. Luce's consideration on this point perfectly fits our problem with vague predicates. We cannot make precise distinctions between two objects with respect to some observable property. That is why we get into trouble with Sorites series. Nevertheless, if we have some more precise standard of precision or a better way of measurement, we can detect more differences between the elements we consider. That is nothing else than the concept of granularity, as we saw above. According to different standards of precision, we can have different models that give rise to different orderings of the objects of our domain. Let us see how this works in a more detailed way.

Let $M = \langle \mathcal{D}, I_{CC}, P \rangle$ be a fixed model, or context structure. $\mathcal{D}$ is the whole domain, I_{CC} the set of comparison classes, P a function that maps the elements of context o to $P(o)$, that is, maps each context o to a subset of o containing all and only the elements of o that has the property represented by P.

Then, we define the polar counterpart $\overline{P}$ of P as a function that applies to the elements in a context to which P does not apply:

Definition 2 $\overline{P}(o) = \{x \in o : x \notin P(o)\}$.

[7]My approach to count nouns and adjectives is such that relative comparison is formalised, but not absolute comparison. That means, in the model presented here only sentences like 'John is taller than Mary', with John and Mary included in the same comparison class of human beings, are formally represented. Sentences like 'That dog is taller than this child' are excluded, since dogs and children do not belong to the same comparison class. However, Bennett's theory is able to account also for absolute comparison.

Here P and $\overline{P}$ are considered contradictories. The model could be improved to treat P and $\overline{P}$ as contraries, but this is left, for the moment, to further developments.

In the following paragraphs, a way to account for the context-sensitivity of adjectives is sketched.

4.1. *Van Benthem's constraints*

To account for the cross-contextual change of meaning of relative gradable adjectives, some constraints can be put on functions P, Q, R in order to make them behave in a different way in each comparison class, and produce an ordering relation.

Consider the cross-contextual constraints that van Benthem used in [16], based on the concept of difference pair (DP):

Definition 3 *Two elements x, y form a difference pair in a context o iff x is in the extension of P and y in the extension of $\overline{P}$, that is:*

$$\langle x, y\rangle \in DP(o) \textit{ iff}_{def}\ x \in P(o) \textit{ and } y \in \overline{P}(o).$$

The constraints are the following:

Upward Difference (UD)

Let $\langle e, e'\rangle$ be a difference pair in a context o. In each context o' containing o, there exist different pairs.

Put otherwise, if in a context o one element is tall, another short, (UD) makes sure that all the supersets of o will contain at least one element that is tall and one that is short. Those elements are not necessarily e, e'.

No Reversal (NR)

Let $\langle e, e'\rangle$ be a difference pair in a context o. There is no context o' such that $\langle e', e\rangle \in DP(o')$.

If in a context o one element e is tall and another e' short, in any other context o' the reverse cannot be the case. Maybe both e and e' are tall, or short, but it can never be the case that e' is tall and e short.

Downward Difference (DD)

Let $\langle e, e'\rangle$ be a difference pair in a context o. In each context o' contained in o which includes e, e', there exist some difference pairs.

If e is tall and e' short in a large context o, in a smaller context o' containing e, e' there will be difference pairs too.

4.2. *Comparative Relation*

Given the three constraints (NR), (UD), (DD), we can define first the comparative relation $>_P$ (to read: "more P than"):

Definition 4 $x >_P y$ *iff* $x \in P(\{x, y\}) \wedge y \notin P(\{x, y\})$

The relation $>$ is defined with respect to a predicate P and gives rise to a *weak order*. A weak order is a structure $\langle I, R\rangle$ with R a binary relation on the domain I such that R is irreflexive, transitive and almost-connected:

$$\forall x : \neg R(x, x) \quad \textbf{(IR)}$$

$$\forall x, y, z : (R(x, y) \wedge R(y, z)) \rightarrow R(x, z) \quad \textbf{(TR)}$$

$$\forall x, y, z : R(x, y) \rightarrow (R(x, z) \vee R(z, y)) \quad \textbf{(AC)}$$

Define now the relations 'being as P as' (i.e. the similarity relation $\sim_P$) and 'being at least as P as' ($\geq_P$) as follows, respectively:

Definition 5 *$x \sim_P y$ iff$_{def}$ it is not the case that $x >_P y$ nor $y >_P x$.*

Definition 6 *$x \geq_P y$ iff$_{def}$ $x >_P y$ or $x \sim_P y$.*

4.3. On the Set of Context Structures

The conditions for comparatives do not uniquely determine the behaviour of function P across comparative classes. Let M be a context structure of type $\langle \mathcal{D}, I_{CC}, P \rangle$. Different context structures can give rise to different $>_P$ orderings for the same set of contexts. That means that some context structures detect more differences between the elements in the contexts than other context structures; this fact corresponds to the intuition of granularity.

Consider a context $o \in O_s$. Since any context structure provides us with an equivalence relation $\sim_P$, equivalence classes partitioning the context are obtained. In each context structure equivalence classes represent groups of objects that turn out to be indistinguishable:

Definition 7 *Let $e \in o$. Define the equivalence class of e under $\sim_P$ as follows:*

$$[e]_{\sim_P} =_{def} \{x \in o : x \sim_P e\}.$$

Different context structures can give rise to different partitions, and therefore to different orderings between objects in contexts. We can partially order the context structures that give rise to the orderings, from the coarsest to the finest. In such a way, we also order the levels of granularity from which we consider individuals in contexts.

An important observation has to be made at this point. For contexts with more than two equivalence classes, even if two context structures give rise to the same $>_P$ ordering, they might have different functions of type P and $\overline{P}$. In general, for contexts with more than two equivalence classes there are at least two context structures that give rise to the same ordering.

Consider some examples of context structures for the context $o_1 = \{a, b, e, f\}$ that order o_1 as follows: $a >_P b >_P e >_P f$ and that agree on the behaviour of P on the pairs $\{a, b\}$, $\{b, e\}$, $\{e, f\}$. These are:

$M_1 \models P(o_1) = \{a\}, \overline{P}(o_1) = \{b, e, f\}$
$M_2 \models P(o_1) = \{a, b\}, \overline{P}(o_1) = \{e, f\}$
$M_3 \models P(o_1) = \{a, b, e\}, \overline{P}(o_1) = \{f\}$

But I do not want to have all these context structures that give so different extensions to P and $\overline{P}$. I want to rule out some of them. To do that, the suggestion is the following: given a context $o \in O_s$, consider also the context o^s, defined as follows:

Definition 8 $o^s =_{def} \{z \in s \mid \exists x, y \in o : x \geq_P z \geq_P y\}$.

That is, any context o contains some elements of s. Given a set of context structures that give rise to the same ordering for all contexts in O_s, when we consider some context $o \in O_s$ we consider also the elements in the domain of s that are 'in-between' the elements of o in the order raised by the set of context structures considered.

In the following lines I try to explain what I mean with 'in-between' elements and which purpose they do serve.

The intuition is the following. When we consider contexts, we look at real objects. Namely, when we use gradable adjectives we want to judge on some situation in the world. However, to correctly attribute gradable adjectives to objects, we can think that we add possible objects that are 'in-between' the real objects according to the comparative relation. That means, if John and Bill are men that relevantly differ along their height, we say that John is tall and Bill is short because there can be other men, whose height differs and is less than John's but more than Bill's. This intuition goes together with the fact that all vague relative adjectives suffer from the Sorites Paradox: the crucial point of Sorites paradox is that we have a series of objects, such that there are small differences between every two objects that are contiguous in the series. We want to model vague relative adjectives that give rise to Sorites series: so, we need to assume that we *can* have a domain of individuals that are "equally distributed" with respect to a property, and such that each element of any $s \in CC$ is indistinguishable from at least two others from an observational point of view. Put otherwise, what we want is each set to have possible objects that form a Sorites series, and each real object to correspond to one of the possible objects of the domain. So, we need one restriction concerning the domain of individuals of each comparison class.

Here I state the condition that needs to be imposed to each comparison class. Consider a comparison class s: each element of s is observably indistinguishable from at least two others. It might happen only for two elements that each of them is indistinguishable from another element[8]. So, every comparison class s comes with an indistinguishable relation $\approx_{s_P}$ with respect to a property P, such that the following formula holds:

$$\begin{aligned} &\exists u \exists v (u \neq v \wedge \forall x((x \neq u \wedge x \neq v) \rightarrow \exists y \exists z (x \approx_{s_P} y \wedge x \approx_{s_P} z \wedge \\ &\wedge \neg(y \approx_{s_P} z))) \wedge \exists z (z \neq u \wedge u \approx_{s_P} z) \wedge \exists z (z \neq v \wedge v \approx_{s_P} z)) \end{aligned} \qquad \textbf{(SC)}$$

The point is: with the primitive relation $\approx_{s_P}$ we only constrain the comparison classes in order to contain elements that are indistinguishable from at least two others (with two exceptions). We do not impose any order to the sets of possible objects. The ordering between elements is given by the function P in each context structure.

At this point, context structures need to be redefined. Let $M = \langle \mathcal{D}, I_{CC}, \approx_{s_P}, P \rangle$ be a fixed model, or context structure. $\mathcal{D}$ is the whole domain, I_{CC} the set of comparison

[8]These last two elements are predicted to be the minimal and the maximal element in the set of the individuals when ordered.

classes, $\approx_{s_P}$ the indistinguishability relation with respect to a $s \in CC$ and a property P, P a function that maps the individuals of context o into $P(o)$.

There is an important observation to make at this point. It concerns the difference between $\sim_P$ and $\approx_P$. They are not the same relation. While $\approx_P$ is given primitively, to describe what each comparison class looks like according to our observation, $\sim_P$ is defined on the behaviour of P and $\overline{P}$. Moreover, $\sim_P$ is an equivalence relation, but $\approx_P$ is not: it is reflexive, symmetric, but not necessarily transitive. While $\approx_P$ is used only to assure that we have enough individuals, that is, they form a soritical series, $\sim_P$ is defined on the function P within a specific context structure.

Now, contexts of type o^s are elements of O_s:

$$\forall o : o^s \in O_s. \tag{\textbf{Q}}$$

It must be noticed that o^s are Sorites series; namely, they contain a sequence of elements that satisfy (**SC**). Any context structure $M = \langle \mathcal{D}, I_{CC}, \approx_{s_P}, P \rangle$ with P satisfying van Benthem's constraints gives an ordering to *all* the contexts $o \in O_s$; so, it gives rise to a weak ordering also in contexts of type o^s.

As we partition all the contexts into equivalence classes, we consider also the contexts of type o^s as partitioned into equivalence classes.

In such a way, we get that all comparison classes are weakly ordered. In fact, if each context structure orders the elements of the subsets (i.e. the contexts) of a comparison class s, the whole comparison class gets an ordering of the same kind (weak) within the same context structure. That means, for each $s \in I_{CC}$, we have structures of type $\langle s, >_P \rangle$.

Assume now that the extensions of each predicate P and of their complements $\overline{P}$ in o^s, for all $o \in O_s$, have to behave as follows:

$$|[x]_{\sim_P} \in P(o^s)| = |[x]_{\sim_P} \overline{P}(o^s)| \pm 1. \tag{\textbf{E}}$$

Now, considering a set of context structures that give rise to the same ordering, and a context $o \in O_s$, I accept only the context structures that make the following formula true:

$$\forall x \in o : x \in P(o) \text{ iff } x \in P(o^s). \tag{\textbf{R}}$$

(**R**) says that the elements in o are P in o if and only if they are P in the *fulfilled* context o^s. (**R**) is the constraint that restricts the set of context structures, allowing exclusively the context structures that correctly say which objects are P and which are $\overline{P}$.

Consider again the example stated above: Let P be a function and o_1 the context $\{a, b, e, f\}$. Let o_1^s be $\{a, b, c, d, e, f\}$. That is, o_1^s contains two elements more than o_1, namely c and d. These two added objects are possible objects that are in-between b and e according to the comparative relation $>_P$ and are part of the natural set s under the constraint (**SC**).

Since I take into consideration only function P in the example, in the discussion about the example I will omit 'P' as index of the symbol '$>$' and I will use '$[x]$' to refer to the equivalence class $[x]_{\sim_P}$.

Consider the set of models that give rise to the ordering $a > b > e > f$ for o_1. Assume that the same set gives rise to the ordering $a > b > c > d > e > f$ for o_1^s. By (**E**) we have: $P(o_1^s) = \{[a], [b], [c]\}$, $\overline{P}(o_1^s) = \{[d], [e], [f]\}$.

Consider the models M_1, M_2, M_3:

$$M_1 \models P(o_1) = \{[a]\}, \overline{P}(o_1) = \{[b], [e], [f]\}$$
$$M_2 \models P(o_1) = \{[a], [b]\}, \overline{P}(o_1) = \{[e], [f]\}$$
$$M_3 \models P(o_1) = \{[a], [b], [e]\}, \overline{P}(o_1) = \{[f]\}$$

Applying (**R**) as a restriction on them, only M_2 turns out to be acceptable.

M_1 is ruled out because the equivalence class $[b]$ is mapped into the set $\overline{P}(o_1)$, while constraint (**R**) wants it to be mapped into $P(o_1)$. Informally speaking, the extension of $P(o_1)$ in M_1 contains 'too few' equivalence classes.

M_3 is ruled out for the opposite reason: the extension of $P(o_1)$ contains an equivalence class more than what is accepted. According to (**R**), $[e]$ has to be in the extension of $\overline{P}(o_1)$.

4.4. Tall and Short

Consider now the application of our model to the pair of vague adjectives *tall* and *short*. Let T stand for *tall*, $\overline{T}$ for *short*.

Recall the example stated above about John, Bill and Marc and the property of tallness. Consider the finest ordering, i.e. John is taller than Bill and Marc, Bill taller than Marc: $j >_T b >_T m$. There are a certain number of context structures that give rise to that ordering for the context o containing j, b, m. Now, some of them give the following extensions for T and $\overline{T}$ in o:

$$T(o) = \{j, b\}, \overline{T}(o) = \{m\}.$$

Other context structures give the following extensions:

$$T(o) = \{j\}, \overline{T}(o) = \{b, m\}.$$

In context o^s we have many individuals, and also many equivalence classes, between b and m. In fact, we can imagine many men whose height is between 165 cm and 183 cm. In such a case, actually, the elements of o_s are not only possible: some of them are real.

Considering constraint (**E**) for the extension of T and $\overline{T}$ in o^s, we have $j \in T(o^s)$, $b \in T(o^s)$, $m \in \overline{T}(o^s)$. So, John and Bill have to be considered tall, while Marc short. And that is exactly what speakers intuitively do when they are asked to use vague predicates in English in a situation concerning John, Bill and Marc as described above.

5. Concluding Remarks

To describe entities, that are object of ontological investigations, we commonly use predicates and, as we have seen, part of them are vague. The semantic granular model proposed in this paper can be considered as a tool to clarify how we correctly apply vague predicates to entities in natural language. Adopting more or less refined standards of precision we become able to describe the objects of some domain with a better degree of precision. We can say that the semantic model for vague predicates offers a conceptualization of vague predicates necessary to describe the entities to which they apply.

Bittner and Smith support also a granular approach to vagueness, but my proposal is quite different from theirs: while they propose a formal framework that connect the idea of granularity with mereotopology, what I present here is rather a semantic framework connecting granularity with algebra. Sets of fixed models (that have been called *context structures*) are taken to belong to granular levels, and granular partitions are meant to be exactly equivalent classes from an algebraic point of view.

Moreover, I did not handle the problem of indeterminate identities, the main problem considered by Bittner and Smith. The attempt here was to include granularity in a *semantic* treatment of vague predicates, that possibly turns out to be computational.

In Smith and Brogaard [17] it is pointed out that the term 'partition' is not used to mean 'equivalence class'. A granular partition is a grid of cells that gives an abstract classification of objects in reality:

> A granular partition is a way of dividing up the world, or some portion of the world, by means of cells.[9]

We can say, then, that while Bittner and Smith's granular partitions are a way to divide things and get several categories of objects related one each other, the granular partitions I propose are ways to describe objects in the world taken one by one and considered in their similarity relations with the others. In other terms, while granular partitions as systems of cells can be used to give a conceptualisation of the world itself and its constituents, granular partitions as equivalence relations for vague adjectives can be used to describe single items.

In the sorites series contiguous elements are indistinguishable with respect to a property P from an observational point of view. Using a more efficient way of measurement we can determine a larger number of differences between the objects. Hobbs suggests that the fact that we cannot do it by ourselves is not due to some shortcomings of our cognitive system. Instead, it shows that we have been

> attuned to the aspects of our environment that are most likely to be relevant to our interests.[10]

This idea might bring about a change in the epistemic conception of vagueness. Vagueness is not seen as a mere defect of our epistemic capacities, that is, of our capacity to be acquainted with the world around us. On the contrary, it is positively considered: it is the result of human adaptation to the world. If we cannot distinguish some differences is also because, from a pragmatic point of view, we do not usually *need* to do that.

The idea of granularity applied to vagueness not only can be a further tool to consider the problem, but can be also helpful to improve the epistemic approach to vagueness. Philosophical considerations about the implications of the idea of granularity in the debate about vagueness deserve a research that is worth pursuing.

[9] Smith and Brogaard [17], p. 6.
[10] Hobbs [11], p. 433.

Acknowledgements

The model sketched in the article had been developed during my study period at the ILLC, University of Amsterdam. The main ideas in this paper came up from my discussions with Frank Veltman and Robert van Rooij. I am deeply indebted and grateful to them.

Moreover, I would like to warmly thank the referees for their very relevant comments and helpful suggestions for further improvement.

Massimiliano Carrara encouraged me from the beginning to work on the topic of vagueness and read several versions of this paper. I want to thank him for paying so much attention to my research.

References

[1] T. Bittner and B. Smith, Vague reference and approximating judgements, *Spatial Cognition and Computation*, **3(2)**, 2003, 137-156.

[2] T. Bittner and B. Smith, A theory of granular partitions, in M. Duckham, M.F. Goodchild, and M.F. Worboys (Eds.), *Foundations of Geographic Information Science*, London: Taylor & Francis, 2003, 117-151.

[3] T. Bittner and B. Smith, Granular partitions and vagueness, in C. Welty and B. Smith (Eds.), *Formal Ontology and Information Systems*, New York: ACM Press, 2001, 309-321.

[4] J.Y. Halpern, Intransitivity and Vagueness, in D. Dubois, C.A. Welty and M.-A. Williams, *Proceedings of the Ninth International Conference on Principles of Knowledge Representation and Reasoning*, AAAI Press, 2004, pp. 121-129.

[5] C. Kennedy, On the Monotonicity of Polar Adjectives, in J. Hoeksema, H. Rullmann, V. Sanchez-Valencia and T. van der Wouden (Eds.), *Perspectives on Negation and Polarity*, 2001, 201-221.

[6] C. Kennedy, Vagueness and Grammar: the Semantics of Relative and Absolute Gradable Adjectives, *Linguistics and Philosophy* **30(1)**, 2007, 1-45.

[7] C. Kennedy, Modes of Comparison, in *Proceedings of CLS 43*, forthcoming.

[8] R. Keefe and P. Smith, Introduction: Theories of Vagueness, in R. Keefe and P. Smith (Eds.), *Vagueness: A Reader*, MIT Press, Cambridge (MA), 1997, 1-57.

[9] R. Keefe, *Theories of Vagueness*, Cambridge University Press, 2000.

[10] Fara, D.G., Shifting Sands: An Interest-Relative Theory of Vagueness, *Philosophical Topics*, **28(1)**, 2000, 45-81. Originally published under the name 'Delia Graff'.

[11] J.R. Hobbs, Granularity, in A.K. Joshi (Ed), *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, Los Angeles, CA, 1985, 432-435.

[12] I. Mani, A Theory of Granularity and its Application to Problems of Polysemy and Underspecification of Meaning, in A.G. Cohn, L.K. Schubert, and S.C. Shapiro (Eds), *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, 1998, 245-257.

[13] B. Bennett, A Theory of Vague Adjectives Grounded in Relevant Observables, in P. Doherty, J. Mylopoulos and C.A. Welty, *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning*, AAAI Press, 2006, pp. 36-45.

[14] R.D. Luce, Semiorders and a theory of utility discrimination, *Econometrica*, **24(2)**, 1956, 178-191.

[15] K. Arrow, Rational Choice Functions and Orderings, *Economica* **26**, 1959, 121-127.

[16] J. van Benthem, Later Than Late: On the Logical Origin of the Temporal Order, *Pacific Philosophical Quarterly* **63**, 1982, 193-203.

[17] B. Smith and B. Brogaard, Quantum Mereotopology, *Annals of Mathematics and Artificial Intelligence* **36**, 2002, 153-175.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-266

Counterparts in Language and Space

Similarity and $\mathcal{S}$-Connection

Joana HOIS and Oliver KUTZ
SFB/TR 8 Spatial Cognition
University of Bremen, Germany [1]

Abstract We aim to combine the semantics of spatial natural language specified as a linguistically motivated ontology, the Generalized Upper Model, with spatial logics or ontologies that specify space according to certain conceptualisations, based on regions, shapes, orientations, distances, or object properties.

Such combinations, however, introduce uncertainties of various kinds, caused by different levels of detail in the definition of one of the spatial ontologies, under-specifications within parts of an ontology, or different viewpoints of the topics the ontologies address.

To model these problems formally, we extend the combination technique of $\mathcal{E}$-connections by adding (heterogeneous) similarity measures. Local similarity compares objects within one domain, whilst comparing objects across domains leads to similarity measures that are motivated by and based on counterpart-theoretic semantics. The new formalism is called $\mathcal{S}$-connection.

