Frontiers in Artificial Intelligence and Applications

FORMAL ONTOLOGY IN INFORMATION SYSTEMS

Proceedings of the 11th International Conference (FOIS 2020)

Edited by
Boyan Brodaric
Fabian Neuhaus

IOS Press

FORMAL ONTOLOGY IN INFORMATION SYSTEMS

FOIS is the flagship conference of the International Association for Ontology and its Applications, a non-profit organization which promotes interdisciplinary research and international collaboration at the intersection of philosophical ontology, linguistics, logic, cognitive science, and computer science, as well as in the applications of ontological analysis to conceptual modeling, knowledge engineering, knowledge management, information-systems development, library and information science, scientific research, and semantic technologies in general.

This volume presents the 17 papers accepted for the 11th Formal Ontology in Information Systems conference (FOIS 2020). These papers cover a broad range of topics and are organized into 5 groups. *Foundations* is dedicated to the general ontological decisions providing a foundation for any ontology, both from a philosophical perspective and with an emphasis on applications. *Social Entities* is dedicated to the ontological analysis and formalization of various social entities, including secrets, legal theories, decisions, kinship, and cultural heritage. The papers in *Intentionality and Embodiment* analyze aspects of an agent's intentions, beliefs and desires, as well as the embodiment of functional relations. The section on *Parts and Wholes* is dedicated to mereology as well as the mereological analysis of certain types of entities (e.g., pluralities, information entities, and computer programs). Lastly, the papers in *Methods* are about ontology evaluation and use.

Altogether, the papers reflect traditional FOIS themes with perhaps a greater emphasis on social and agent aspects, and will be of interest to all those whose work involves ontology