Keywords. Ontology, Natural Language Semantics, Similarity, Counterparts, $\mathcal{E}$-Connection, $\mathcal{S}$-Connection

1. Introduction

"Tesco is the second building from there", "Take the left where the trees are on the corner", "Boots is past Plymouth university on the right hand side", "I'm going down 50 meters past the pine forest towards the wheat fields"—natural language describes spatial situations in a flexible way: within one description, it changes fluently in terms of granularity, combines different modes of spatial relationships, gives as much information as necessary needed for a specific purpose, refers to situation-dependent knowledge given by the dialogue discourse, or specifies attributes of spatial entities [1, 2, 3]. Spatial logics, in contrast, specify axiomatically only select aspects of the environment, but they do this with a relatively high degree of precision concerning those aspects. Spatial qualitative calculi as one group of spatial logics, for instance, differ in terms of the spatial entities and kinds of relationships they describe, as well as reasoning support. Specifications within a calculus may correspond to aspects about regions, orientations, shapes, distances, movements, topology, or metric spaces [4, 5].

Both, linguistic and logical formalisations of space, however, are applied at different levels within spatially aware information systems interfaced with a natural language

[1]Correspondence: Joana Hois/Oliver Kutz, SFB/TR 8 Spatial Cognition, Cartesium, University of Bremen, 28359 Bremen, Germany; E-mail: `joana, okutz@informatik.uni-bremen.de`

dialogue system [6]. Hence, relations between both these representations, linguistic and logical, that provide descriptions of the environment from different viewpoints, have to be aligned and integrated with each other.

In this paper, we provide a method that formally connects both viewpoints on the basis of $\mathcal{E}$-connections. While giving examples of how natural language, specified in a linguistically motivated ontology, is related to different spatial logics, we will elucidate the impact of uncertainties and similarities influencing this relationship. Connections of these viewpoints are strongly influenced by external factors, and so the relationship between instances in different domains can only be determined to a certain degree. A framework that supports a formalisation of such relationships is given, enriching the technique of $\mathcal{E}$-connections with (heterogeneous) similarity measures. These so-called $\mathcal{S}$-connections are motivated by and based on counterpart-theoretic semantics.

2. Linguistic Spatial Semantics

Language has a broad but structured range of ways for relating entities of different kinds to each other, both semantically and syntactically [7], and can therefore be partly specified as a formal theory or ontology. A linguistic categorisation particularly for spatial descriptions has been developed in the Generalized Upper Model (GUM) [8], which has been successfully applied in a natural language system [6] and which is evaluated against linguistic corpora with more than 600 entries for English and German. Its structure is governed by results from linguistic evidence, empirical research, and grammatical indications: it classifies language into groups of categories and relations according to their semantics. Hence, GUM is strictly based on the requirement that the distinctions that should be covered are those that are derived from linguistic evidence. This implies that GUM captures precisely those aspects given by the semantics, but not by the pragmatic principles and distinctions associated with particular lexicogrammatical items and structures.[2]

GUM's spatial categorisation is not based on groups of prepositions, but on the way language characterises spatial relationships either grammatically or inherently.[3] Natural language utterances about spatial contexts are specified accordingly as instances in GUM. Those distinction not covered by the linguistic structure are therefore not represented in GUM. Talmy [1] points out that language schematises spatial information only into *underspecified* qualitative concepts. These concepts then need to be adapted and interpreted with respect to specific spatial situations. This *underspecification* renders the connection between linguistic descriptions and formal spatial theories with uncertainty.

Given the ontological structure of GUM, the most expressive categorisation of linguistic aspects are those describing dynamic or static spatial configurations and, in particular, different kinds of spatial relationships [8]. In fact, different modes of spatial relationships give the strongest indication about relative positions or motions of spatial entities and their attributes [9]. These relationships, however, can only be seen in the context of the linguistic entities participating in the relationship. Lexical terms, however, are less

[2]A detailed overview of GUM would go beyond the scope of this paper; see [8] for details.

[3]Although GUM is based on the semantics of English and German, it is rooted in a language-based approach to cognition across different languages [7]. Language-dependent differences in spatial semantics should therefore result in refinements or extensions of GUM.

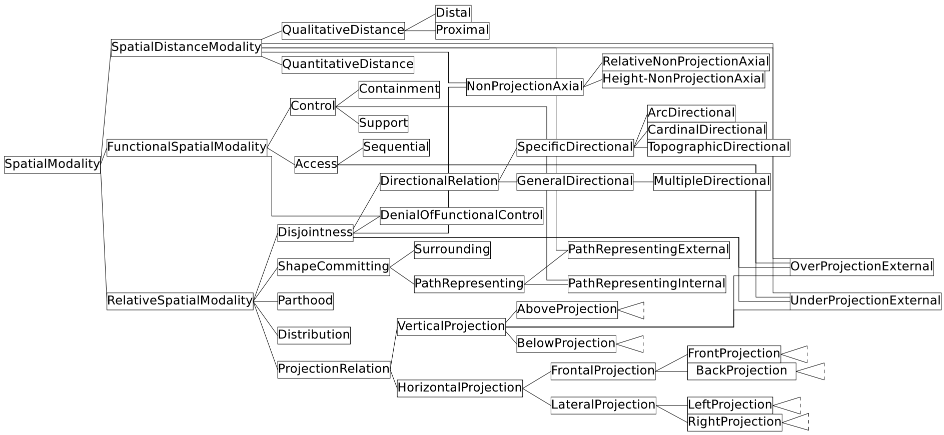


Figure 1. Spatial modalities represent modes of spatial relationships between entities. (ProjectionRelation leafs are further distinguished into internal and external projections, subsumed by Parthood or Disjointness and SpatialDistanceModality respectively.)

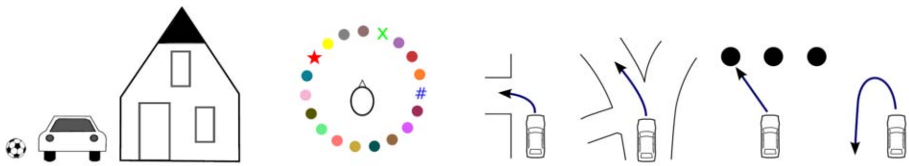
indicating the meaning of a spatial linguistic description, as they can be conceptualised in many ways according to the spatial relationship in use (cf. [10] on the meaning of "place"). We will therefore focus on these relationships.

GUM[4], as a formal theory, is specified in first-order logic. However, large parts of it can also be expressed in description logics such as $\mathcal{SROIQ}$ [11] (underlying the Web Ontology Language OWL 2.0). Its signature contains *categories* (unary predicates) and *relations* (binary predicates). The spatial extension of GUM introduces all categories and relations necessary for specifying utterances of spatial descriptions. Different kinds of spatial relationships are specified by the category SpatialModality. This category consists of several subtypes, which are defined by their use in natural language and possible entities they relate to. Related objects are then specified by the relations locatum in static and actor in dynamic spatial descriptions and the relatum [12], i.e. the locatum/actor has a certain spatial position with respect to the relatum (corresponding to *figure* and *ground* in [1] or *trajectory* and *landmark* in [13]).

All spatial descriptions indicate the type of relationship being described, typically expressed by a spatial preposition, an adverb, an adjective, parts or implications of the verb, that defines a specific SpatialModality. The most general distinction between spatial modalities is made by distance-, functional-, and property-dependent positions between entities. There are, however, intersections between these three general categories. An overview of GUM's spatial modalities is shown in Fig. 1.

The structure of these spatial modalities are given precisely but solely on the basis of linguistic evidence. Further distinctions made by spatial logics then have to be derived by situation-dependent, context-sensitive, or world knowledge, i.e. external factors. Possible *realisations* of specific linguistic descriptions in models of a spatial logic can therefore only be defined by elements that satisfy a certain similarity.

[4] `http://www.ontospace.uni-bremen.de/ontology/GUM-3-space.owl`



2.1: Ambiguities among static left projections 2.2: Ambiguities among dynamic left projections

Figure 2. Ambiguities for GUM's LeftProjection spatial modality

3. Connections between Spatial Language and Logics

The following examples illustrate the way linguistic descriptions tend to underspecify their possible spatial realisations. As a consequence, these descriptions can be related to different models of spatial logics.

ProjectionRelations in GUM define directional relationships between entities. They represent relationships between entities based on orientations. One of its subclasses, LeftProjection, defines spatial relations as used in the examples "Three steps to the left", "Turn to the left", "It is to the left of you", or "In the left part"; it denotes:

1. static locations, on/in the left side or half-plane of the relatum,
2. static locations with respect to the orientation of the relatum,
3. re-orientations towards the direction or an angle to the left,
4. (re-)directions of motions, to the left side of the moving entity, or
5. combinations of movements and re-orientations to the left of the moving entity or an external left [8].

Although the linguistic surface can reduce the range of realisations, not all possible distinctions are made. As GUM's specification of spatial language has been designed to cover all possible meanings in a flexible (linguistic) way, interpretations of specific utterances have to be determined in spatial situations by external (non-linguistic) factors [14]. Possible realisations of ProjectionRelations might therefore be defined in spatial logics that specify orientations, such as [15, 16, 17]. However, which concrete model corresponds to the linguistic description and vice versa depends on external aspects. Whether one or more connections between language and space are necessary, and to what degree they hold, has to be determined based on indications from these external aspects.

Fig. 2 illustrates spatial situations, in which LeftProjection can be used to describe relationships between entities. In the left part of Fig. 2.1, for instance, LeftProjection is defined in "The ball is *to the left of* the car and the car is *to the left of* the house". From the perspective of someone sitting inside the car, however, "The house is *to the left of* the car" is also acceptable without falsifying the previous example. Hence, LeftProjection has to be interpreted according to the spatial perspective. Furthermore, "The house is *to the left of* the ball" from the perspective of the car might be less acceptable depending on the Figure vs. Ground phenomenon [12], i.e. contextual aspects influence the interpretations of "left" as well.

Although a geometric relation according to a 90 degree angle or half-plane could be a logical definition for LeftProjection in this example, 'left' can be used to reflect further realisations. In the right part of Fig. 2.1, multiple objects are arranged as a circle. Here, one entity is to the left of the other. Various possibilities for "Drive to the left"

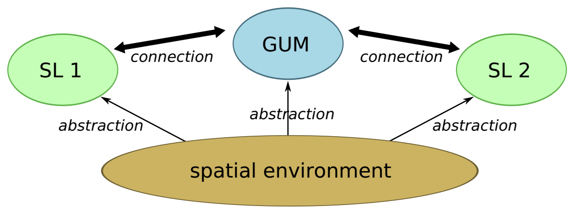


Figure 3. Connections between linguistic description (GUM) and spatial logics (SL 1, SL 2)

are illustrated in Fig. 2.2. Which specific direction is meant may depend on the course of the road, external entities or the intrinsic orientation of the car. In contrast to a linguistic LeftProjection, spatial logics define 'leftness' in an axiomatic way. In [15], a left-like relationship is divided into five possible regions according to orientations between two entities. In [16], 'left' is defined as a range of degrees of a point-based orientation with variable granularity. In [17], a *leftside(x,y,z)* relation is defined among three entities, which is specified as non-collinear. Given the examples in Fig. 2, spatial logics provide different realisations for particular relationships.

Taking into account only the linguistic input from the clauses above, nothing more than a LeftProjection ("left") is defined and possible realisations have to be determined by the context. In particular, these diverse interpretations of "left" cannot be covered by a logical relation left(a,b) together with spatial axioms such as transitivity, antisymmetry, and irreflexivity. Parts of the circle objects, for instance, violate transitivity. That the $\star$ is to the left of the x and the x is to the left of the $\sharp$ does not indicate that the $\star$ is to the left of the $\sharp$ (but rather opposite of it). And in case this would be an acceptable implication because of the circle-like arrangement, then left(a,b) would actually be symmetric (and the $\sharp$ to the left of the $\star$) and reflexive (and the $\star$ to the left of itself). Instead, the linguistic description has to be related to different models of spatial logics. Those objects in a model of a spatial logic that we take to be most adequate as a realisation of the linguistic description, we call the (spatial) *counterparts*. Hence, language specifies space according to linguistic evidence whereas logic specifies space according to its underlying theory of space. Formal relationships (connections) between both layers then have to be defined in order to determine counterparts.

Merging all kinds of spatial information into one theory that formulates all connections between language and space, however, would adversely affect effective reasoning techniques, decidability, expressiveness, modularity, and flexibility. The semantics of a spatial description can instead refer to distinct spatial models of spatial logics while underspecifying external factors (e.g. world knowledge, contextual and environmental information, or the dialogue history). Spatial language and logic can then be formally related by indicating their similarities. For instance, a LeftProjection may be realised as one of the examples in Fig. 2. As a result, the spatially-aware system should be able to determine at least the most likely connection.

In summary, language is connected to different spatial logics with regard to certain environments (see Fig. 3). This connection can be specified together with a similarity value determined by external factors, such as the context, domain-knowledge, environment, properties of spatial objects, alignment, and discourse. Most closely connected entities are called counterparts. $\mathcal{E}$-connections between language and spatial logics together with similarity values can realise this connection, as described in the next section.

4. Counterparts, Connections, and Similarity

David Lewis provided the first formal theory of counterparts [18], a two-sorted first-order theory, whose sorts are objects and worlds, and which has four predicates: $W(x)$ says that x is a world, $I(x, y)$ that x is in the world y, $A(x)$ that x is an actual object, and $C(x, y)$ that x is a counterpart of y.

He described the basic intuition underlying the idea of counterparthood as follows:

> Your counterparts resemble you closely in content and context in important ways. They resemble you more closely than do the other things in their worlds. But they are not really you. For each of them is in his own world, and only you are here in the actual world. [18], p. 27–28

The general idea of counterpart relations being based on a notion of *similarity across worlds* also lies at the heart of heterogeneous knowledge representation, and was a major motivation for the design of 'modular languages', $\mathcal{E}$-connections in particular [19].[5]

4.1. $\mathcal{E}$-Connections as Counterpart Theory

In $\mathcal{E}$-connections, a finite number of formalisms talking about distinct domains are 'connected' by relations between entities in different domains, capturing different aspects or representations of the 'same object'. For instance, an 'abstract' object o of a description logic $\mathcal{L}_1$ can be related via a relation R to its life-span in a temporal logic $\mathcal{L}_2$ (a set of time points) as well as to its spatial extension in a spatial logic $\mathcal{L}_3$ (a set of points in a topological space, for instance). Essentially, the language of an $\mathcal{E}$-connection is the (disjoint) union of the original languages enriched with operators capable of talking about the link relations. The possibility of having multiple relations between domains is essential for the versatility of this framework, the expressiveness of which can be varied by allowing different language constructs to be applied to the connecting relations.[6]

$\mathcal{E}$-connections have also been adopted as a framework for the integration of ontologies in the Semantic Web [22], and, just as DLs themselves, offer an appealing compromise between expressive power and computational complexity: although powerful enough to express many interesting concepts, the coupling between the combined logics is sufficiently loose for proving general results about the transfer of decidability: if the connected logics are decidable, then their connection will also be decidable. More importantly in our present context, they allow the heterogeneous combination of logical formalisms without the need to adapt the semantics of the respective components.

Note that the requirement of disjoint domains is not essential for the expressivity of $\mathcal{E}$-connections. What is essential, however, is the disjointness of the *formal languages* of the component logics. What this boils down to is the following simple fact: while more expressive $\mathcal{E}$-connection languages allow to express various degrees of *qualitative* identity, for instance by using number restrictions on links to establish partial bijections, they lack means to express 'proper' *numerical* trans-module identity. This issue, clearly,

[5] A general overview and discussion of counterpart-theoretic semantics can be found in [20].

[6] Thus analysed, the main difference between distributed description logics (DDLs) [21] and various $\mathcal{E}$-connections then lies in the expressivity of the 'link language' $\mathcal{L}$ connecting the different ontologies: while the link language of basic DDL is a certain sub-Boolean fragment of many-sorted $\mathcal{ALC}$, the basic link language of $\mathcal{E}$-connections is many-sorted $\mathcal{ALCI}$ (i.e. $\mathcal{ALC}$ with inverses).

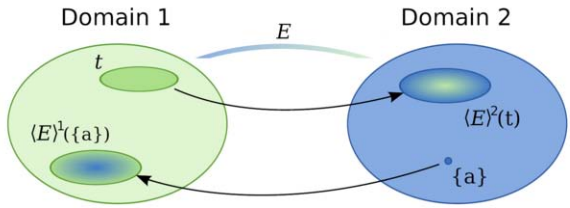


Figure 4. A two-dimensional connection.

is closely related to the problem of trans-world identity well known from counterpart theory; we will expand on this below when introducing $\mathcal{S}$-connections.

For lack of space, we can here only roughly sketch the formal definitions, but compare [19]: we assume that the **languages** L_1 and L_2 of two logics $\mathcal{L}_1$ and $\mathcal{L}_2$ are pairwise disjoint. To form a connection $\mathcal{C}^{\mathcal{E}}(\mathcal{L}_1, \mathcal{L}_2)$, fix a non-empty set $\mathcal{E} = \{E_j \mid j \in J\}$ of binary relation symbols. The **basic $\mathcal{E}$-connection language** is then defined by enriching the respective languages with operators for talking about the link relations. A structure

$$\mathfrak{M} = \langle \mathfrak{W}_1, \mathfrak{W}_2, \mathcal{E}^{\mathfrak{M}} = (E_j^{\mathfrak{M}})_{j \in J} \rangle,$$

where $\mathfrak{W}_i = (W_i, .^{\mathfrak{W}_i})$ is an interpretation of $\mathcal{L}_i$ for $i \in \{1, 2\}$ and $E_j^{\mathfrak{M}} \subseteq W_1 \times W_2$ for each $j \in J$, is called an **interpretation** for $\mathcal{C}^{\mathcal{E}}(\mathcal{L}_1, \mathcal{L}_2)$. Given concepts C_i of logics $\mathcal{L}_i$, $i = 1, 2$, denoting subsets of W_i, the semantics of the basic $\mathcal{E}$-connection operators is[7]

$$(\langle E_j \rangle^1 C_2)^{\mathfrak{M}} = \{x \in W_1 \mid \exists y \in C_2^{\mathfrak{M}} \ : (x, y) \in E_j^{\mathfrak{M}}\}$$
$$(\langle E_j \rangle^2 C_1)^{\mathfrak{M}} = \{y \in W_2 \mid \exists x \in C_1^{\mathfrak{M}} \ : (x, y) \in E_j^{\mathfrak{M}}\}$$

Fig. 4 displays the connection of an ontology with a spatial logic of regions such as $\mathbf{S4_u}$, with a single link relation E interpreted as the relation 'is the spatial extension of'. As follows from the complexity results of [19], $\mathcal{E}$-connections add substantial expressivity and interaction to the component formalism.

In [14], the problem of relating GUM [8] with spatial calculi, using the example of the double-cross calculus DCC [15] for projective relations (orientations), is analysed. The general relation between GUM and DCC is analysed to be a loose coupling as can be adequately modelled by an $\mathcal{E}$-connection. However, two entirely independent layers need to be added for a 'complete' formal representation of a spatial configuration: domain knowledge including naïve physics information is added in a KB $\mathcal{D}$, while contextual information (such as intrinsic orientations, reference system, etc.) is added by a KB $\mathcal{O}$. Both these layers of information are typically formalised in different (heterogeneous) logics. The resulting layered formalism is called **perspectival** $\mathcal{E}$-connections. However, while these extended $\mathcal{E}$-connections formally reflect different layers of a representation, they do not take into account loose couplings in the sense of link-relations that are based on notions of probability or similarity. We next generalise $\mathcal{E}$-connections in this direction.

[7]Note the close resemblance of this definition with the definition of the semantics of existential restrictions in DLs, with the important exception that the former is 'two-sorted'.

4.2. $\mathcal{S}$-Connections: Similarity-based $\mathcal{E}$-Connections

Research on similarity is of a rather broad nature, including work in areas such as philosophy and general cognitive science, (description) logics, bio-informatics, and information retrieval, among others. Technically, the notions of probability, fuzziness, and similarity are closely related, as [23] discusses. For instance, there is no (conceptual or technical) problem with attaching fuzzy-values or probabilities to link-relations: we can say that y is in the spatial extension $E(x)$ of point x with probability $p \in [0,1]$, etc.[8]

Here, we concentrate on modelling a notion of *heterogeneous similarity*, i.e. similarity of objects drawn from conceptually different domains, specified by means of (heterogeneous) similarity measures which are closely modelled on the notions of distance functions and metrics. The notion of similarity-based $\mathcal{E}$-connections defined below thus combines the ideas of $\mathcal{E}$-connections [19], distance logics [26], and similarity logics [27].

4.3. (Heterogeneous) Similarity Spaces

By $\mathbb{R}^+_{0,\infty}$ we denote the positive real numbers including zero and the symbol ∞, denoting infinity. For $i = 1, 2$, we set $\bar{i} = 1$ if $i = 2$ and $\bar{i} = 2$ if $i = 1$.

Definition 1 *A* ***similarity space*** $\mathbb{S} = \langle \mathcal{S}, \int \rangle$ *(*sim*-space for short) consists of a set* $\mathcal{S}$ *together with a* ***similarity measure*** $\int$*, i.e. a function* $\int : \mathcal{S} \times \mathcal{S} \mapsto \mathbb{R}^+_{0,\infty}$ *satisfying* $\int(x,x) = 0$ *for all* $x \in \mathcal{S}$*. In case* $\forall x, y \in \mathcal{S} : \int(x,y) = 0 \iff x = y$ *holds, we call* $\int$ ***discrete****. If* $\int$ *satisfies* $\forall x, y \in \mathcal{S} : \int(x,y) = \int(y,x)$*, we call* $\int$ ***symmetric****, and if it satisfies* $\forall x, y, z \in \mathcal{S} : \int(x,y) + \int(y,z) \geq \int(x,z)$*, we call it* ***triangular****. If* $\mathbb{S}$ *is discrete, symmetric and triangular, and* $\infty \notin \mathsf{range}(\int)$*, it is also called a* ***metric****, and* $\langle \mathcal{S}, \int \rangle$ *is called a* ***metric space****.*

Here, $\int(x,y) = 0$ means that x is *perfectly similar* to y.[9] However, note that perfect similarity implies identity only in the case of discrete spaces. $\int(x,y) < \int(x,z)$ means that x is *more similar* to y than to z, and $\int(x,y) = \int(x,z)$ means that x is *equally similar* to y and z. Moreover, we say that x is *discernibly similar* to y if $\int(x,y) < \infty$ and *indiscernibly similar* otherwise, i.e. if $\int(x,y) = \infty$. For $X, Y \subseteq \mathcal{S}$ sets (rather than just elements), similarity is defined by extending $\int$ as follows:

$$\int(X,Y) := \begin{cases} \inf\{\int(x,y) \mid x \in X, y \in Y\}, & \text{if } X, Y \neq \emptyset \\ \infty, & \text{otherwise} \end{cases}$$

If in fact the minimum exists for all non-empty sets X and Y, $\mathcal{S}$ is also called a min-space, compare [27]. Clearly, whenever a space is finite, it is a min-space.

When relating *different* sets of objects, such as when connecting linguistic ontologies and spatial logics, the above definitions need to be adapted. For simplicity, we here restrict our attention to the case of only two such sets.

[8]This natural idea has been studied for instance in the work of Suzuki on graded accessibility relations [24]. Also, Williamson [25] pursued similar semantic ideas when developing his propositional logics of clarity.

[9]Contrary to other formal approaches to similarity, closeness in the similarity space (i.e. a low value of the similarity measure) corresponds to high similarity: this intuition derives from the spatial interpretation of metric spaces.

Definition 2 *A (2-dim)* ***heterogeneous similarity space*** *(*hsim*-space for short) is a quadruple* $\mathbb{H} = \langle \mathbb{S}_1, \mathbb{S}_2, \int_1^2, \int_2^1 \rangle$ *consisting of, for* $i = 1, 2$, sim*-spaces* $\mathbb{S}_i = \langle \mathcal{S}_i, \int_i \rangle$*, and* ***heterogeneous similarity measures*** $\int_i^{\bar{i}} : \mathcal{S}_i \times \mathcal{S}_{\bar{i}} \mapsto \mathbb{R}^+_{0,\infty}$*.* $\mathbb{H}$ *is* ***het-symmetric*** *if for all* $x \in \mathcal{S}_i$ *and all* $y \in \mathcal{S}_{\bar{i}}$ *we have* $\int_i^{\bar{i}}(x, y) = \int_{\bar{i}}^i(y, x)$ *(for* $i = 1, 2$*). It is* ***het-triangular*** *if for all* $x, z \in \mathcal{S}_i$ *and* $y \in \mathcal{S}_{\bar{i}}$ *we have* $\int_i^{\bar{i}}(x, y) + \int_{\bar{i}}^i(y, z) \geq \int_i(x, z)$ *(for* $i = 1, 2$*).*

In the heterogeneous case, *perfect similarity* now means that $x \in \mathcal{S}_1$ and $y \in \mathcal{S}_2$ are indistinguishable from the perspectives of both similarity measures, $\int_1$ and $\int_2$.[10]

4.4. Counterparts in Similarity-based $\mathcal{E}$-Connections

Note that, in this setting, the problems of transworld identity and counterparthood can be neatly separated: transworld identity may be taken to be synonymous with perfect similarity as defined above. Counterparthood understood as *maximal similarity* is a looser notion, and may be explicated by the following principle (see [28]).

> For $x \in \mathcal{S}_i$ and $y \in \mathcal{S}_{\bar{i}}$, y is a counterpart of x only if nothing in $\mathcal{S}_{\bar{i}}$ is more similar to x as it is in $\mathcal{S}_i$ than is y as it is in $\mathcal{S}_{\bar{i}}$.

We take this principle as the defining criterion for counterparthood in similarity spaces:

Definition 3 (Counterparts) *Let* $\mathbb{H} = \langle \mathbb{S}_1, \mathbb{S}_2, \int_1^2, \int_2^1 \rangle$ *be a* hsim*-space. We call* $b_{\bar{i}} \in \mathcal{S}_{\bar{i}}$ *an* $\bar{i}$***-counterpart of*** $a_i \in \mathcal{S}_i$ *if* $\int_i^{\bar{i}}(a_i, b_{\bar{i}}) = \inf\{\int_i^{\bar{i}}(a_i, b) \mid b \in \mathcal{S}_{\bar{i}}\} < \infty$*, which we also write as* $\mathsf{Cp}_i^{\bar{i}}(a_i, b_{\bar{i}})$*. This gives us two relations:* $\mathsf{Cp}_i^{\bar{i}} \subseteq \mathcal{S}_i \times \mathcal{S}_{\bar{i}}, i = 1, 2$*. Moreover, for* $X \subseteq \mathcal{S}_i$*, we denote by* $\mathsf{Cp}_i^{\bar{i}}(X)$ *the set* $\{y \in \mathcal{S}_{\bar{i}} \mid \exists x \in X.\mathsf{Cp}_i^{\bar{i}}(x, y)\}$*.*

Note that counterparts thus defined may or may not be unique. Moreover, $b_{\bar{i}}$ may be an $\bar{i}$-counterpart of a_i without a_i being an i-counterpart of $b_{\bar{i}}$; counterparthood is *directional*. Although counterparts need not be unique, in applications it is often desirable to select amongst the elements with maximal similarity a unique element, according to certain *external* criteria. We here solve this problem by incorporating into the structures an explicit choice function selecting a counterpart.

Definition 4 (Counterpart choice) *A* hsim*-space* ***with choice*** *is a triple* $\langle \mathbb{H}, \lambda_1, \lambda_2 \rangle$*, where* $\mathbb{H} = \langle \mathbb{S}_1, \mathbb{S}_2, \int_1^2, \int_2^1 \rangle$ *is a* hsim*-space, and, for* $i = 1, 2$*,* $\lambda_i : \mathcal{S}_i \longrightarrow \mathsf{Cp}_i^{\bar{i}}(\mathcal{S}_i)$ *are* ***choice functions*** *such that, for all* $x \in \mathcal{S}_i$*, we have that* $\lambda_i(x) \subseteq \mathsf{Cp}_i^{\bar{i}}(x)$ *is a singleton.*

Of course, often the λ_i are uniquely determined by the similarity measures $\int_i^{\bar{i}}$, in which case we call λ_i a **deterministic choice function**. Apart from the elements with maximal similarity, i.e. the counterparts, it is also of interest to be able to refer to elements of a foreign domain that are similar *to some degree* (i.e. discernibly similar). This can be achieved by simulating the notion of *link relation* from $\mathcal{E}$-connections as follows:

Definition 5 (Link-relation) *Given a* hsim*-space* $\mathbb{H} = \langle \mathbb{S}_1, \mathbb{S}_2, \int_1^2, \int_2^1 \rangle$*, we define the* ***induced link relations*** $E^1_{\mathbb{H}}, E^2_{\mathbb{H}}, E_{\mathbb{H}} \subseteq \mathcal{S}_1 \times \mathcal{S}_2$ *by setting, for all* $x \in \mathcal{S}_1$ *and* $y \in \mathcal{S}_2$*:*

$$E^1_{\mathbb{H}}(x, y) \iff \textstyle\int_1^2(x, y) < \infty; \quad E^2_{\mathbb{H}}(x, y) \iff \int_2^1(y, x) < \infty;$$

[10] The notion of *discrete* similarity measure makes no immediate sense in the heterogeneous case as identity is not available. However, the notion can be 'simulated' by replacing identity with an independently defined notion of trans-module identity, 'equalising' cross-domain elements whilst respecting the similarity measures.

$$E_{\mathbb{H}}(x,y) \iff \min\big(\int_1^2(x,y), \int_2^1(y,x)\big) < \infty (= E^1_{\mathbb{H}} \cup E^2_{\mathbb{H}}).$$

Intuitively, the relation $E_{\mathbb{H}}(x,y)$ holds if x and y are discernibly similar from at least one 'viewpoint', and $E^i_{\mathbb{H}}(x,y)$ holds if x and y are discernibly similar from the point of view of $\int_i^{\bar{i}}$. We can now recover standard $\mathcal{E}$-connections in the following sense:

Proposition 6 *For every $\mathcal{E}$-connection model $\mathfrak{M} = \langle \mathfrak{W}_1, \mathfrak{W}_2, E^{\mathfrak{M}} \rangle$ there is a* hsim*-space $\mathbb{H} = \langle \mathbb{S}_1, \mathbb{S}_2, \int_1^2, \int_2^1 \rangle$ such that $E_{\mathbb{H}} = E^{\mathfrak{M}}$.*

PROOF. Fix $\mathfrak{M} = \langle \mathfrak{W}_1, \mathfrak{W}_2, E^{\mathfrak{M}} \rangle$. Essentially, we need to show that induced link relations can be arbitrary relations: set, for $x \in \mathcal{S}_1$ and $y \in \mathcal{S}_2$

$$\int_1^2(x,y) = \int_2^1(y,x) = \begin{cases} 0, \text{ if}(x,y) \in E^{\mathfrak{M}} \\ \infty, \text{ otherwise} \end{cases}$$

Clearly, $E^{\mathfrak{M}} = E_{\mathbb{H}}$. □

4.5. Similarity Bridge Logic

So far, we have only (generically) described the model-theory of similarity based $\mathcal{E}$-connections. Whilst the component logics can be assumed to be given, we need to describe possibilities to (syntactically) define the bridge logic of such $\mathcal{E}$-connections. As we have mentioned above, the spectrum of languages that can be used for this can be varied almost arbitrarily. We here describe a language that we consider basic in that it reflects the essential features of the underlying structures. We assume two logics $\mathcal{L}_1$ and $\mathcal{L}_2$ are given, with disjoint sort structure. For $\mathcal{L}_i, i = 1, 2$, assume object names a_i (denoting elements of the domains) and terms A_i (denoting subsets of the domains) belonging to the respective logics are given.

Fix a hsim-space with choice $\mathbb{H} = \langle \mathbb{S}_1, \mathbb{S}_2, \int_1^2, \int_2^1, \lambda_1, \lambda_2 \rangle$, and assume, for $i = 1, 2$, the logics $\mathcal{L}_i$ are interpreted in models $\mathfrak{M}_i$ over sim-spaces $\mathbb{S}_i$, i.e. $\mathsf{dom}(\mathfrak{M}_i) \supseteq \mathsf{dom}(\mathbb{S})$.

Definition 7 *The **basic similarity bridge logic** $\mathcal{B}_{\mathsf{sim}}(\mathcal{L}_1, \mathcal{L}_2)$ contains:*

- ***projection operators**: $\langle E \rangle^{\bar{i}} A_i$ and $\langle E \rangle^{\bar{i}} a_i$, for $i = 1, 2$ and $E \in \{E^1_{\mathbb{H}}, E^2_{\mathbb{H}}, E_{\mathbb{H}}\}$.*

These are the basic $\mathcal{E}$-connection-operators (with the standard semantics), with link-relations E inherited from the similarity measures as defined in Def. 5.

- ***counterpart operators**: $\langle C \rangle^{\bar{i}} A_i$ and $\langle C \rangle^{\bar{i}} a_i$, $i = 1, 2$.*

Given the term A_i of logic $\mathcal{L}_i$, the operator $\langle C \rangle^{\bar{i}} A_i$ yields the set of all counterparts of elements of A_i, i.e.

$$(\langle C \rangle^{\bar{i}} A_i)^{\mathfrak{M}} = \{y \in \mathcal{S}_{\bar{i}} \mid \exists x \in A_i^{\mathfrak{M}_i} \textit{ and } \mathsf{Cp}_i^{\bar{i}}(x,y)\},$$

and similarly for object names.

- ***choice operators**: $\langle\lambda\rangle^{\bar{i}} a_i, i = 1, 2$.*

These pick out the unique counterpart of a_i as a singleton subset whenever there are counterparts, and returns $\perp^{\bar{i}}$ otherwise, i.e.

$$(\langle\lambda\rangle^{\bar{i}} a_i)^{\mathfrak{M}} = \begin{cases} \{\lambda_i(a_i{}^{\mathfrak{M}_i})\}, & \textit{if defined} \\ \perp^{\bar{i}}, & \textit{otherwise} \end{cases}$$

- ***heterogeneous similarity operators**: $\langle\downdownarrows\rangle^i(A_1, A_2)$, $\langle\upuparrows\rangle^i(A_1, A_2)$, $i = 1, 2$.*

Intuitively, $\langle\downdownarrows\rangle^1(A_1, A_2)$ gives a term of $\mathcal{L}_1$, consisting of all those members of $\mathbb{S}_1$ that are closer to something in A_1 than to any of A_2's counterparts in $\mathbb{S}_1$ (similarity is evaluated locally*). Conversely, $\langle\upuparrows\rangle^1(A_1, A_2)$ gives a term of $\mathcal{L}_1$, consisting of all those members of $\mathbb{S}_1$ all of whose counterparts are closer to some of A_1's counterparts than to any element in A_2 (similarity is evaluated* externally *for the counterparts). Formally, the semantics is as follows, for $i = 1, 2$:*

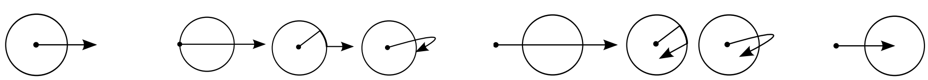
$$\big(\langle\downdownarrows\rangle^i(A_1, A_2)\big)^{\mathfrak{M}} = \{y \in \mathcal{S}_i \mid \int_i\big(y, A_i{}^{\mathfrak{M}_i} \cap \mathcal{S}_i\big) < \int_i\big(y, \mathsf{Cp}^i_{\bar{i}}(A_{\bar{i}}{}^{\mathfrak{M}_{\bar{i}}}) \cap \mathcal{S}_i\big)\} \textit{ and}$$

$$\big(\langle\upuparrows\rangle^i(A_1, A_2)\big)^{\mathfrak{M}} = \{y \in \mathcal{S}_i \mid \int_{\bar{i}}\big(\mathsf{Cp}^{\bar{i}}_i(y), \mathsf{Cp}^{\bar{i}}_i(A_i{}^{\mathfrak{M}_i}) \cap \mathcal{S}_{\bar{i}}\big) < \int_{\bar{i}}\big(\mathsf{Cp}^{\bar{i}}_i(y), A_{\bar{i}}{}^{\mathfrak{M}_{\bar{i}}} \cap \mathcal{S}_{\bar{i}}\big)\}$$

As in standard $\mathcal{E}$-connections, we assume that these operators yield new terms of the respective logics to which the operators of those logics can then be further applied. This process, inductively, defines the **basic similarity language** of **$\mathcal{S}$-connections**.[11]

5. $\mathcal{S}$-Connection for Directions and Regions in Language and Space

An example how $\mathcal{S}$-connections can be used to relate natural language and spatial logics is outlined in the following. Here, GUM is '$\mathcal{S}$-connected' with the *9$^+$-intersection for topological relations between a directed line segment (DLine) and a region* (9$^+$-calculus) [29]. Similarities between examples of linguistic motion descriptions in GUM and related 9$^+$-calculus examples are presented. The linguistic descriptions (a) "They went out of the park", (b) "They left the park" are defined by source:GeneralDirectional and (c) "They entered the park" is defined by destination:GeneralDirectional in GUM. GeneralDirectional defines directions of motions or orientations determined by the relatum and specified by the relations source and destination. (For reasons of space and simplicity, the reader is referred to [8] for further documentation.)



5.1: Ex. x 5.2: Closest Neighbours (1) 5.3: Closest Neighbours (2) 5.4: Ex. y

Figure 5. Directed line segments and possible relations with a region.

While actor and relatum are linguistically described by "they" and "park" respectively, their counterparts in the 9$^+$-calculus are the *DLine* for the motion of "they" and

[11]For simplicity, we have here defined only the 'concept language' of $\mathcal{S}$-connections. Assertions and KBs can be defined in the same way as for $\mathcal{E}$-connections, with the addition of object statements allowing to explicitly declare the similarity between named objects such as $\mathsf{sim}^{\bar{i}}_i(a_i, a_{\bar{i}}) = 3$, with the obvious semantics.

the *region* for "park". A sample of 9^+-models are illustrated in Fig. 5. The topological dependence in Fig. 5.1 between the DLine and the region is defined as the most similar realisation for a and b. Given the neighbourhood graph for x by the 9^+-calculus, Fig. 5.2 shows its direct neighbours. Some of them are also elements with high similarity for GUM's source:GeneralDirectional. Fig. 5.3 shows neighbours directly related to the first neighbours in Fig. 5.2. Those are, however, rather indiscernibly similar with a and b. As a and b are equally instantiated in GUM, they are not distinguishable and $\int_{GUM}(a, b) = 0$. A set of similar 9^+-elements for a and b are illustrated in Fig. 6, ordered by decreasing similarity. The first one (denoted x) is the counterpart. Clearly, a and b are equally similar to x, and so $\mathsf{sim}_1^2(a, x) = \mathsf{sim}_1^2(b, x)$.

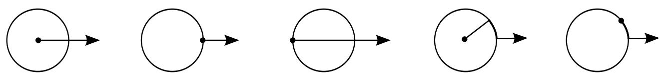


Figure 6. 9^+-calculus counterparts for a "They left the park"

Conversely, the counterpart of c "They entered the park" is y illustrated in Fig. 5.4. Here, the DLine has exactly the opposite direction of the DLine in x. y is also indiscernibly similar to a and b. Hence, the $\mathcal{S}$-connections between GUM and the 9^+-intersections differ in similarities of linguistic descriptions and topological relationships, as indicated by the neighbourhood relation and equal specifications in GUM. An excerpt from these similarity relations and $\mathcal{S}$-connections is illustrated in Fig. 7.

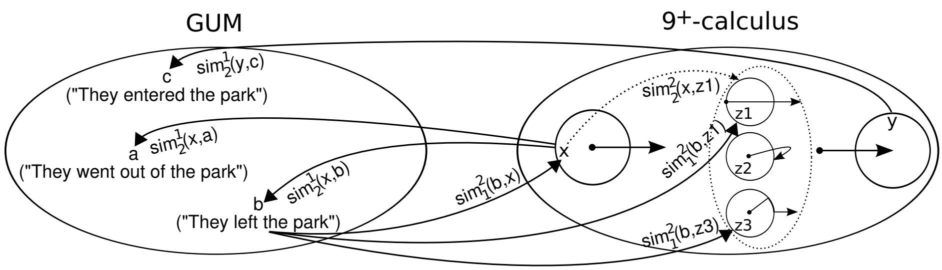


Figure 7. Example of $\mathcal{S}$-connections between GUM and 9^+-calculus. Similar counterparts are $\mathsf{sim}_1^2(b, x)$ (from GUM to SL), $\mathsf{sim}_2^1(x, b) = \mathsf{sim}_2^1(x, a)$ (from SL to GUM), and $\mathsf{sim}_2^2(x, z1)$ (similarities within SL).

6. Discussion

We have introduced $\mathcal{S}$-connections as an extension of $\mathcal{E}$-connections adding similarity measures across domains and corresponding formal apparatus to interpret these measures. We have shown that this framework is well-suited to deal with the problem of relating linguistic semantics and spatial logics whilst respecting the uncertainties or underspecifications that are involved in their relationship. Various examples illustrating how language underspecifies spatial information are given together with aspects causing such underspecifications. However, further investigations will need to elaborate on specific definitions of such measures and on algorithms for calculating them, based on external linguistic and spatial factors, as described for instance in [30].

As concerns the general theory of $\mathcal{S}$-connections, there are many interesting open problems. Most obviously, decidability and complexity issues for various component and bridge logics should be addressed, and an axiomatisation of the basic logic of $\mathcal{S}$-connections should be given (extending the results of [31]). Other interesting areas are the following: (i) analyse structural properties on the interplay between 'local' and 'global' (i.e. heterogeneous) similarity measures; (ii) formulate various notions of qualitative (trans-module) identity compatible with similarity measures; (iii) investigate notions such as transitivity of similarity that have a different flavour in the setting of $\mathcal{S}$-connections.

To elaborate just on the last point, note that the triangular inequality gives us a particular (quantitative) version of transitivity of similarity. Namely, if a is x-similar to b and b is y-similar to c, then a is at least $x + y$-similar to c. Stricter transitivity assertions could, of course, be defined, and would correspond to global '*elasticity*' restrictions on the similarity space. However, similarities between entities in a spatial model will not always directly entail corresponding similarities between spatial language and spatial logic configurations, as indicated by the example in Fig. 5. Therefore, a careful analysis of appropriate transitivity principles for the interplay between spatial language and spatial logics will be necessary.

Acknowledgements

We gratefully acknowledge the financial support of the Deutsche Forschungsgemeinschaft through the Collaborative Research Centre SFB/TR 8. The authors would also like to thank J. Bateman, T. Mossakowski, and T. Tenbrink for fruitful discussions.

References

[1] L. Talmy. *Toward a Cognitive Semantics: Volume I Toward a Cognitive Semantics: Concept Structuring Systems*. MIT Press, 2000.

[2] A. Herskovits. *Language and Spatial Cognition: an interdisciplinary study of the prepositions in English*. Studies in Natural Language Processing. Cambridge University Press, 1986.

[3] T. Tenbrink. *Space, time, and the use of language: An investigation of relationships*. Mouton de Gruyter, 2007.

[4] A. G. Cohn and S. M. Hazarika. Qualitative spatial representation and reasoning: an overview. *Fundamenta Informaticae*, 43:2–32, 2001.

[5] A. G. Cohn and J. Renz. Qualitative spatial representation and reasoning. In F. van Harmelen et al., editor, *Handbook of Knowledge Representation*, pages 551–596. Elsevier, 2007.

[6] R. Ross, H. Shi, T. Vierhuff, B. Krieg-Brückner, and J. A. Bateman. Towards Dialogue Based Shared Control of Navigating Robots. In *Spatial Cognition IV*, pages 478–499. Springer, 2005.

[7] M. A. K. Halliday and C. M. I. M. Matthiessen. *Construing experience through meaning: a language-based approach to cognition*. Cassell, 1999.

[8] J. A. Bateman, T. Tenbrink, and S. Farrar. The Role of Conceptual and Linguistic Ontologies in Discourse. *Discourse Processes*, 44(3):175–213, 2007.

[9] L. Talmy. The fundamental system of spatial schemas in language. In B. Hampe, editor, *From Perception to Meaning: Image Schemas in Cognitive Linguistics*, pages 37–47. Mouton de Gruyter, 2006.

[10] B. Bennett and P. Agarwal. Semantic Categories Underlying the Meaning of 'Place'. In *COSIT-07*, pages 78–95. Springer, 2007.

[11] I. Horrocks, O. Kutz, and U. Sattler. The Even More Irresistible $\mathcal{SROIQ}$. In *Proc. of KR-06*, pages 57–67, 2006.

[12] S. C. Levinson. *Space in Language and Cognition: Explorations in Cognitive Diversity*. Cambridge University Press, 2003.

[13] R. W. Langacker. *Foundations in Cognitive Grammar. Volume 1, Theoretical Prerequisites*. Stanford University Press, 1987.

[14] J. Hois and O. Kutz. Natural Language meets Spatial Calculi. In Freksa et al., editor, *Spatial Cognition VI*, Freiburg, Germany, 2008. Springer.

[15] C. Freksa. Using orientation information for qualitative spatial reasoning. In *Theories and methods of spatio-temporal reasoning in geographic space*, volume 639 of *LNCS*, pages 162–178. 1992.

[16] R. Moratz, F. Dylla, and L. Frommberger. A Relative Orientation Algebra with Adjustable Granularity. In *IJCAI Workshop on Agents in Real-Time and Dynamic Environments*, 2005.

[17] Roland Billen and Eliseo Clementini. *Projective Relations in a 3D Environment*, pages 18–32. 2006.

[18] D. K. Lewis. Counterpart Theory and Quantified Modal Logic. *Journal of Philosophy*, 65:113–126, 1968.

[19] O. Kutz, C. Lutz, F. Wolter, and M. Zakharyaschev. $\mathcal{E}$-Connections of Abstract Description Systems. *Artificial Intelligence*, 156(1):1–73, 2004.

[20] M. Kracht and O. Kutz. Logically Possible Worlds and Counterpart Semantics for Modal Logic. In D. Jacquette, editor, *Philosophy of Logic, Handbook of the Philosophy of Science*, volume 5, pages 943–996. Elsevier, Amsterdam, 2007.

[21] A. Borgida and L. Serafini. Distributed Description Logics: Assimilating Information from Peer Sources. *Journal of Data Semantics*, 1:153–184, 2003.

[22] B. Cuenca Grau, B. Parsia, and E. Sirin. Ontology Integration Using $\mathcal{E}$-connections. In H. Stuckenschmidt and S. Spaccapietra, editors, *Ontology Modularization*. Springer, 2008. Forthcoming.

[23] P. Hájek. Why Fuzzy Logic? In D. Jacquette, editor, *A Companion to Philosophical Logic*, pages 595–607. Blackwell, 2002.

[24] N.-Y. Suzuki. Kripke frames with graded accessibility and fuzzy possible world semantics. *Studia Logica*, 59:249–269, 1997.

[25] T. Williamson. *Vagueness*. The Problems of Philosophy. Routledge, London, 1994.

[26] O. Kutz, H. Sturm, N.-Y. Suzuki, F. Wolter, and M. Zakharyaschev. Logics of metric spaces. *ACM Transactions on Computational Logic (TOCL)*, 4(2):260–294, 2003.

[27] M. Sheremet, D. Tishkovsky, F. Wolter, and M. Zakharyaschev. A Logic for Concepts and Similarity. *J. of Logic and Computation*, 17(3):415–452, 2007.

[28] G. Forbes. *The Metaphysics of Modality*. Oxford Clarendon Press, Oxford, 1985.

[29] Y. Kurata and M. Egenhofer. The 9^+-intersection for topological relations between a directed line segment and a region. In B. Gottfried, editor, *1st Workshop on Behaviour Monitoring and Interpretation, KI 2007*, pages 62–76, 2007.

[30] M. Kracht. *Language and Space*. 2008. Book manuscript.

[31] O. Kutz, H. Sturm, N.-Y. Suzuki, F. Wolter, and M. Zakharyaschev. Axiomatizing distance logics. *Journal of Applied Non-Classical Logic*, 12:425–440, 2002.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008
© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-280

An Ontology for Grounding Vague Geographic Terms

Brandon BENNETT, David MALLENBYand Allan THIRD
School of Computing,
University of Leeds, UK
e-mail: {davidm,brandon,thirda}@comp.leeds.ac.uk

Abstract. Many geographic terms, such as "river" and "lake", are vague, with no clear boundaries of application. In particular, the spatial extent of such features is often vaguely carved out of a continuously varying observable domain. We present a means of defining vague terms using *standpoint semantics*, a refinement of the philosophical idea of supervaluation semantics. Such definitions can be grounded in actual data by geometric analysis and segmentation of the data set. The issues raised by this process with regard to the nature of boundaries and domains of logical quantification are discussed. We describe a prototype implementation of a system capable of segmenting attributed polygon data into geographically significant regions and evaluating queries involving vague geographic feature terms.

Keywords. Vagueness, Geographic Entities, Query Answering

1. Introduction

In recent years increasing attention has been paid to the ontology of geographic entities. A major motivation for this has been the recognition that the implementation of computational Geographic Information Systems (GIS) which can support functionality for sophisticated data manipulation, querying and display requires robust and detailed specification of the semantics of geographic entities and relationships. A second, more philosophical, motivation for attention to this domain is that it presents a concrete manifestation of many ontological subtleties. For instance issues of individuation, identity and vagueness arise in abundance, when one tries to give precise specifications of the meanings implicit in geographic terminology [1,2,3,4].

Our concerns in this paper will relate to both these motivations. On the one hand, we will examine the particular ontological issues associated with interpretation of vague geographic feature terms (especially hydrological terms such as 'lake' and 'river) and will outline how the general semantic framework of *standpoint semantics* can be applied to provide a framework within which such vagueness can be represented explicitly. We shall also see that when deployed in conjunction with a geometry-based theory of feature segmentation, this semantics gives an account of how vague features are individuated with respect to the material structure of the world. On the other hand, we shall also be very much concerned with the implementation of certain GIS functionality for which a coherent theory of vagueness and its relation to individuation is a necessary pre-requisite.

We look specifically at the problem of interpreting logical queries involving vague predicates with respect to a geographic dataset. We shall assume that such data takes a typical form consisting of a set of 2-dimensional polygons, each of which is associated with one or more labels describing the type of region that the polygon represents. This is a simplification of geographic data in general, which will often include other types of information such as point or line entities, altitudes, additional cartographic entities such as icons or textual strings and meta-annotations relating to the provenance or accuracy of data items. Moreover, the data will not normally consist simply of a set of entities but a complex data structure supporting indexing and various kinds of computational manipulation of data elements. Nevertheless, labelled 2-dimensional polygons form the core of most real geographic information systems.

The structure of the rest of the paper is as follows. In section 2 we present an overview of the basic theory of standpoint semantics, which is a refinement of supervaluation semantics. Section 3 considers the ontological principles that govern the ways in which one can divide up the geographic realm into distinct regions corresponding to geographic features. In section 4 we consider the implementation of a geographic query interpretation system and see that severe difficulties arise regarding finding an appropriate computationally tractable domain of quantification. We shall see that finding a solution to this problem requires a theory of individuation (such as was developed in section 3). Section 5 then looks in detail at the implementation of our prototype system, which provides a limited proof of concept of our theoretical analysis. Finally, concluding remarks and discussion of future work are given in section 6.

2. Standpoint Semantics

In making an assertion or a coherent series of assertions, one is taking a *standpoint* regarding the applicability of linguistic expressions to describing the world. Such a standpoint depends partly on one's beliefs about the world. This epistemic component will *not* be considered in the current paper: we shall assume for present purposes that one has correct knowledge of the world — albeit at a certain level of granularity (which in the context of geographic information is likely to be rather coarse). The other main ingredient of a standpoint, which we *will* be concerned with here, is that it involves a linguistic judgement about the criteria of applicability of words to a particular situation. This is especially so when some of the words involved are vague. For instance, one might take the standpoint that a certain body of water should be described as a 'lake', whereas another smaller water-body should be described as a 'pond'.

The notion of 'standpoint' is central to our analysis of vagueness. Vagueness is sometimes discussed in terms of different people having conflicting opinions about the use of a term. This is somewhat misleading since even a person thinking privately may be aware that an attribution is not clear cut. Hence a person may change their standpoint. Moreover this is not necessarily because they think they were mistaken. It can just be that they come to the view that a different standpoint might be more useful for communication purposes. Different standpoints may be appropriate in different circumstances. The core of standpoint semantics does not explain why a person may hold a particular standpoint or the reasons for differences or changes of standpoint, although a more elaborate theory dealing with these issues could be built upon the basic formalism.

In taking a standpoint, one is making somewhat arbitrary choices relating to the limits of applicability of natural language terminology. But a key feature of the theory is that all assertions made in the context of a given standpoint must be mutually consistent in their use of terminology. Hence, if I take a standpoint in which I consider Tom to be tall, then if Jim is greater in height than Tom then (under the assumption that height is the only attribute relevant to tallness) I must also agree with the claim that Jim is tall.

Our *standpoint semantics* is both a refinement and an extension of the *supervaluation* theory of vagueness that has received considerable attention in the philosophical literature (originating with [5]). Supervaluation semantics enables a vague language to be logically interpreted by a set of possible precise interpretations (*precisifications*). This provides a very general framework within which vagueness can be analysed within a formal representation. Here we do not have space to give a full account of supervaluation semantics. Detailed expositions can be found in the philosophical literature (e.g. [6]).

By itself, supervaluation semantics simply models vagueness in terms of an abstract set of possible interpretations, but gives no analysis of the particular modes of semantic variability that occur in the meanings of natural language vocabulary. A key idea of standpoint semantics is that the range of possible precisifications of a vague language can be described by a (finite) number of relevant *parameters* relating to objectively observable properties; and the limitations on applicability of vocabulary according to a particular standpoint can be modelled by a set of *threshold values*, that are assigned to these parameters. To take a simple example, if the language contains a predicate Tall (as applicable to humans), then a relevant observable is 'height'. And to determine a precisification of Tall we would have to assign a particular threshold value to a parameter, which could be called tall_human_min_height.[1] In general a predicate can be dependent on threshold valuations of several different parameters (e.g. Lake might depend on both its area and some parameter constraining its shape.) Thus, rather than trying to identify a single measure by which the applicability of a predicate may be judged, we allow multiple vague criteria to be considered independently.

In the current paper (as in several previous papers on this topic [4,8,9]) we shall assume that standpoints can be given a model theoretic semantics by associating each standpoint with a threshold valuation. In so far as standpoints may be identified with an aspect of a cognitive state, this idea is perhaps simplistic. It is implausible that an agent would ever be committed to any completely precise value for a threshold demarcating the range of applicability of a vague predicate. Cognitive standpoints are more plausibly associated with constraints on a range of possible threshold values (e.g. if I call someone tall then my claim implies an upper bound on what I consider to be a suitable threshold for tallness — the threshold cannot be higher than the height of that person) rather than exact valuations of thresholds.[2] But in the context of cartographic displays, we may more plausibly propose that any useful depiction of geographic entities corresponding to geographic terms should be determined by application of precise criteria associated

[1]Vague adjectives tend to be context sensitive in that an appropriate threshold value depends on the category of things to which the adjective is applied. This is an important aspect of the semantics of vague terminology but is a side issue in relation to our main concerns in the current paper. Here we shall assume that vague properties are applied uniformly over the set of things to which they can be applied. To make this explicit we could always use separate properties such as Tall-Human and Tall-Giraffe, although we won't actually need to do this for present purposes. A formal treatment of category dependent vague adjectives is given in [7].

[2]This elaboration of the status of standpoints in relation to thresholds is being developed in a separate strand of research.

with the term, and that such criteria require a definite value to be associated with every threshold parameter.

A *threshold valuation* appropriate for specifying a standpoint in relation to the domain of hydrographic geography might be represented by:

$$V = [\text{pond\_vs\_lake\_area\_threshold} = 200m^2,\ \text{river\_min\_linearity\_ratio} = 3,\ ...]$$

Here one parameter determines a cut-off between ponds and lakes in terms of their surface area and another fixes a parameter indicating a linearity[3] requirement used to characterise rivers.

3. Geographic Entities and their Boundaries

As noted by Smith and Mark in [3], the geographic domain is distinctive in that typical geographic objects are attached to the world and are not easily demarcated in the way that physically detached objects such as organisms and artifacts can be. Thus the individuation of geographic features is ontologically problematic. Previously, Smith [10,11] had drawn attention to a distinction between of *bona fide* and *fiat* boundaries:

> *Fiat boundaries are boundaries which owe their existence to acts of human decision or fiat, to laws or political decrees, or to related human cognitive phenomena. Fiat boundaries are ontologically dependent upon human fiat. Bona fide boundaries are all other boundaries.* [11]

A paradigm case of a fiat boundary is the border of a country whose location does not depend on any physical boundary in the world.[4] In [3] it is argued that, in so far as they may be said to exist at all, the boundaries of mountains must be fiat because they rely on human judgement for their demarcation. Whilst we have no objection to this use of terminology, we believe that there is a significant difference between the national border and mountain cases. Although any particular demarcation of a border around a mountain is certainly dependent on human judgement, the range of reasonable boundaries is also to a large extent determined by the lie of the land.

In order to understand this distinction more clearly, it will be instructive first to consider another kind of boundary, which we call an *implicit geometrical boundary*. Such

[3]Note that we use the term 'linearity' to refer to elongation of form rather than straightness. Thus we would describe a river as linear, even though it may bend and wiggle. A geometric characterisation of linearity of this form has been presented in previous papers [8].

[4]Of course particular national boundaries may be aligned to physical boundaries such as the banks of rivers but this is a contingent circumstance.

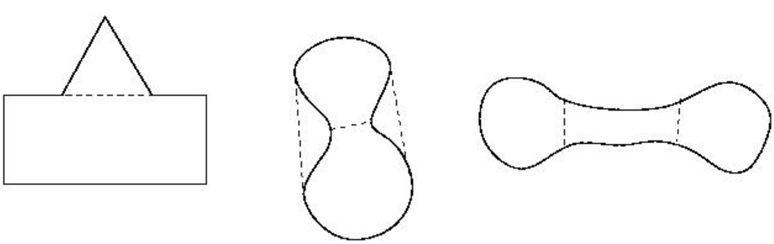


Figure 1. Implicit geometric boundaries.

a boundary does not lie upon an actual discontinuity in the fabric of the world but follows a line that is determined by the spatial configuration of other boundaries, which may be either bona fide or fiat (or a combination of both). Such boundaries are depicted in Figure 1. On the left we see a region within which there is an implicit boundary between a rectangular portion and a triangular projection. The middle region involves a 'neck' flanked by concavities, and these features also imply certain geometric boundaries.

In the region on the right, implicit boundaries are not so clear cut. In describing the region one may be inclined to mention two bulbous parts joined by an elongated section. This suggests the existence of implied boundaries between these three portions. These are examples of *vague boundaries* whose course is hinted at, but not completely determined, by the geometric form of a concrete boundary.

This analysis suggests a four-fold classification of kinds of boundary:

- *Bona fide* boundaries between matter or terrain types.
- *Fiat* boundaries imposed on the world by conscious agents
- *Implicit Geometrical* boundaries determined geometrically in relation to bona fide and/or fiat boundaries.
- *Vague* boundaries, which can be made precise in relation to some standpoint taken on an appropriate precisification of vague properties or relations. The resulting precise properties/relations will then determine a geometrical boundary (which will be demarcated in relation to bona fide and/or fiat boundaries).

The latter two types could be regarded as special cases of *bona fide* or *fiat* boundaries. However, it is not completely clear to which camp they should be assigned. Whether implicit geometric boundaries are considered *bona fide* or *fiat* depends upon whether one takes a Platonist or constructivist view of the existence of geometrical entities. It may be argued that vague boundaries must involve an element of human judgement and hence must be *fiat*. However, if one takes a Platonist view of implied geometric boundaries, then vague boundaries also have a *bona fide* underpinning.

Meta-terminological confusion notwithstanding, it is clear that many kinds of natural geographic feature have vague boundaries and that the demarcation of these is determined by a combination of physical properties of the world and human judgement. We believe that the way that this occurs can be explained by standpoint semantics.

This is well illustrated by consideration of the division of a water system into lakes and rivers. As described in [8,9], such a segmentation can be achieved by specifying geometric predicates that can identify linear/elongated *stretches* of a water system (as represented by polygons) and distinguish these from from expansive (lake-like) regions of the system. Indeed these have been implemented in prototype GIS software (GEOLOG). A feature of these predicates is that they depend on a small number[5] of parameters, for which specific values must be chosen to obtain a segmentation into lakes and rivers. This parameterised variability of geometry-based predicates can be directly described within the framework of standpoint semantics. Each choice of parameters given to the computational segmentation procedure corresponds to a standpoint taken with respect to the interpretation of the terms 'river' and 'lake'.

Of course more factors are relevant to the meanings of these natural language terms; so this shape-based characterisation is only part of a full explanation of the usage of

[5]In our simplest implementation there is just one such parameter, but better results have been obtained by adding a second parameter.

hydrographic terms. For instance, water flow is such an essential part of our concept of river that it might appear that no satisfactory characterisation of rivers could omit this aspect. But, GIS and other cartographic data rarely includes flow information (such information is hard to obtain and to depict); and yet, it seems that humans usually have little difficulty in identifying rivers represented in a 2-dimensional map display. One explanation for this is that, although flow is an important criterion in its own right, the dynamic behaviour of water distributed over an uneven but approximately horizontal surface is closely correlated (due to physical laws) with the geometry of the projection of the water system onto the horizontal plane. Thus, given our knowledge of the way the world works, we can infer a lot about flow just from a 2-dimensional representation of a water system.

Having said that, we would in future like to incorporate flow into our hydrographic ontology and believe that can be done within the general framework that we propose. A simple approach would be to take a field of flow vectors (this would have to be interpolated from some set of data points) and segment the water system according to a threshold on flow magnitude, so that we would obtain polygons labelled as either flowing or (comparatively) still. We could then define types of hydrographic feature in terms of a combination of both shape-based and flow-based characteristics. (We could also investigate correlations between the two types of characteristic.)

In many cases there is ambiguity with regard to which objective properties are relevant to a particular natural language term (e.g. is salinity relevant to lake-hood). Such controversy may be modelled by allowing standpoints to vary not-only in respect of threshold parameter values but also in the assignment of definitions to terms. Although this is clearly an important issue, it will not be considered in the present paper.

3.1. Land Cover Types and their Extensions

As well as by referring to geographic features, the geographic domain is very often described in terms of its terrain or land cover. A region may be wooded, ice covered, rocky etc.. In some cases the boundaries of such regions may be clearly *bona fide*, whereas in others, especially where there is a transitional region between terrain types (e.g. jungle $\leftrightarrow$ scrub-land $\leftrightarrow$ desert), the boundary may be vague. In either case there is certainly a physical basis to land cover demarcations; and in the case where the boundary is vague, the range of reasonable demarcations can be modelled within standpoint semantics in terms of thresholds on appropriate parameters relating to properties of the Earth's surface.

However, apart from such vagueness, there is another characteristic of land cover that is potentially problematic for computational manipulation of geographic data. Land cover types are *downward inherited*, meaning that, if a region is covered by a given type of terrain, then all sub-regions are also covered by this terrain type.[6,7] It is also clear that, if we have a set of regions all covered by the same terrain type, then the mereological sum of these regions is also covered by that type. Both these conditions are entailed by

[6]This kind of inheritance of properties among spatial regions is discussed in detail in [12].

[7]In fact downward inheritance will not normally apply beyond a certain fineness of granularity, but for present purposes we shall ignore this complication and assume that we do not have to worry about fine grained dissections of the world.

the following equivalence, which applies to properties that may be said to be manifest homogeneously over extended regions of space:[8]

TT-hom) $\mathsf{HasTerrainType}(r,t) \leftrightarrow \forall r'[\mathsf{P}(r',r) \rightarrow \mathsf{HasTerrainType}(r',t)]$

With regard to computational manipulation of land cover information this homogeneity property has both positive and negative implications. On the negative side it suggests that if a GIS ontology includes land cover terms that can be predicated of arbitrary regions of geographic space, then the set of regions that can instantiate such predicates, must include arbitrary sub-regions of its base polygons. For example, if the ontology includes a predicate $\mathsf{Water}(r)$, meaning that r is completely covered with water then this will be satisfied by arbitrary dissections (and unions) of those data polygons labelled with the 'water' attribute.[9]

But on the positive side it is clear that one would never want to actually exhibit all water-covered polygons. Once we give the total extent of a given terrain type we can simply exhibit this, and the fact that all its sub-regions also have that type is implicit. It is obvious to a GIS user that an extended region of blue represents water and moreover that every sub-region of the blue area is also wet. (By contrast it is also obvious that, where regions corresponding to countries are indicated on a map, their sub-regions are not themselves countries.) Hence, although a geo-ontology must certainly take account of the downward inheritance of land cover types, it seems that it should be possible to do this without requiring an explicit representation of arbitrary subdivisions of the Earth's surface.

4. Handling Geographic Data: Queries, Definitions and Domains of Quantification

In order to construct an ontology-based GIS capable of answering queries expressed in terms of formally defined geographic concepts and evaluated with respect to geographic data represented by labelled polygons, the following rather challenging problems must be addressed:

P1) The ontology must define all terms in a way that enables their extensions to be somehow computable from the spatial properties and attributes of polygon data.

P2) The formalism must enable the characterisation of features with vague boundaries.

P3) The implementation must be able to deal with regions with implicit geometrical boundaries that are determined by but not explicitly present in the base polygons, without explicitly modelling potentially infinite geometrical dissections of space.

P4) The implementation must be able to take account of the fact that predicates relating to spatially homogeneous properties (such as terrain types) are downwardly inherited (without explicitly modelling arbitrary dissections of space).

P5) An effective method of ontology-based geographic query evaluation must be implemented.

[8] In natural language, such properties are typically associated with mass nouns.

[9] The situation here can be contrasted with the case of a non-downward-inherited feature type predicate such as $\mathsf{Lake}(r)$. In this case, even if we consider geographic space to include arbitrary polygons, only a finite number of these could satisfy this predicate. Hence, it is plausible that instances of $\mathsf{Lake}(r)$ can be obtained by some finitary computation over the base water polygons. Indeed, we have implemented such a computation.

4.1. Spatial Regions and Relations

In order to address **P1**, we need a method of determining the spatial relations that hold between two regions. We use the Region Connection Calculus (RCC) [13], which allows us to express topological relations between regions and to use these to define features involving complex configurations of spatial parts.

However, the standard models of RCC are infinite domains — typically, the sets of all regular closed (or regular open) subsets of Cartesian space (either two or three dimensional). Relating such models to actual data is problematic, because in a computational implementation, one can only refer explicitly to a finite set of entities. Real spatial data usually consists of finite sets of polygons, but the domain of quantification in the standard RCC would include not only these polygons but also all possible ways of carving these up into further polygons.

Our approach to solving this problem is to find a way of working with a finite set of regions, which is adequate to characterise the domain in so far as is relevant to any given spatial query. As discussed in [14], the full set of regions contains many regions we are not interested in, such as tiny regions or obscure shapes with convoluted boundaries, thus we would prefer to work only with the set of regions that we can derive useful or interesting features from. For example, if we are interested only in inland water features, we are only interested in segmenting up the inland water regions, and it may be sufficient to represent the land as a single polygon. We thus choose to restrict our domain of regions to polygons, as previously proposed in [15,16]. To expand upon this, our domain consists of polygons which are initially generated from the data, with further polygons derived from this polygonal information through predicates using standpoints. In [8], we showed how the calculation of the RCC relations between a set of polygons can be performed efficiently.

A problem that arises with such an approach is the generation of this domain. Ideally we would generate all possible polygons to begin with, but this would be too large a set to work with when answering queries. Instead, we start with an initial set of polygons designed to represent the basic separation of *matter types* [17], thus each initial polygon is filled by some specified matter type. These polygons may be further segmented during the query interpretation process.

Such further segmentation will normally arise from shape related or metrical predicates being used in a query (or occurring in the definition of a predicate used in a query). Moreover, since shape and measurement predicates will often be vague, these can correspond to different geometrical conditions, and thus different ways of carving up the initial polygons, according to the standpoint relative to which the query is evaluated.

4.2. Demarcating Vague Regions

Our approach to demarcating vague regions is of course based upon standpoint semantics. This has been explained above and also in several previous papers [8,9] and some further details will be given below in describing our prototype implementation. Here we just give a brief overview. Our procedure first determines a medial axis skeletonisation of the region occupied by a given land cover type. This is then used to segment the region into linear and expansive sub-regions based on threshold values of certain parameters determined by a given standpoint. Vague regions corresponding to different types of geographic feature can then be specified definitionally, in terms of the distribution of land cover types over topological configurations of the regions in this segmentation and over regions derived by further geometrical dissection of these regions.

4.3. Controlled Quantification over Geometrically Derived Regions

We now turn to problem **P3**. One method of constructing an ontology that is computationally tractable over a concrete domain, is to constrain quantification in such a way that all entities (in our case spatial regions) that are relevant to the evaluation of a given formula are either present in an initial finite set of entities, or are members of further finite sets that can be effectively computed from the initial entity set. We now sketch a relatively limited modification of first order logic in which this can be achieved.

Let Base be the finite set of entities (e.g. polygons) present in our data-set. Restricting quantification to range just over entities in Base is clearly tractable, so we can certainly allow quantification of the form:

QB) $(\forall x \in \mathsf{Base})[\phi(x)]$

Many domains have a natural Boolean structure which may be useful for defining properties and relations. Thus in the spatial domain we are often concerned with sums, intersections and complements of regions. Let Base^* be the elements of a Boolean Algebra over Base. We may then allow quantification of the form:

QB*) $(\forall x \in \mathsf{Base}^*)[\phi(x)]$

If Base is finite then so is Base^*. So quantification can still be evaluated by iterating over the domain. But unfortunately Base^* will be exponentially larger than Base, so it would almost certainly be impractical to do this in a real application. However, there is another way of extending the domain of quantification, which is both more controllable and more flexible.

Let $\Gamma(t_1, \ldots, t_m;\, x_1, \ldots, x_n)$ be a relation, such that given any m-tuple of ground terms $\langle t_1, \ldots, t_m\rangle$, one can effectively compute the set of all n-tuples $\langle x_1, \ldots, x_n\rangle$, such that $\Gamma(t_1, \ldots, t_m;\, x_1, \ldots, x_n)$ holds. We may call $\langle t_1, \ldots, t_m\rangle$ an input tuple and $\langle x_1, \ldots, x_n\rangle$ an output tuple. The condition on Γ means that for any given finite set of input tuples there is a finite set of output tuples such that some pair of input and output tuples satisfies Γ. For example, Γ might be a spatial relation $\mathsf{BisectNS}(r; r_1, r_2)$ which is true when r_1 and r_2 are the two parts of r obtained by splitting it into northern and southern parts across the mid-line of its extent in the north-south dimension. Another example is $\mathsf{Concavity}(r, r')$, where given an input polygon r there are a finite number of polygons r' corresponding to concavities of r (i.e. maximal connected regions that are parts of the convex hull of r but do not overlap r).

We shall call relations of this kind *effective generator relations*. They are simply logical representations of a certain kind of algorithm that could be implemented in computer software — and indeed much of the functionality of a GIS depends on algorithms of this kind. Given an effective generator relation Γ, we can now define the following form of controlled quantification:

QEGR) $(\forall x_1, \ldots, x_n : \Gamma(t_1, \ldots, t_m;\, x_1, \ldots, x_n))[\, \phi(x_1, \ldots, x_n)\,]$

Here, the variables $t_1, \ldots, t_m$ must be already bound to wider scope quantifiers, which can be either quantifications over Base or over domains specified by other effective generators. Hence, the range of each variable is restricted either to Base or to a set of entities that can be computed from Base by applying algorithms corresponding to a series of effective generator relations.

Semantically, **QEGR** is interpreted as equivalent to:

- $(\forall x_1, \ldots, x_n)[\,\Gamma(t_1, \ldots, t_m;\, x_1, \ldots, x_n) \rightarrow \phi(x_1, \ldots, x_n)\,]$

4.4. Spatially Homogeneous Properties and Downward Inheritance

So far we have not implemented any mechanism for handling downward inheritance. Instead we have circumvented the issue by limiting our predicates to those satisfied either by maximal components of uniform land cover, or by regions derived from these by particular geometrical decompositions. For instance, we define 'linear stretches' of water which are geometrically dissected (relative to a given standpoint) from the total region of water. In the future we would like to handle spatially homogeneous properties by representing their logical relationship to base polygons.

4.5. Query Evaluation

We express a query by means of the notations $\mathsf{?} : \phi$ representing a test as to whether ϕ is true in relation to a given data-set and $\mathsf{?}(x) : \phi(x)$, which means: return a list of all entities e_i such that $\phi(e_i)$ is true as determined by interpreting the symbols of ϕ in relation to the data-set. More generally, $\mathsf{?}(x_1, \ldots, x_n) : \phi(x_1, \ldots, x_n)$ would return a list of n-tuples of entities satisfying the given predicate. In our context, the entities returned will normally be polygons. Query variables cannot occur within any of the quantifiers of our representation, however they can be identified with values of these variables by the use of an equality predicate.

Since queries will be interpreted in relation to actual geographic data, it is natural to use a *model-based* approach to query evaluation.[10] General purpose model building systems, such as MACE [19], allow consistency checking of arbitrary first order formulae, by checking all possible assignments to predicates. But in our case we have a single interpretation of basic predicates that can be derived directly from the geographic dataset. Thus, we can compute sets of all tuples satisfying the predicates that occur in a query and then evaluate the query formula over this model.

Boolean connectives can be evaluated in an obvious way, but the treatment of quantifiers is somewhat more complex. Since quantification is restricted to range over either base polygons or derived polygons generated by the **QEGR** quantifiers, this means that the domain of regions that must be considered is finite. By examining the structure of nested **QEGR** quantifiers occurring in a query, we can determine sequences of spatial function applications which, when applied to the base polygons, will generate all polygons that are relevant to that query. Once these polygons have been computed, quantifiers can be evaluated over this extended domain. Our current prototype does not explicitly include the **QEGR** quantification syntax, but implements a simplified version of this

[10]Model-based reasoning has been applied in various areas of AI. For instance, a similar approach to ours has been used in Natural Language Processing [18].

mechanism. It is geared towards evaluating queries containing a limited range of predicates and generates domains of polygons that are sufficient to deal with these. This will be described in the next section.

5. Implementation within a Prototype GIS

We now give some details of our GIS prototype which we call GEOLOG. The system is implemented in Prolog and operates on several hydrographic data-sets covering estuarine river systems in the UK. The system implements geometric shape decomposition algorithms based on a number of parameters. These are linked to an explicit representation of shape predicates using a first order formalisation in which the parameters attached to predicates are interpreted according to standpoint semantics. First order queries can be evaluated and their instantiations depicted on a cartographic display.

5.1. Shaped-Based Properties and Segmentation

Since queries may contain RCC relations describing topological relations between regions, a database of RCC relations over all stored polygons is maintained. This requires a considerable amount of storage but means that these relations do not have to be recomputed whenever a new query is executed, which greatly speeds up query answering times. As described in [20,8] segmentation of regions into linear and expansive parts is computed using a *medial-axis* approach which is supported by use of the VRONI software package [21]. The idea is to measure width variation along the medial axis. Given a medial axis point p of region r which is distance d from the edge of r, we compute the maximum and minimum distances, max, min, to the edge of r of all medial axis points within distance d of p. The value l=$\mathsf{max}/\mathsf{min}$ gives a useful measure of the width variation at p. $l = 1$ means the width is constant, and a value of 1.2, for example, means that there is a 20% width variation in a section of the medial axis centred at p along a length equal to the width at p. Using this value as a standpoint parameter, the predicate $\mathsf{Stretch}(r)$, corresponding to the vague concept of a ‘linear stretch of water’ is defined. This is a maximal connected water region all of whose medial axis points have a value of l less than a given threshold.

5.2. Query Evaluation

In developing an effective implementaton, we wanted to minimise both the number of polygons stored in the system and the time required to construct polygons by geometrical computation. This led us to an approach of ‘just in time’, incremental expansion of the domain. The basic idea is that that when presented with a query, GEOLOG ensures that all polygons relevant to its interpretation are generated before evaluating the query as a whole. But it then stores the generated polygons as they are likely to be required again for subsequent queries.

The initial dataset consists simply of a partition of the geographic space comprising polygons labelled with the basic land cover types: *land*, *sea* and (fresh) *water*. Queries relating to the base polygons themselves can be answered straightforwardly, although they are of little interest as they do not take any account of the semantics of geographic features. However, a number of higher level geometric and hydrographic predicates are also available for use in queries. Each of these predicates is associated with an algorithm for expanding the domain of polygons by geometrical computations, to include additional polygons corresponding to all their possible instances. When a query containing one or more of these non-basic predicates is entered, the domain is first expanded according to the associated algorithms (in general this must be done recursively until a fixed point is reached), and the newly generated polygons are labelled with appropriate attributes. Once this procedure has has been carried out, the dataset contains polygons corresponding to all possible instances of predicates occurring in the query. Quantifiers can now be evaluated by iterating over polygons in this expanded dataset.

For instance, if one enters a query $\mathsf{Stretch}(x)$ GEOLOG would perform a linearity segmentation relative to a given standpoint, so that the required linear and expansive polygons are generated. We can now answer queries involving reference to stretches or to any concepts that have been defined in terms of linear and/or expansive polygons. A user of the system has direct access to the threshold assignment defining the standpoint and can modify the thresholds. When this is done the system must recompute the segmentation, and this in turn will lead to different polygons being returned from queries that depend on the segmentation.

5.3. Results of Querying for Stretches and Rivers

Results of executing the query $\mathsf{Stretch}(x)$ with different input datasets and linearity parameter thresholds are shown in Figure 2. The images on the left correspond to a threshold of 1.2, whereas those on the right are for a threshold of 1.4. Thus, the interpretation on the right takes a more liberal view of what can count as linear than the one on the left.

As is clear from Figure 2, the artificial concept of ‘linear water stretch’ does not correspond directly to the natural concept of ‘river’. Typically a river will consist of many such stretches, interspersed with more expansive areas of water, corresponding to bulges in the watercourse. We experimented with a range of threshold parameters governing how loosely or strictly the predicate ‘linear’ is interpreted; but found that there is no threshold that yields a natural interpretation of ‘river’. If we use a loose definition that allows bulges to be classified as parts of a stretch, we find that very expansive, lake-like water regions become incorporated into stretches. But if we tighten the linearity threshold to rule out obvious lakes, then rivers must be consist of fragmented sequences of stretches.

In order to circumvent this problem, we propose that a river should indeed be modelled as a sequence of stretches interspersed by bulges. To make this precise we have introduced a further artificial concept of *interstretch*. This is a water region that is expansive but such that all its parts are ‘close’ to a water stretch, where closeness is defined by a second threshold applied to a suitable geometric measure. This enables us to incorporate small bulges into rivers without needing to unduly weaken our general criteria for identifying linear water features. As described in [8], this has been found to interpret the concept river in a very plausible way.[11]

The introduction of interstretches might at first sight appear to be an *ad hoc* hack. However, we believe that a plausible general explanation can be given as to why this seems to work. In classifying a vague feature, we suggest that one is looking for criteria

[11] Further complications arise from the branching structure of water systems. These have been only partially solved and are a topic of ongoing work.

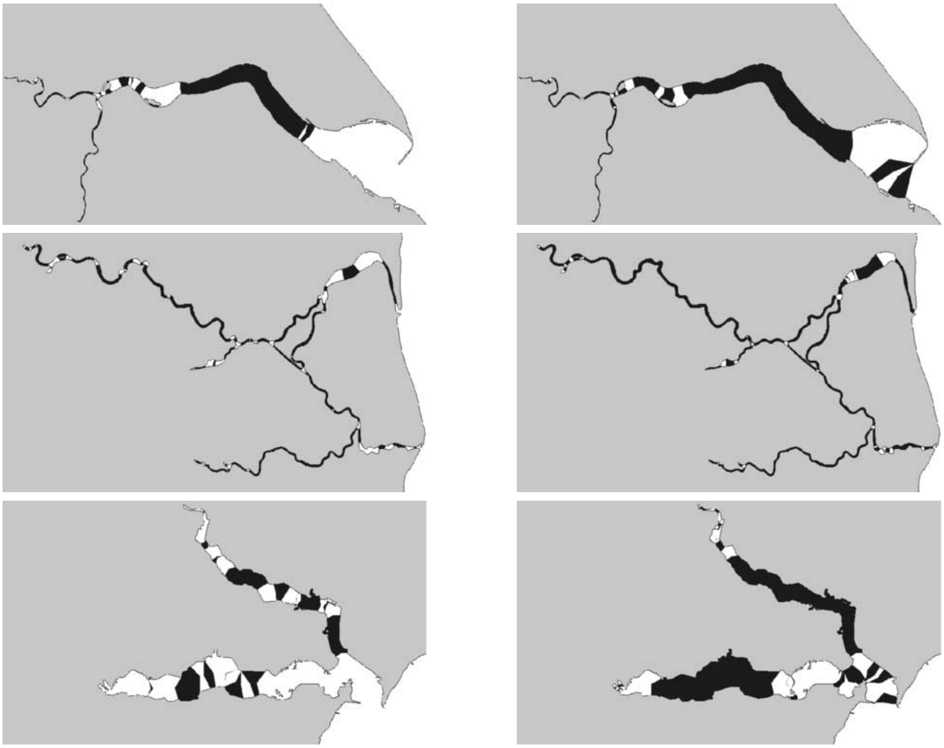


Figure 2. A comparison of marking 'linear water stretches' relative to different The top images are of the Humber Estuary, the middle images are of the Norfolk Broads at Great Yarmouth and Lowestoft. The bottom images are of the Stour And Orwell Estuary.

that are satisfied globally by a region but is also prepared to allow exceptions in regard to small parts of the region that deviate from these criteria. For instance, to classify a surface as approximately planar, one is looking for a global approximation to a plane but will accept small areas where the surface departs considerably from planarity, which are regarded as insignificant bumps on the surface. We thus plan to apply a similar approach to classifying other types of geographic feature.

6. Conclusion

We have described a variety of ontological issues that complicate the issue of defining and individuating geographic regions and features. From theoretical analysis of the semantics of vagueness and of computational manipulation of geometric decompositions of polygonal data, a possible architecture for implementing an ontology-based GIS is taking shape. Our current prototype gives a strong indication that this can lead to a new kind of GIS in which geographic terminology is grounded upon data *via* rigorous definitions rather than *ad hoc* segmentations. However, much work remains to be done, both in terms of specifying a more extensive geographic ontology and also in relation to developing a more flexible and efficient query answering mechanism.

Acknowledgements

We are grateful to the EPSRC for supporting this work *via* grant EP/D002834 and to the Ordnance Survey, for their contribution towards funding a PhD on this topic. Valuable comments from the reviewers are also gratefully acknowledged.

References

[1] Achille C. Varzi. Vagueness in gegraphy. *Philosophy and Geography*, 4:49–65, 2001.

[2] Brandon Bennett. What is a forest? on the vagueness of certain geographic concepts. *Topoi*, 20(2):189–201, 2001.

[3] Barry Smith and David M. Mark. Do mountains exist? towards an ontology of landforms. *Environment and Planning B: Planning and Design*, 30(3):411–427, 2003.

[4] Paulo Santos, Brandon Bennett, and Georgios Sakellariou. Supervaluation semantics for an inland water feature ontology. In L.P. Kaelbling and A. Saffiotti, editors, *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI-05)*, pages 564–569, Edinburgh, 2005.

[5] Kit Fine. Vagueness, truth and logic. *Synthèse*, 30:263–300, 1975.

[6] Timothy Williamson. *Vagueness*. The problems of philosophy. Routledge, London, 1994.

[7] Brandon Bennett. A theory of vague adjectives grounded in relevant observables. In Patrick Doherty, John Mylopoulos, and Christopher A. Welty, editors, *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning*, pages 36–45. AAAI Press, 2006.

[8] David Mallenby and Brandon Bennett. Applying spatial reasoning to topographical data with a grounded ontology. In F. Fonseca, M. Andrea Rodrígues, and Sergei Levashkin, editors, *GeoSpatial Semantics, proceedings of the second international conference*, number 4853 in Lecture Notes in Computer Science, pages 210–227, Mexico City, November 2007. Springer.

[9] Allan Third, Brandon Bennett, and David Mallenby. Architecture for a grounded ontology of geographic information. In F. Fonseca, M. Andrea Rodrígues, and Sergei Levashkin, editors, *GeoSpatial Semantics, proceedings of the second international conference*, number 4853 in Lecture Notes in Computer Science, pages 36–50, Mexico City, November 2007. Springer.

[10] Barry Smith. On drawing lines on a map. In Andrew Frank and Werner Kuhn, editors, *Spatial Information Theory — Proceedings of COSIT'95*, number 988 in Lecture Notes in Computer Science, pages 475–484, Berlin, 1995. Springer.

[11] Barry Smith. Fiat objects. *Topoi*, 20(2):131–148, 2001.

[12] Alberto Belussi and Matteo Cristani. Mereological inheritance. *Spatial Cognition and Computation*, 2:467–494, 2000.

[13] D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic-based on regions and connection. In *Principles Of Knowledge Representation And Reasoning: Proceedings Of The Third International Conference (Kr 92)*, pages 165–176. Morgan Kaufmann Pub Inc, San Mateo, 1992.

[14] Ian Pratt and Dominik Schoop. A complete axiom system for polygonal mereotopology of the real plane. *Journal of Philosophical Logic*, 27(6):621–661, 1998.

[15] Volker Haarslev, Carsten Lutz, and Ralf Möller. Foundations of spatioterminological reasoning with description logics. In *Principles of Knowledge Representation and Reasoning*, pages 112–123, 1998.

[16] Rolf Grütter and Bettina Bauer-Messmer. Towards spatial reasoning in the semantic web: A hybrid knowledge representation system architecture. In S. Fabrikant and M. Wachowicz, editors, *The European Information Society: Leading the Way With Geo-information*, volume XVII of *Lecture Notes in Geoinformation and Cartography*, page 486, 2007.

[17] Brandon Bennett. Space, time, matter and things. In *FOIS '01: Proceedings of the international conference on Formal Ontology in Information Systems*, pages 105–116, New York, NY, USA, 2001. ACM.

[18] Patrick Blackburn and Johan Bos. *Representation and Inference for Natural Language. A First Course in Computational Semantics*. CSLI, 2005.

[19] William McCune. Mace4 reference manual and guide. *CoRR*, cs.SC/0310055, 2003.

[20] David Mallenby. Grounding a geographic ontology on geographic data. In E. Amir, V. Lifschitz, and R. Miller, editors, *Proceedings of the 8th International Symposium on Logical Formalizations of Commonsense Reasoning (Commonsense-07)*. AAAI, 2007.

[21] Martin Held. VRONI: An engineering approach to the reliable and efficient computation of voronoi diagrams of points and line segments. *Computational Geometry: Theory and Applications*, 18(2):95–123, 2001.

This page intentionally left blank

8. Formal Concept Analysis

This page intentionally left blank

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-297

Utility Ontology Development with Formal Concept Analysis

Gaihua Fu and Anthony G Cohn
School of Computing, University of Leeds, UK
{gaihua, agc}@comp.leeds.ac.uk

Abstract. Although it is well recognised that ontologies have an important role to play in data integration, the lack of established ontologies in domains of interest often makes ontology-based integration a difficult task. Previous research on ontology design methodologies shows that manual construction of ontologies is a complex process and it is very hard for a designer to develop a consistent ontology. This paper contributes a formal and semi-automated approach for the development of ontologies in the utility infrastructure domain. It arises from a practical industrial problem of integrating the vast network of underground asset records. These asset records are typically autonomous, i.e. owned and maintained by individual organisations, and are encoded in an uncoordinated way, i.e. without consideration of interoperability with other utility information systems. The proposed approach is based on formal concept analysis (FCA) which is a mathematical approach for abstracting from attribute-based object descriptions. This paper describes techniques developed to support utility ontology development, with a focus on resolving implicit and mismatch data. Some experiments have been carried out to construct a utility ontology with data from utility companies. Though issues addressed in the paper arise in utility ontology development, we anticipate that they should be interesting and relevant to other application domains.

Keywords: Ontology, Integration, Formal Concept Analysis

1. INTRODUCTION

In many domains, one faces the need for exchanging and sharing information that comes from different resources. Obtaining mapping information of the excavation site in street works is one such example. Every year, in excess of four million holes are dug in UK roads to maintain utility assets [2]. In order to avoid unnecessary holes dug in wrong places, it is required that information of buried utilities must be obtained before excavation occurs. However, the mapping information supplied by utilities is often of limited use. One main reason for this is that asset records are usually created and maintained by a range of private companies with little thought towards interoperability. As a result, utility data differs from one company to another not only in *what* is encoded but also *how* it is encoded. This data heterogeneity makes it extremely difficult for the excavators to synthesize an integrated view of the excavation site.

Overcoming data heterogeneity has been an active area of research in database and information integration communities [3, 14]. Ontology research is another discipline that deals with data heterogeneity [18, 12]. The common definition of the term ontology is that an ontology is some formal, explicit specification of a domain of discourse. Ontology-based integration systems are usually characterised by a global ontology which represents a reconciled, integrated view of the underlying data sources. Systems taking this approach usually provide users with a uniform interface – all queries made to source data are expressed in terms of the global ontology, thus freeing them from the need to understand each individual data source. This approach looks straightforward but a shared ontology is required. Unfortunately, in many domains one faces the problems of having no accepted ontologies that can be employed in the integration work. The utility domain is one example of this. According to our investigations, little work has been attempted previously to develop utility ontologies. The most relevant one is the standard proposed by the FGDC [4]. However, our research [7] show that the knowledge encoded in FGDC standard is insufficient to serve as a reference ontology on which to base UK utility data integration.

To attempt to remedy this, we investigate techniques that support ontology development for utility domain. In [7], we reported on how a basic ontology was developed manually for the water utility domain. Our experiences show that manual construction ontology is a time-consuming process and it is very hard for a designer to develop a consistent ontology. In this paper we present an alternative approach for utility ontology development. The method is based on formal concept analysis (FCA). By deriving conceptual structures based on attribute descriptions of utility asset features, the method supports the development of utility ontologies in a systematic and semi-automated manner. We discuss interesting issues when applying FCA in ontology development, particularly the treatment of implicit and ambiguous information. We report on our experiments on employing the proposed techniques to construct a sewer utility ontology.

The remaining part of the paper is organized as follows. Section 2 recalls key notions of FCA and reviews related research. Section 3 presents our method for utility ontology development based on FCA. Sections 4 and 5 discuss techniques that deal with implicit and ambiguous information. Section 6 reports our experimental results. Section 6 concludes the paper and points out future research.

2. Basic Concepts and Related Research

2.1. Formal Concept Analysis (FCA)

FCA was developed in the 1980s [26]. A typical task that FCA can perform is data analysis, making the conceptual structure of the data visible and accessible.

A basic notion for FCA is a formal context, which is defined as a triple $K := <G, M, I>$, where G is a set of objects, M is a set of attributes, and $I \subseteq G \times M$ is a binary relation between G and M. A relation $(g, m) \in I$ is read as "object g has the attribute M". A formal context can be depicted by a cross table as shown on the left side of Figure 1, where the elements on the left side are objects; the elements at the top are attributes; and the relation between them is represented by the crosses.

A formal concept of a context $K := <G, M, I>$ is defined as pair (A, B), where $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. A' is the set of attributes common to all the objects in A and B' is the set of objects having the attributes in B. The *extent* of the concept (A, B) is A and its *intent* is B. The formal concepts of a context are ordered by the *sub*- and *super*-concept relation. The set of all formal concepts ordered by such relations forms a concept lattice. For example, the right side of Figure 1 shows the classic concept lattice corresponds to the context on the left side of Figure 1.

	A1	A2	A3
O1	X		X
O2	X	X	
O3		X	X

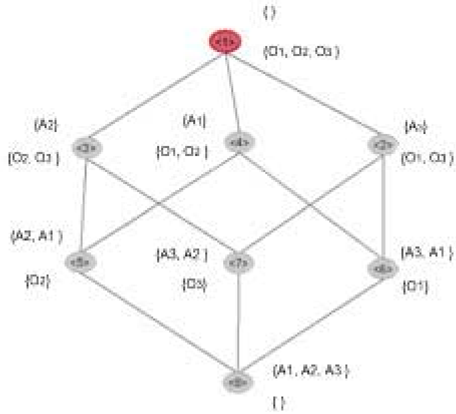


Figure 1 A Formal Context and its Corresponding Concept Lattice

The formal contexts we introduced above are not the ones that occur most frequently in applications of FCA. Most often data is encoded in form of *many valued contexts*. A many valued context K :=<G, M, W, I> consists of a set of objects G, a set of attributes M, a set of attribute values W, and a ternary relation I $\subseteq$ G × M × W. A relation (g, m, w) $\in$ I is read as "object g has the attribute M and its value is w". A many valued context can be unfolded into a one valued context through conceptual scaling (the reader is referred to [6] for discussion on conceptual scaling).

2.2. Related Research

A growing number of methods have been proposed in recent years to address the issues of ontology development [12, 10]. Most methods are based on the traditional knowledge engineering approach [5, 16, 22, 23]. These methods usually start with defining the domain and scope of ontologies. This is followed by a data acquisition process: important classes are collected; a class hierarchy is derived; and properties and semantic constraints are defined and attached to classes. The next step is the specification of ontologies in some formalisms. Finally, ontology evaluation is performed against pre-defined criteria. It is recognised that developing ontologies from scratch with the traditional knowledge engineering approach is time consuming and labour intensive. Reusing and integrating existing ontologies is thus considered as part of ontology development process by most methods described above. However few of these methods address integration in detail. In [20], a method that is based on the integration is proposed. Other systems and frameworks that have been developed for supporting ontology merging or integration tasks include those described in [8, 17].

Fundamental operations for ontology integration are mapping discovery and ontology merging. Mapping discovery takes two or more ontologies as input and produces a mapping between elements that semantically correspond to each other. Most approaches rely on matching heuristics for identifying mappings, comparing names and the natural language definitions of two concepts and checking the closeness of two concepts in the concept hierarchy [15, 13]. Based on the inter-ontology mappings derived in mapping discovery, a merging process integrates the source ontologies and generates a concept hierarchy that embeds the knowledge encoded in the initial ones. However, as recognised in [25], deriving a meaningful concept hierarchy is a hard problem even with the ground set of inter-ontology mappings provided. Most methods that support the merging process are performed in an interactive manner with the interference of human users. In cases of large scale ontologies, the resulting class hierarchies generated tend to be inconsistent.

Another branch of research studies ontology construction with formal methods. Of particular interest here is research on FCA. FCA is a formal method for conceptual structure derivation. FCA related tools enable considerable of knowledge processing activities automated, particularly concept generation and hierarchy derivation. As a result FCA has been attracting great interest to support systematic, semi-automated development of ontologies. For example, FCA has been employed to construct domain specific ontologies in research [11, 19]. The theory of FCA has also been adapted to problems of ontology or concept hierarchy merging [25, 19, 21].

Building on advances made in [25, 21, 19], in this research we propose a FCA-based method for utility ontology construction, with the focus on treating implicit and ambiguous information. Previous work publication is either implicit on how these problems are resolved, or only addresses particular types of these problems. For example, in [25] there is an interesting discussion on attribute naming conflicts, but the authors do not address in detail how these problems are resolved.

3. FCA-based Utility Ontology Integration

With FCA theory as the backbone, we have developed a method that is designed to support utility ontology integration. The method supports automated concept hierarchy derivation and mapping identification, and it also supports ontology integration in the presence of implicit and ambiguous information. Implicit information is caused by the fact that utility companies tend to explicitly encode asset types having specific attributes, but leave others unspecified. As an example in the sewer domain, a sewer pipe is characterised by how it conveys sewerage: either by *gravity* or by *pressure*, with the gravity distribution employed more often than pressurised one. Most utility companies explicitly specify pressurised sewer pipes, but not gravity sewer pipes. If not restored in FCA, this missing information can lead to an ontology that is ill-formed, and does not correctly capture critical concepts and semantics of the domain.

Another challenge in applying FCA in ontology integration is to deal with inconsistent or ambiguous information, particularly inconsistencies/ambiguities existing with asset attributes. For example, different terms may be employed to refer to the same attribute, and attributes may be modelled at different level of detail. A simple example is that one utility data resource may model a sewer pipe as either *main* or *lateral* and another may classify it as *trunk main*, *non-truck main*, and *servic*e. These mismatches pose considerable difficulties in applying FCA to ontology integration.

In this section, we present a generic framework that supports ontology integration with FCA. We will describe in detail the techniques that deal with implicit and ambiguous data in Section 4 and 5. Figure 2 shows the main components of the FCA-based integration framework. The process of utility ontology development consists of three steps: Context Formation, Context Composition, and Ontology and Mapping

Derivation. Context Generation takes data and meta-data from a utility dataset as input, and generates a many valued context K := <G, M, W, I>, where G contains asset types, M contains attributes of asset types, W contains attribute values, and a ternary relation $I \subseteq G \times M \times W$ contains object attribute value relationships. The next step, Information Explication, restores implicit information, which will be covered in more detail in Section 4. The final step, Conceptual Scaling, transforms many valued contexts into one valued contexts, in order for classic FCA techniques to be applicable.

Context Composition takes two contexts as input and generates an integrated concept lattice. A data dictionary that maintains utility asset and related terms is employed to disambiguate conflict attributes in context composition, which will be detailed in Section 5. The integrated context is then fed to a lattice generating component to produce a concept lattice. To prevent too many irrelevant concepts (with respect to the concepts in source contexts) from being generated, a pruned concept lattice is derived here instead of classic concept lattice of the context. A pruned concept lattice is a lattice that eliminates the concepts that have empty *intent* and *extent* [9]. The computation of the concept lattice is done with the FCA tool Galicia [24].

Ontology and Mapping Derivation takes the concept lattice generated in the previous step as input and generates an integrated ontology as well as mappings between utility asset types. Each formal concept of the lattice is a candidate for an ontology class or relation (human interaction is required in this step for decision making). The edge between two concepts indicates an *is-a* relationship. The derived ontology is represented in OWL to form a formal ontology specification. This step also generates candidate mappings between asset types. Given a formal concept, if its extent contains more than one object (asset type), then it indicates a potential mapping between these asset types.

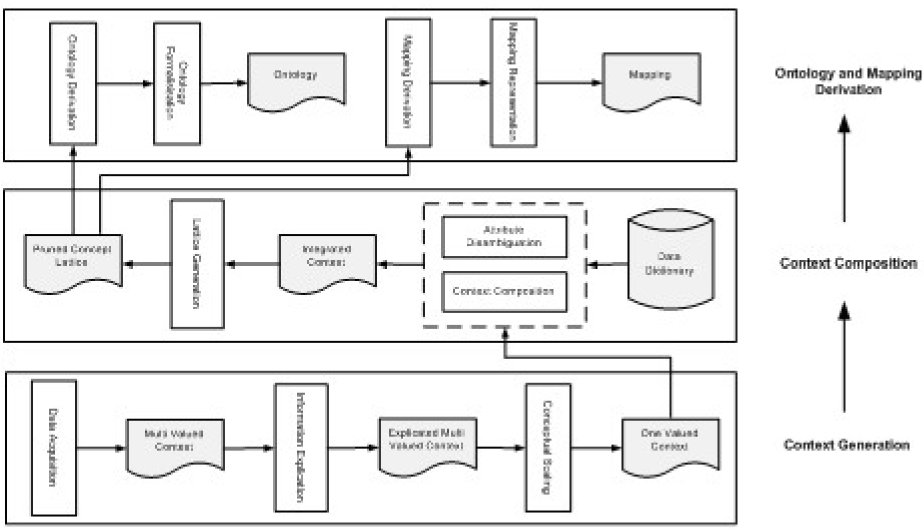


Figure 2 FCA-based Utility Ontology Integration

4. Information Explication

The data acquisition step (shown in Figure 2) acquires asset types from a utility dataset and results in a many valued context. Table 1 shows a portion of a many valued context we generated from a utility dataset. It explicitly specifies attributes such as *pressurized, above_ground* and *abandoned,* leaving those such as *gravity, underground* and *operational* implicit. Figure 3 shows the corresponding concept lattice.

Table 1

	size	what	how	position	status
sewerPipeType1	main		pressurised		
sewerPipeType2	main			above_ground	
sewerPipeType3	main	sludge			
sewerPipeType4	main				abandoned

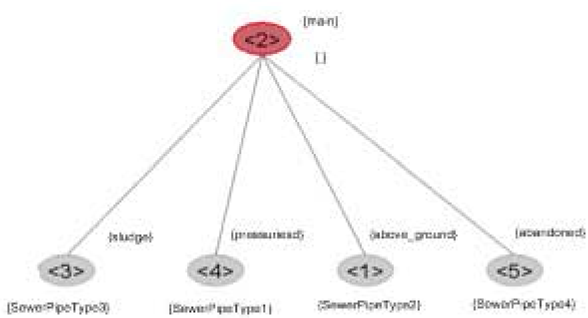


Figure 3 Concept Lattice Before Restoring Implicit Information

To restore implicit information, we use a set of domain specific rules[1], e.g., the rules to restore implicit information for the above context table are shown in Figure 4.

> *1) if a sewer is not explicitly specified as a pressurised sewer, then it is a gravity sewer;*
> *2) if a sewer is not explicitly specified as an above ground sewer, then it is an underground sewer;*
> *3) if a sewer is not explicitly specified as an abandoned sewer, then it is an operational sewer;*
> *4) a sludge sewer is a pressurised sewer unless it is specified otherwise;*

Figure 4 A Set of Rules for Restoring Implicit Information.

Several methods have been suggested in this research to apply the rules to the contexts, and each results in a concept lattice with different structure and granularity.

[1] Rules are collected based on domain knowledge and with the assistance of utility experts. The detail on the rule selections is not described here due to space limitation.

I. This method applies all rules to each object, which results in a context with same number of objects and objects are updated with new attributes. For example, *sewerPipeType1* will have following new attribute: *wastewater, underground and operational*. Table 2 shows the updated context with this method.

Table 2

	size	**what**	**how**	**position**	**status**
sewerPipeType1	main	wastewater	Pressurised	underground	operational
sewerPipeType2	main	wastewater	Gravity	above_ground	operational
sewerPipeType3	main	sludge	Pressurised	underground	operational
sewerPipeType4	main	wastewater	Gravity	underground	abandoned

II. This method retains original objects, and extends the context table with new objects generated by applying different combination of rules. For example, by using the first two rules shown in Figure 3, three new objects can be derived from the object *sewerPipeType1*, which is shown in Table 3. This option is potentially useful for generating a lattice with the richest semantics, but the number of objects derived is exponential in the number of rules. The number of concepts is even bigger, which grows exponentially with the numbers of objects.

Table 3

	size	**what**	**How**	**Position**	**status**
sewerPipeType1	main		Pressurised		
object1	main	wastewater	Pressurised		
object2	main		pressurised	underground	
object3	main	Wastewater	pressurised	underground	
::: :::					

III. The third approach, which is a balance between the first and the second approaches, retains original objects, but for each object, generates a new one by applying all rules. The context generated with this approach is shown in Table 4.

Table 4

	size	**what**	**how**	**Position**	**Status**
sewerPipeType1	main		pressurised		
object1	main	wastewater	pressurised	Underground	operational
sewerPipeType2	main			above_ground	
object2	main	wastewater	gravity	above_ground	operational
sewerPipeType3	main	sludge			
object3	main	sludge	pressurised	underground	operational
sewerPipeType4	main				abandoned
object4	main	wastewater	gravity	underground	abandoned

Figure 5 shows the concept lattice which has implicit information restored with approach I. Comparing with the original concept lattice shown in Figure 3, extra formal concepts such as *({underground}, {})* and *({gravity}, {})*, are identified here, which capture important semantics of the application domain and will form critical ontological concepts. We are still investigating which approach discussed above best recovers these missing semantics, which is largely performed by doing ontology coverage analysis and some evaluation with domain users.

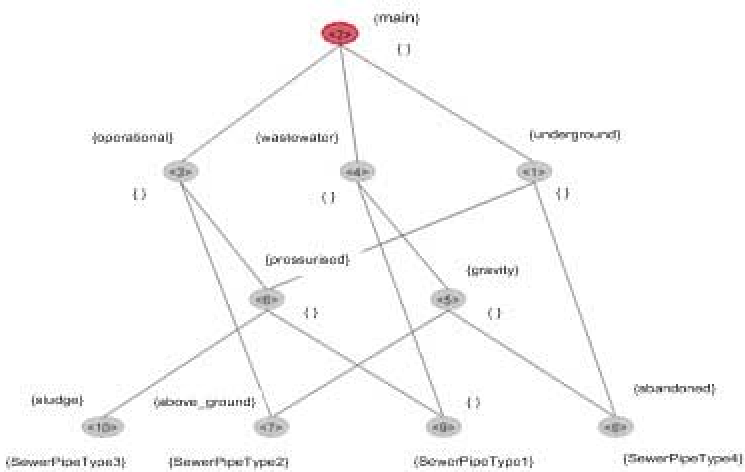


Figure 5 Concept Lattice After Restoring Implicit Information

5. Attribute Disambiguation

Context composition takes two formal contexts[2] as input, and generates an integrated lattice. Due to the possibility of inconsistencies arising from attributes of the source contexts, context composition is not a simple union or disjoint union operation. To deal with this, we employed a utility data dictionary. The data dictionary is developed with techniques described in [7], and it maintains a set of terms that describes utility asset types, and it also encodes terminological relationships between utility terms, such as BT/NT (Broader Term/Narrower Term), USE/UF (Use/Used For) and RT (Related Term). With the assistance of the data dictionary, we are not only able to identify whether two attributes are equivalent (by utilising USE/UF relationships), but also able to generate subsumption relationships between two attributes (by utilising BT/NT relationships). In what follows, we describe the operations that resolve attribute mismatch, and we will use the context tables shown in Figure 6 to illustrate the context composition process.

Context K1

	operational	abandoned	proposed recommission
ob1	X		
ob2			X
ob3		X	

Context K2

	live	abandoned intact	proposed	standby
ob1		X		
ob2	X			
ob3			X	
ob4				X

Figure 6 Example Source Contexts

[2] Which are one valued after the conceptual scaling step.

Given two contexts K1: =<G1, M1, I1> and K2:=<G2, M2, I2>, the integrated context K: =<G, M, I> is computed by first performing a disjoint union of object sets of two contexts, i.e., $G = G_1 \cup^{*} G_2$. For an object (g, n) ∈ G of K, we say that (g, n) *originates* from K1 when n=1 and from K2 when n=2. M and I are assigned M1 and I1 from K1 at this stage. That is $M \equiv M_1$ and $I \equiv I_1$. Table 5 shows the integrated context K after the above operations.

Table 5

	operational	abandoned	proposed recommission
(ob1, 1)	X		
(ob2, 1)			X
(ob3, 1)		X	
(ob1,2)			
(ob2,2)			
(ob3,2)			
(ob4,2)			

The next step expands context K with respect to attributes of K2. For each attribute Ai ∈ M2 of K2, we perform a semantic mapping operation with attributes in K. Depending on the type of match resulted, different operations are performed:[3]

I. Ai finds an equivalent attribute Aj ∈ M of K. Such an equivalent attribute may either be specified with the same term, or with a synonym. For example for the context K2 in Figure 5, if the attribute *live* finds an equivalent attribute *operational* in K, Ai will be unified with Aj, and context table K is expanded with existing relationships between Ai (or Aj after unification) and objects that originated from K2, as shown in Table 6. This context expansion is highlighted with columns and rows having a emboldened border in the table.

Table 6

	operational	abandoned	proposed recommission
(ob1, 1)	X		
(ob2, 1)			X
(ob3, 1)		X	
(ob1,2)			
(ob2,2)	**X**		
(ob3,2)			
(ob4,2)			

[3] In many situations, complications can occur, e.g.. an attribute may have multiple matches. The primitive operations described can be composed to deal with these complex cases. We will not elaborate this further here due to space limitations.

II. Ai finds attribute Aj that is more generic to it. For example, the closest match for *abandoned intact* in K2 is *abandoned* which is a broader term to it. In this case, the resulting context K is expanded with attribute Ai and existing relationships between Ai and objects from K2. New binary relationships (as shown with shaded cells) are established in K between those objects having attribute Ai from K2 and attributes Aj (which is originally from K1). This is shown in Table 7. The underlying theory is that if Ai is a sub term of Aj, then any object which has attribute Ai should also have attribute Aj. In our example, if a sewer pipeline is *abandoned intact*, then it is also *abandoned*.

Table 7

	operational	abandoned	proposed recommission	abandoned intact
(ob1, 1)	X			
(ob2, 1)			X	
(ob3, 1)		X		
(ob1,2)		**X**		**X**
(ob2,2)	X			
(ob3,2)				
(ob4,2)				

III. Ai finds a matching attribute Aj that is more specific to it. For example, the closest match for the attribute *proposed* in K2 is *proposed recommission* in K which is a narrower term for it. In this case, the context K is expanded with Ai and existing relationships between Ai and objects originating from K2, as shown in Table 8. New binary relationships (as shown with shaded cells) are established in K between those objects having attribute Aj (originally from K1) and attribute Ai (which is originally from K2).

Table 8

	operational	abandoned	proposed recommission	abandoned intact	proposed
(ob1, 1)	X				
(ob2, 1)			X		**X**
(ob3, 1)		X			
(ob1,2)		X		X	
(ob2,2)	X				
(ob3,2)					**X**
(ob4,2)					

IV. Ai finds no match in K. For example there is no semantic match in K for the attribute *standby* in K2. In this case the context K is simply expanded with Ai and existing relationships between Ai and objects originating from K2, as shown in Table 9.

Table 9

	operational	abandoned	proposed recommission	abandoned intact	proposed	standby
(ob1, 1)	X					
(ob2, 1)			X		X	
(ob3, 1)		X				
(ob1,2)		X		X		
(ob2,2)	X					
(ob3,2)					X	
(ob4,2)						**X**

6. Experiments

The techniques described in the previous sections have applied to real world utility data to build an ontology for the sewer utility domain. Data that specifies sewer pipelines were collected from two utility companies. From utility dataset one, 54 utility asset types were collected; these asset types are described with 21 attributes. From utility data resource two, 49 utility asset types were collected; these asset types are described with 16 attributes. A set of rules for restoring implicit information was collected for each data resource. Our preliminary experiments demonstrated that these rules overlap considerably between two datasets, which leads to the assumption that these rules are largely domain specific rather than application specific. That is, rules collected are applicable to utility data from different utility companies and therefore can be reused in future integration. However we anticipate that due to the different encoding practices employed by utility companies, the numbers of applicable rules will differ from one dataset to another. Further experiments are still required to prove these assumptions. The approach I (discussed in Section 4) was tested for restoring implicit information, and Figure 7 shows the concept lattices for two datasets with implicit information restored (labels are eliminated here for the reasons of confidentiality and readability). Work is still ongoing in testing approaches II and III.

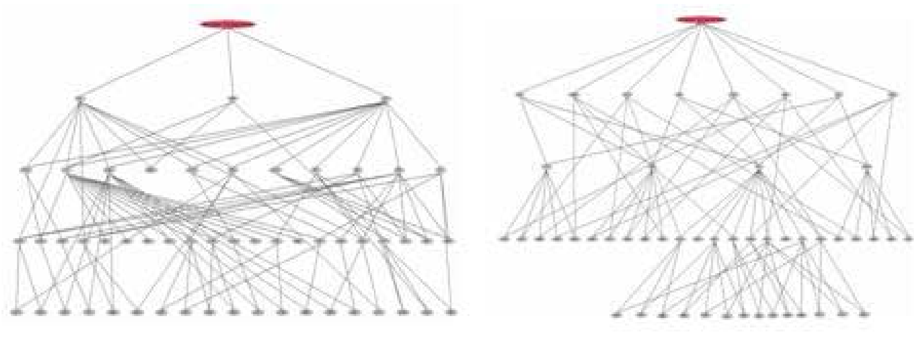


Figure 7 Concept Lattice Generated for Source Datasets

The formal contexts from the two data resources were composed to generate an integrated context with 103 objects and 26 attributes. The disambiguation results show that there were 12 Type I matches, 0 Type II matches, 2 Type III matches, and 5 type IV matches. Multi matches existed for one attribute and it had 1 equivalence match (Type I) and 2 specific matches (Type III). This revealed that utility companies tend to use common set of attributes to describe their asset types, but diversity does exist. Figure 8 shows the pruned concept lattice for the integrated context.

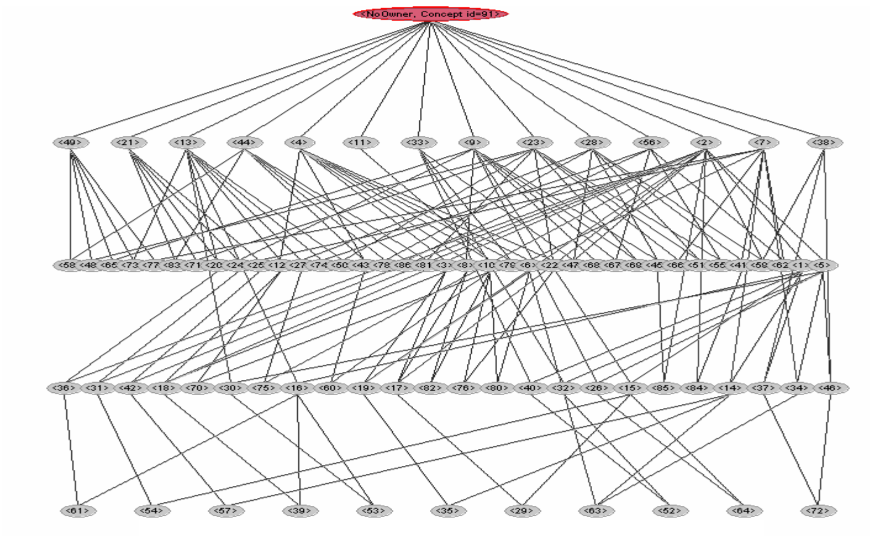


Figure 8 Integrated Concept Lattice

7. Conclusions and Future Work

This paper presents our approach to utility ontology development for the purpose of providing an improved information mapping service for underground utility apparatus records. The research employs formal concept analysis techniques as the backbone for formal concept identification and hierarchy derivation, and the method can also be used to identify utility asset type mapping cross different utilities. Techniques have been proposed to resolve implicit information and ambiguous information, and preliminary experiments have been carried out with the aim to constructing a sewer utility ontology with real world utility data. The techniques have several advantages over traditional knowledge engineering approaches for ontology construction, the key one being that it supports systematic and a semi-automated method for ontology integration.

Ongoing research is on deriving ontology concepts and the concept hierarchy from the integrated lattice with respect to the usefulness of ontology generated. Further work is required to take into account of different data modelling styles when applying proposed techniques. Other research we plan to perform in the future includes testing the techniques proposed with more utility datasets, and evaluation of the ontologies generated with the different approaches.

Acknowledgement

This work is funded by EPSRC Grant EP/C014707/1 and DTI Grant DTI/TSB 15820. Thanks are given to our utility partners for the valuable data and domain knowledge they provided.

References

[1] A. Beck, Gaihua Fu, A. G. Cohn, B. Bennett & J. G. Stell , A framework for utility data integration in the UK, in Proceedings of *UDMS Symposium2007*, pp.261-276, 2007.

[2] M. Burtwell, E. Faragher, D. Neville, C. Overton, C. Rogers, and T. Woodward. Locating Underground Plant and Equipment: Proposals for a Research Programme. *Technical Report, UKWIR*, 2004.

[3] A. Doan, P. Domingos, and A.Y. Halevy, Reconciling Schemas of Disparate Data Sources: A Machine-Learning Approach. *SIGMOD record, 30*(2), pp.509-520, 2001.

[4] Federal Geographic Data Committee, Utilities Data Content Standard. *http://www.fgdc.gov/standards/projects/FGDC-standards-Projects/utilities/utilities.pdf*.

[5] M. Fernandez and A. Gomez-Perez and N. Juristo, METHONTOLOGY: from Ontological Art towards Ontological Engineering, in Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering, pp.33--40, 1997.

[6] B. Ganter and R. Wille (Author), Formal Concept Analysis: Mathematical Foundations, Springer, ISBN-10: 3540627715, 1998.

[7] G. Fu and A.G. Cohn, Semantic Integration for Mapping the Underworld, to appear in *GeoInformatics2008*, http://www.comp.leeds.ac.uk/mtu/pubs/semantic_geoinformatics08.pdf.

[8] A. Gangemi, D. Pisanelli and G. Steve. Ontology Integration: Experiences with Medical Terminologies. in Proceedings of International Conference of FOIS, pp.163-178, 1998.

[9] R. Godin , H. Mili , G. Mineau, R. Missaoui, A. Arfi and T. Chau, Design of class hierarchies based on concept (Galois) lattices, Theory and Practice of Object Systems, Vol.4(2), pp.117 – 134, 2006.

[10] D.M. Jones, T.J.M. Bench-Capon and P.R.S. Visser, Methodologies for Ontology Development, in Proc. IT&KNOWS Conference, XV IFIP World Computer Congress, 1998.

[11] N. Jyotirmaya, S. Timothy, S. Kumara And S. Shooter, A methodology for product family ontology development using formal concept analysis and web ontology language, in Journal of computing and information science in engineering, vol.6(2), pp.103-113, 2006.

[12] Y. Kalfoglou and M. Schorlemmer, Ontology Mapping: The State of the Art, in *Proceedings of Dagstuhl Seminar -Semantic Interoperability and Integration*, 2005.

[13] M. Kavouras and M Kokla, A method for the formalization and integration of geographical categorizations, in *International Journal of GIS,* 16(5), pp.439 – 453, 2002.

[14] M Lenzerini. Data Integration: A Theoretical Perspective. *In Proc. PODS*, pp 233—246, 2002.

[15] J. Madhavan, P. Bernstein and E. Rahm,. Generic schema matching with cupid, in *The VLDB Journal*, pp.49-58, 2001.

[16] N. Noy and D. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology, Technical Report, Stanford, 2001.

[17] N. Noy and M. Musen, PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment, in Proceedings of the 16th National Conference on Artificial Intelligence, pp.450-455, 2000.

[18] N. Noy, Semantic Integration: a Survey of Ontology-Based Approaches, in *SIGMOD Record*,33(4), pp.65-70, 2004.

[19] I. Schmitt and G. Saake, Merging Inheritance Hierarchies for Database Integration**,** in Proceedings of the 3rd IFCIS International Conference on Cooperative Information Systems, pp.322 - 331, 1998.

[20] H. Pinto and J. Martins, A methodology for ontology integration. in Proceedings of the 1st international conference on Knowledge capture, pp.131-138, 2001.

[21] G Stumme and A Maedche, FCA-MERGE: Bottom-Up Merging of Ontologies, in Proc. International Joint Conference On AI, pp.225-234, 2001.

[22] M. Uschold and M. Grüninger, Ontologies: principles, methods, and applications, in Knowledge Engineering Review, Vol 11(2), pp.93-155, 1996.

[23] M. Uschold, Towards a Methodology for Building Ontologies, in Workshop on Basic Ontological Issues in Knowledge Sharing, 1995.

[24] P. Valtchev, D. Grosser, C. Roume and M. Hacene, Galicia: An Open Platform for Lattices, http://citeseer.ist.psu.edu/669715.html.

[25] M. Rouane, P. Valtchev, H. Sahraoui and M. Huchard, Merging conceptual hierarchies using concept lattices, MASPEGHI Workshop*, 2004.*

[26] R. Wille, Restructuring lattice theory: an approach based on hierarchies of concepts, in *Ordered Sets*. Reidel, Dordrecht-Boston, pp.445-470, 1982.

Formal Ontology in Information Systems
C. Eschenbach and M. Grüninger (Eds.)
IOS Press, 2008

© 2008 The authors and IOS Press. All rights reserved.
doi:10.3233/978-1-58603-923-3-311

A proposal for an Interactive Ontology Design Process based on Formal Concept Analysis

Rokia BENDAOUD, Amedeo NAPOLI and Yannick TOUSSAINT
Campus Scientifique B.P. 239,
54506 Vandoeuvre-lès-Nancy Cedex, FRANCE

Abstract. Building a domain ontology usually requires several resources of different types, e.g. thesaurus, object taxonomies, terminologies, databases, sets of documents, etc. where objects are described in terms of attributes and relations with other objects. One important and hard problem is to be able to combine and merge knowledge units extracted from these different resources within the representation formalism supporting the ontology. The purpose of this paper is to show which kinds of resources can be taken as starting points for building an ontology, using FCA and its extension RCA. A real-world example in microbiology is proposed, detailing the interaction with domain experts during the ontology design process. Finally, an evaluation based on recall and precision gives an idea of the efficiency of the approach and points out several research perspectives.

Keywords. Ontology, FCA, RCA, Interaction with experts

1. Introduction

Ontologies are now widely introduced in the semantic web technology as they help software and human agents to communicate and to share domain knowledge [9,12]. In theory, an ontology is considered as an explicit specification of a domain conceptualization [7] represented within a formal language such as description logics (DL). In practice, despite of methodologies and methods for building ontologies [6] as well as concrete experiments [1], representing domain knowledge within a formal language such as description logics remains a very complex task using a manual approach: detecting inconsistencies in formal definitions of concepts may lead to a time-consuming and difficult restructuring of the whole ontology. One way of guiding the design task is to rely on an iterative ontology design process where the expert is only asked for very simple descriptions of the entities. In this way, as this is the case most of the time, the domain expert and the knowledge engineer are asociated for achieving the always complex task of real-world knowledge acquisition.

Building an ontology is an interactive process which requires several iterations before the expert agrees on the target ontology. One strategy, difficult to carry on and to scale, would be for the expert to successively correct the target ontology according to his needs by directly changing the OWL code. However, if he wants to re-built the ontology by adding new resources or updates, the expert previous interactions are lost: they cannot be applied again to the OWL concepts that changed. A better way of considering interactions with the system is to ask the expert to adjust source material and to apply again the ontology design process. In fact, resources data are not modified but a preprocessing step has to be defined and acts as a filter. It enables the experts to perform operations on the source material.

Accordingly, one point of this paper is an iterative approach for building an ontology where the expert is asked to assign – or select into resources – objects, very simple attributes, and relations between two objects. These descriptions can be found in various resources having different types, e.g. thesaurus, vocabularies, dictionaries, sets of documents and databases. Starting with domain objects, a process that automates (in a certain way) the definition and design of concepts is of first importance. The FCA and RCA formalisms can guide the design of the ontology [10,3]. Then, concepts emerging from FCA and RCA can be encoded within a DL formalism. The evolutions of the ontology, *i.e.* addition, modification, and deletion, are not performed on the ontology itself but rather on the source material used to build the ontology. The source material for the ontology consists in a set of prepared data that will be the basis of the ontology design : for example, a binary table `objects` $\times$ `attributes` in FCA. Every time the ontology has to be changed, the source material is changed, the ontology design process is replayed and the target ontology is rebuilt.

Three main types of resources are distinguished in the following: a thesaurus, a database, and a set of documents. In a standard way, the thesaurus provides a set of hierarchically organized classes as `Klebsiella Pneumonae (Klebsiella P.)` is a `Proteobacteria`. The database and the set of documents provide a set of pairs (`object`, `attribute`) (attribute or property) and a set of triples ($\mathtt{object_i}, \mathtt{relation}, \mathtt{object_j}$). For example, the class of `Helicobacter Pylory (Helicobacter P.)` bacteria can be described by pairs such as `(Helicobacter P., aerobic)`, `(Helicobacter P., negativeGram)` and `(Helicobacter P., spherical)`. The relation `Resist` whose co-domain includes ten families of antibiotics is defined by triples such as `(Helicobacter P., resist, Ciprofloxacin)`.

FCA and RCA are the processes on which is based the transformation between source materials towards the target ontology. One important idea on which relies the process is the existence of a "source" or "pivot" ontology extracted from the database or the set of documents, and then to extend the source ontology by progressively adding units extracted from the chosen resources. This source ontology is important with respect to the evaluation of the target ontology resulting from the whole design process. The addition of these units is based on the one hand on standard operations from FCA, such as apposition for example, and on the other hand on non standard operations such as RCA. A lattice with binary and relational attributes results from that process. Then, the elements in the target lattice –built thanks to FCA– can be represented within a knowledge representation language such as OWL. In this way, FCA is considered as the "core" process in the design of the target ontology from a set of heterogeneous resources. Firstly, FCA and RCA as well take into account all elements included within an ontology, namely objects (or individuals), attributes, and relations, for building concept lattices. Secondly, the FCA framework provides operations to manage concept lattices, e.g. updating the lattice when the set of objects or the set of attributes is modified, merging or linking concept lattices. Finally, the resulting concept lattice can be transformed into a concept hierarchy within a description logic (DL) or an OWL concept hierarchy. A classifier can then be used for classification-based reasoning, e.g. answering queries. There are approaches similar to the present work but the novelty here lies in the articulation of the different operations for building up the target ontology.

An operational platform has been designed and an experiment in microbiology is detailed at the end of the paper to show the capabilities of the approach, the efficiency of an FCA-based transformation approach, and the usefulness of expert interactions with the system for reaching a consensus with respect to the target ontology.

The paper is organized as follows. The second section discusses requirements for designing an ontology from a set of heterogeneous resources. The third section introduces FCA and RCA, and the transformation process from a concept lattice to a target concept hierarchy within a DL-based framework. The fourth section presents interaction with experts in a real-world example for the design of a target ontology in microbiology. An evaluation of the ontology design process follows. Related and future work is examined at the end of the paper.

2. Merging simple descriptions to build an ontology

In this section, we analyze the basic objects and the resources that can be considered for building an ontology. For making precise every notion, the application domain chosen in this paper is microbiology. Three main kinds of basic objects are involved, namely genes, bacteria and antibiotics. The current problem is to build an ontology on the base of a collection of heterogeneous resources about resistance of bacteria to antibiotics by genes mutations. For bacteria, the following resources have been considered:

- The NCBI taxonomy (from the National Center for Biotechnology Information) includes 13380 species of bacteria.
- A collection of textual documents composed of 1244 abstracts has been selected by domain experts from PubMed (`http://www.ncbi.nlm.nih.gov/sites/entrez`), with a large collection of texts in the NCBI library.
- The pathogenic bacteria database (`http://bac.hs.med.kyoto-u.ac.jp/`).

For antibiotics, a concept lattice of ligands has been designed based on expert available knowledge (mainly involving chemical properties of antibiotics). For genes, the gene ontology[1] has been used.

[1] http://www.geneontology.org/

2.1. Three main types of object descriptors

Ontologies are usually not built from scratch and several kinds of resources can be used. Actually, the type of the resources does not matter as much as the type of information the resources include. In this paper, three main types of object descriptors are distinguished, (OD1) hierarchical links, (OD2) binary attributes, and (OD3) relational attributes (or binary relations),

(OD1). In an application domain, there are usually existing "source" hierarchies organizing domain objects, e.g. thesaurus or local ontologies from Swoogle[2]. Such hierarchies provide a global and structured view of the domain. In these hierarchies, a class denotes a set of objects and the relation between classes is set inclusion, while objects are instances of the class and all objects in a class are also in the superclasses. For example, `Klebsiella-pneumoniae` (or `Klebsiella-P.`) is a kind of `Proteobacteria`. Such classes can be compared to primitive concepts in description logics, as they do not have any explicit definition. In the context of microbiology, the NCBI taxonomy has played the role of source hierarchy.

(OD2). There are some resources such as databases where domain objects are described by means of a set of attributes. For example, `helicobacter pylori` has the `negativeGram` attribute (in the pathogenic bacteria database).

(OD3). Domain objects are related. Such relations occur in texts, but not exclusively. For example, the sentence "We have previously reported that a significant percentage (44%) of *isoniazid-resistant Mycobacterium tuberculosis* strains carry an arginine to leucine mutation in codon 463 (R463L) in the catalase-peroxidase gene (katG)." indicates that there exists a *resistance* relation from *Mycobacterium tuberculosis* to *isoniazid*. Such relation has been extracted from texts using GATE[3] [2]. It participates to the definition of classes of objects as well as attributes.

2.2. From a pivot ontology to a completed target ontology

The structure of the target ontology and its content has to take into account the three types of descriptors, *(OD1)*, *(OD2)*, and *(OD3)* introduced here-above: hierarchical links, attributes, and relations. Domain objects are grouped into a same class if and only if they share a given set of common attributes and relations. Both attributes and relations are necessary and sufficient conditions for defining a class of objects. For example, let us suppose that the `X` bacteria resists drug `D1`, the `Y` bacteria resists drug `D2`, and that `D1` and `D2` are drugs of the family `D`. In this context, `X` and `Y` can be grouped in the same class as they share the relation "resisting a drug from the class `D`". The resistance relation impacts on the definition of bacteria (here the domain of the relation). This shows in particular that attributes should be combined with relational attributes for forming richer and more precise definitions.

One main idea underlying the design of the target ontology is to rely on a "pivot" or "source" ontology, that will be progressively completed by the concepts

[2] http://swoogle.umbc.edu/
[3] `http://gate.ac.uk/`

extracted from the other resources. In the present framework, the NCBI taxonomy after being processed by FCA (as explained just after) has played the role of source ontology. The other resources that have been analyzed for completing the source ontology hold on genes, bacteria, and drugs.

The purposes of a target ontology depend in part of the type of queries expected to be asked. The structure and the content of the present target ontology should allow to ask three main types of queries.

- *(Q1).* Let $\mathtt{o}_1$ and $\mathtt{o}_2$ be two domain objects. Does it exist a class containing both objects or are these objects incompatible? What are the other objects in the common class. How is defined this common class?
- *(Q2).* Given a new object, say $\mathtt{x}$, that has been observed with some attributes and relations with other objects. What is the best and the right way of inserting this object in the ontology? Is there a class already available for this object or a new class has to be created?
- *(Q3).* What is the class of an object knowing the domain and/or the range of a relation. In particular, when $\mathtt{r}_1(\mathtt{o}_1,\mathtt{o}_2)$ and $\mathtt{o}_1$ is an instance of $\mathtt{C}_1 = \forall\mathtt{r}_1.\mathtt{A}_1$, then it can be inferred that $\mathtt{o}_2$ is an instance of $\mathtt{A}_1$.

3. Formal Concept Analysis

Formal Concept Analysis (FCA) and its extension Relational Concept Analysis (RCA) take into account the three main types of object descriptors discussed in Section 2. The FCA process builds concept lattices and provides various operations for managing concept lattices, in particular merging sets of objects or sets of attributes. RCA extends the scope of FCA by taking into account relational attributes. Moreover, the resulting concept lattice can be transformed into a concept hierarchy represented within the description logic formalism for allowing formal representation and reasoning.

3.1. Formal Concept Analysis

Formal concept analysis (FCA) [5] is a mathematical formalism allowing to derive a concept lattice from a formal context $\mathbb{K} = (G, M, I)$. FCA has been used for a number of purposes among which knowledge modeling, acquisition, and processing, lattice and ontology design, information retrieval, and data mining. In $\mathbb{K}$, G denotes a set of objects, M a set of attributes, and I a binary relation defined on the Cartesian product $G \times M$. In the binary table representing $I \subseteq G \times M$, the rows correspond to objects and the columns to attributes. The concept lattice is composed of *formal concepts* (or simply *concepts*) organized into a lattice by a partial ordering, i.e. a subsumption relation comparing concepts. A concept is a pair (A, B) where $A \subseteq G$, $B \subseteq M$, and A is the maximal set of objects sharing the whole set of attributes in B (and vice versa). In a concept (A, B), A is called the *extent* and B the *intent* of the concept. The concepts in a concept lattice are computed on the basis of a *Galois connection* defined by two derivation operators denoted by $'$:

$$' : A' = \{m \in M | \forall g \in A : (g, m) \in I\}$$
$$' : B' = \{g \in G | \forall m \in B : (g, m) \in I\}$$

A concept (A, B) verifies $A' = B$ and $B' = A$. The subsumption relation ($\sqsubseteq$) between a concept and a superconcept is defined as follows: $(A_1, B_1) \sqsubseteq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2 (or\ B_2 \subseteq B_1)$. Relying on this subsumption relation $\sqsubseteq$, the set of all concepts extracted from a context $\mathbb{K} = (G, M, I)$ is organized within a complete lattice, called *concept lattice* and denoted by $\mathfrak{B}(G, M, I)$.

The standard FCA process is able to deal with object descriptors of type *(OD1)* or *(OD2)*. Given a set of resources including such object descriptors, concept lattices provide a representation of the content of these resources. Then, the content of these resources can be merged using the FCA operation called *apposition*, as explained below.

	Proteobacteria	γproteobacteria	Actinobacteria	Bacilli
Helicobacter P.	X			
Klebsiella P.	X	X		
Mycobacterium S.			X	
Streptococcus P.				X
Klebsiella O.	X	X		

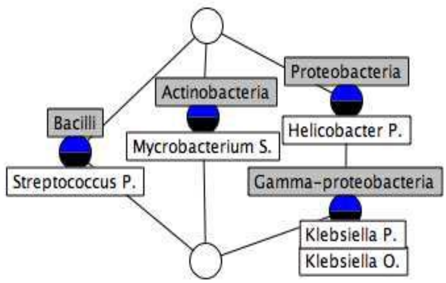


Figure 1. The context Bacteria from the database NCBI $\mathbb{K}_1 := (G, M_1, I_1)$ and the associated concept lattice.

Building a lattice from a hierarchy (OD1 object descriptor). Transforming a set of objects organized within a hierarchy –or described by hierarchical links– into a lattice is a straightforward operation. The formal context $\mathbb{K}_1 := (G, M_1, I_1)$ is defined as follows: G is the set of domain objects, M_1 is the set of classes of objects, and I_1 assigns to an object its class and all superclasses in the hierarchy. For example, the bacteria `Klebsiella P.` is classified in the NCBI hierarchical resource as a `GammaProteobacteria`, which in turn is a subclass of `proteobacteria`. Figure 1 shows the context associated to NCBI classification and the corresponding concept lattice.

Building a lattice from domain expert description of objects (OD2 object descriptor). A classification based on domain expert description of objects, i.e. involving *(OD2)* object descriptors, can be carried out as follows. A formal context $\mathbb{K}_2 := (G, M_2, I_2)$ is composed of a set G of objects, a set M_2 of attributes, and a relation $I_2 \subseteq G \times M_2$ where $I_2(g, m_2)$ states that g has the attribute m_2 (actually, the set G of objects is the same for context $\mathbb{K}_1$ and $\mathbb{K}_2$). Figure 2 shows an excerpt of such a context describing various bacteria, their attributes, and the corresponding concept lattice. In the present case, this concept lattice has been built for associating characteristics attributes to bacteria according to expert domain knowledge.

	spherical	sticks	negativeGram	positiveGram	aerobic	anaerobic
Helicobacter P.	×		×		×	
Klebsiella P.		×	×			×
Mycobacterium S.		×		×	×	
Streptococcus P.		×		×	×	
Klebsiella O.		×	×			×

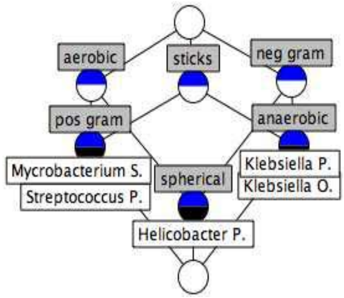


Figure 2. The context Bacteria based on expert knowledge $\mathbb{K}_2 = (G, M_2, I_2)$ and the associated concept lattice.

3.2. Merging two lattices with apposition in FCA

At this point, there are two contexts $\mathbb{K}_1 := (G, M_1, I_1)$ and $\mathbb{K}_2 := (G, M_2, I_2)$, with the same set of objects G and two distinct sets of attributes, M_1 and M_2 $(M_1 \cap M_2 = \emptyset)$. The apposition operation is used in FCA for merging two contexts with the same set of objects and disjoint sets of attributes into a single context [5].

Definition 1 *Let* $\mathbb{K}_1 = (G_1, M_1, I_1)$ *and* $\mathbb{K}_2 = (G_2, M_2, I_2)$ *be two formal contexts. When* $G = G_1 = G_2$ *and* $M_1 \cap M_2 = \emptyset$, $\mathbb{K} := \mathbb{K}_1 | \mathbb{K}_2 := (G, M_1 \cup M_2, I_1 \cup I_2)$ *is the* apposition *of the two contexts* $\mathbb{K}_1$ *and* $\mathbb{K}_2$*.*

The two contexts are $\mathbb{K}_1 = (G, M_1, I_1)$ shown in Figure 1 and $\mathbb{K}_2 = (G, M_2, I_2)$ shown in Figure 2. In the apposition context $\mathbb{K} = (G, M, I)$, G is the set of objects –the same set for $\mathbb{K}_1$ and $\mathbb{K}_2$– $M := M_1 \cup M_2$ where M_1 is the set of attributes in $\mathbb{K}_1$ and M_2 is the set of domain attributes in $\mathbb{K}_2$, and $I := I_1 \cup I_2$. The resulting concept lattice is presented in Figure 3.

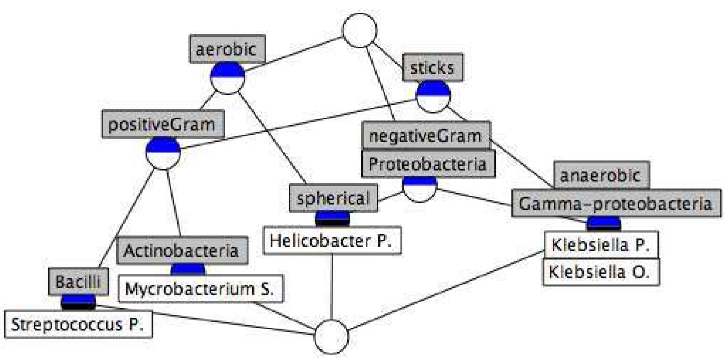


Figure 3. The concept lattice resulting from the apposition of contexts $\mathbb{K}_1$ and $\mathbb{K}_2$.

Table 1. The relation "Resist" between bacteria and antibiotics.

Resist	Clarithromycin	Ciprofloxacin	Cefotaxim	Macrolide
Helicobacter-P.		×		
Klebsiella-P.				×
Mycobacterium-S.			×	
Streptococcus-P.		×		
Klebsiella-O.	×			

3.3. Relational Concept Analysis

Relational Concept Analysis (RCA) [10] was introduced as an extension of FCA for taking into account relations between objects. A concept is then described with standard binary attributes and also with relational attributes. A relational attribute, say r, describes the relation existing between objects that are instances of a concept, say c_1, the domain of the r relation, with objects that are instances of another concept, say c_2, the range of r relation. RCA was already been used in a previous work on text mining and ontology design [2].

	FRB1	FRB3	ARB1	ARB2	HBA5	HBA10
Clarithromycin		×				×
Ciprofloxacin	×			×	×	
Cefotaxim		×	×			×
Macrolide			×			×

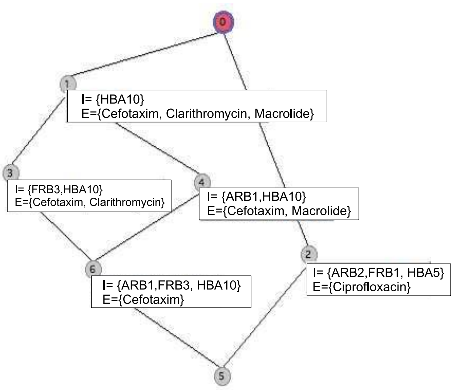


Figure 4. The context Antibiotics $\mathbb{K}_3 = (G_3, M_3, I_3)$ and the associated concept lattice.

Data in RCA are organized within a *relational context family* (RCF) composed of a set of contexts $\mathbb{K}_i = (G_i, M_i, I_i)$ and a set of relations $r_k \subseteq G_i \times G_j$. The sets G_i and G_j are the object sets of the contexts $\mathbb{K}_i$ and $\mathbb{K}_j$, called respectively the *domain* and the *range* of the relation r_k.

RCA uses the mechanism of *relational scaling* for defining relational attributes. For a relation, say $r : G_i \longrightarrow G_j$, linking objects from G_i to objects of G_j, a relational attribute is created and denoted by $r : c$, where c is concept in $\mathfrak{B}(G_j, M_j, I_j)$. Then, for an object $g \in G_i$, the relational attribute $r : c$ characterizes the "correlation" between g and $r(g) = h$ which is an instance of the concept $c = (X, Y)$ in $\mathfrak{B}(G_j, M_j, I_j)$. Many levels of correlation can be considered such as the "existential correlation" –or existential scaling– where $r(g) \cap X \neq \emptyset$, and the "universal correlation" –or universal scaling– where $r(g) \subseteq X$. In the present work, only existential scaling is considered.

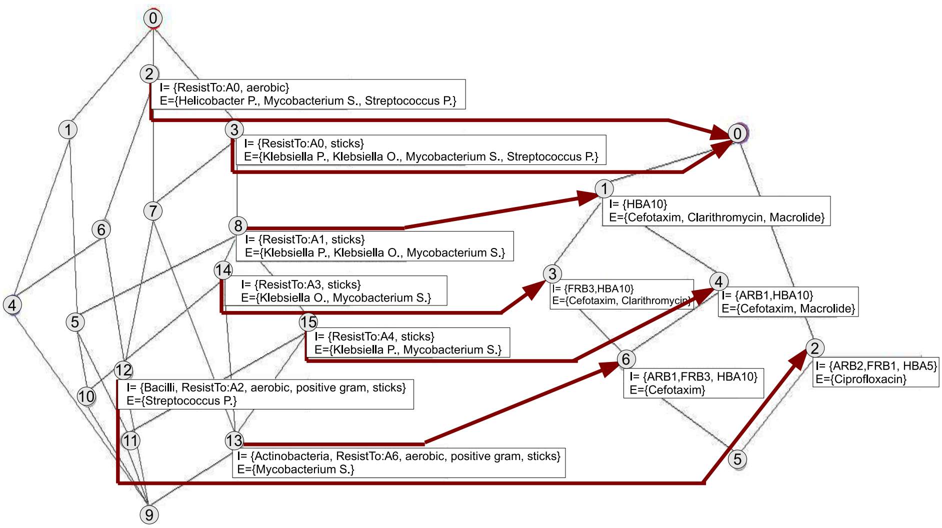


Figure 5. The lattice resulting from the RCA process applied to object descriptors of type *(OD3)*.

Let us consider the relation between bacteria and antibiotics, where the first context is given by context apposition in Figure 3 and the second context $\mathbb{K}_3 = (G_3, M_3, I_3)$ is given in Figure 4. The relation `Resist` between bacteria and antibiotics is given in Table 1. The application of RCA on the contexts of Figure 3 and Figure 4 produces the final concept lattice shown in Figure 5, where the relations explicitly computed by the RCA process are emphasized.

3.4. From concept lattice to DL formalism

The transformation of the final concept lattice resulting from RCA is based on a transformation into a DL knowledge base (KB) [10,11,8]. This transformation allows to introduce primitive and defined concepts, and thus to apply a DL-based reasoner for problem-solving and complex query answering. The target DL formalism is $\mathcal{FLE}$, that includes the constructors $\top$ (top), $\bot$ (bottom), $C \sqcap D$ (concept conjunction), $\forall r.C$ and $\exists r.C$ (universal and existential role quantifications). This set of constructors is large enough for representing all elements from the final concept lattice.

4. Interpretation and evaluation

4.1. Expert interaction with the system

The expert is invited to interpret the target ontology and to identify points in the ontology where there may be no agreement on the classification of objects or on the definition of classes. The reasons of these conflicts are: (1) there may be noise in resources or in the information extraction processes, (2) the expert is not satisfied with the target ontology and wants it to be more in accordance with his needs. In both case, the expert may apply elementary operations on the source material, and then run the FCA/RCA process for obtaining an updates version of the ontology.These operations depends on the object descriptors and are the following:

Operations on hierarchical link resources (OD1).

- *Adding a new class.* A new class is considered in the source hierarchy. This leads to add a new column to the formal context representing this hierarchy the. Then, expert has to assign to this new class the appropriate objects.
- *Changing the class of an object.* When changing the class of an object, the line describing the object in the formal context has to be modified: the new class and all its superclass have to be properly assigned to the object.
- *Deleting a class.* This operation was not used in this experiment. Deleting a class in the source hierarchy is equivalent to a deletion of a column in the formal context describing the resource.

Modifying attributes (OD2 or OD3). Quality of resources may depend on their form: database, text... For example, Natural Language Processing tools extracting information from texts are noisy when the linguistic level is too detailed compared to the ontological level. Some purely linguistically relevant information are deleted by the experts and some other may be introduced. The following operations can be used by experts:

- *Merging attributes.* This operations is relatd to synonymy in the texts. Expert may decide to merge the `positiveGram` with the `neutralGram` attribute for avoiding over-spliting classes in the target ontology.
- *Deleting an attribute for an object.* An attribute has been wrongly assigned to an object while extracting information from a resource; experts want to remove it. In the formal context describing this resource the cell (object,property) is changed to "blank".
- *Deleting an attribute for all objects.* The expert while interpreting the ontology observes that an attribute is not relevant. The column with this attribute in the formal context is simply deleted.
- *Adding an attribute to a set of objects.* The expert considers that an attribute is missing in a class. Either it is missing in the resources, either it has not been extracted (from texts). This operation is used for adding a column in the formal context and the attribute has to be assigned to the appropriate objects.

Operations on relational attributes are similar to operations on attributes. With this set of operations, the systel is able to meet the expert requests to converge towards the final ontology.

4.2. Expert interpretation

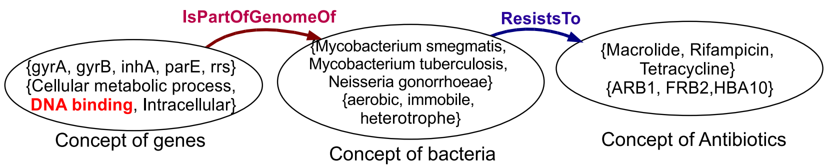


Figure 6. An example of an interpretation for the link between three classes : genes-bacteri-a-antibiotics

In this section, classes resulting from the lattices are presented and discussed. In the example of Figure 6, the expert found an explanation for the resistance of the set of bacteria `{Mycobacterium smegmatis, Mycobacterium tuberculosis, Neisseria gonorrhoeae}` to the set of antibiotics `{Macrolide, Rifampicin, Tetracycline}`. The explication is : the set of antibiotics `{Macrolide, Rifampicin, Tetracycline}` kill bacteria by detroying the DNA and the fact that the set of genes `{gyrA, gyrB, inhA, parE, rrs}` has the property of binding DNA allows bacteria to resist to antibiotics.

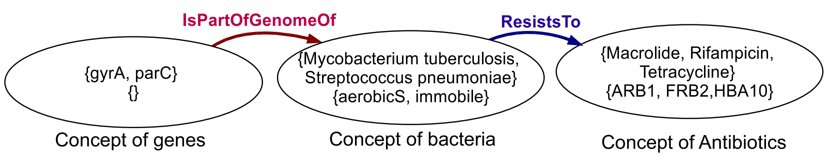


Figure 7. An other example of an interpretation for the link between three classes : genes-bacteria-antibiotics

In the second example on Figure 7, the set of genes `{gyrA, parC}` has not common attribute but the expert found this set interresting because it is knoown that there exists a strong relation between these two genes, and the first cannot be found without the second.

Another example is given by the concept `{Citrobacter freundii, Enterobacter aerogenes, Enterobacteriaceae, Escherichia coli, Pseudomonas aeruginosa, Salmonelle typhimurium, Serratia marcescens} {ResistTo:c4, ResistTo:c7, batonnet, gramNeg, hétérotrophe, mobile}`. The expert did not consider this class as interesting beacause these bacteria are different but there were no discriminant and characteritic attribute for separating these bacteria. One proposition of the expert was to add the attribute “activity Oxydase”.

5. Related work and Conclusion

In this paper, we have presented an original approach for building a target domain ontology in considering resources of different types, such as a thesaurus, term hierarchies, databases, and sets of documents.There are some work similar to the present one.

In [4], the authors use an approach which is able to acquire semantic knowledge from syntactic parsing and they use then FCA for building the concept hierarchy. Our approach deals with FCA, but uses in addition RCA and takes into account heterogeneous resources.

In [13], the authors propose to merge two ontologies for building a new one. The proposed method takes as input a set of documents. NLP techniques are used to capture two formal contexts encoding the relationships between documents and concepts in each ontology. This method combines the knowledge of the collection of texts and expert knowledge. This approach uses texts for merging and not for enriching the two ontologies.

In our framework, the resources are heterogeneous. Objects are described in terms of attributes and relations with other objects. Using FCA and its extension RCA, these different resources are transformed into source material and then represented as concept lattices. These concept lattices are used for completing a chosen reference concept lattice, that is the basis of the target ontology. This final concept lattice is transformed within a description logic formalism. Complex question-answering and classification-based reasoning can then be carried out using the classifier in the framework of description logics. A real-world example in microbiology has been detailed, showing the capabilities of the approach.

References

[1] N. Aussenac-Gilles, B. Biébow, and S. Szulman. Revisiting ontology design: A method based on corpus analysis. In Dieng R. and O. Corby, editors, *12th Internationnal Conference in Knowlegde Engineering and knowledge Management (EKAW'00)*, volume 1937, pages 172–188, 2000.

[2] R. Bendaoud, M. Rouane-Hacene, Y. Toussaint, B. Delecroix, and A. Napoli. Text-based ontology construction using relational concept analysis. In G. Flouris and M. d'Aquin, editors, *Proceedings of the International Workshop on Ontology Dynamics, Innsbruck (Austria)*, pages 55–68, 2007.

[3] R. Bendaoud, Y. Toussaint, and A. Napoli. Pactole: A methodology and a system for semi-automatically enriching an ontology from a collection of texts. In P. Eklund and O. Haemmerlé, editors, *Proceedings of the 16th International Conference on Conceptual Structures, ICCS 2008, Toulouse, France*, Lecture Notes in Computer Science 5113, pages 203–216, 2008.

[4] P. Cimiano, A. Hotho, and S. Staab. Learning concept hierarchies from text corpora using formal concept analysis. *Journal of Artificial Intelligence Research (JAIR)*, 24:305–339, 2005.

[5] B. Ganter and R. Wille. *Formal Concept Analysis, Mathematical Foundations.* Springer, 1999.

[6] A. Gómez-pérez, M. Fernández-Lòpez, and O. Corcho. *Ontological Engineering.* Springer Verlag, 2004.

[7] T. Gruber. Toward principales for the design of ontologies used for knowledge sharing. In Guarino N. and R. Poli, editors, *Formal Analysis in Conceptual Analysis and Knowledge Representation*, The Netherlands, 1993. Kluwer Academic.

[8] M. Huchard, A. Napoli, M. Rouane-Hacene, and P. Valtchev. Mining description logics concepts with relational concept analysis. In P. Brito, P. Bertrand, G. Cucumel, and F. De Carvalho, editors, *Selected Contributions in Data Analysis and Classification*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 259–270. Springer, Berlin, 2007.

[9] A. Maedche. *Ontologies Learning for the Semantic Web.* Springer, 2002.

[10] M. Rouane-Hacene, M. Huchard, A. Napoli, and P. Valtchev. A proposal for combining formal concept analysis and description logics for mining relational data. In S.O. Kuznetsov and S. Schmidt, editors, *Proceedings of the 5th International Conference on Formal Concept Analysis (ICFCA 2007), Clermont-Ferrand*, LNAI 4390, pages 51–65. Springer, Berlin, 2007.

[11] M. Rouane-Hacene, A. Napoli, P. Valtchev, Y. Toussaint, and R. Bendaoud. Ontology learning from text using relational concept analysis. In P. Kropf, M. Benyoucef, and H. Mili, editors, *International Conference on eTechnologies (MCETECH 08), Montréal*, pages 154–163. IEEE Computer Society, 2008.

[12] S. Staab and R. Studer, editors. *Handbook on Ontologies.* Springer, Berlin, 2004.

[13] G. Stumme and A. Maedche. Fca-merge: Bottom-up merging of ontologies. In *Proceedings of IJCAI'01, Seattle (WA)*, pages 225–234, 2001.

This page intentionally left blank

© 2008 The authors and IOS Press. All rights reserved.

Author Index

This page intentionally left blank

This page intentionally left blank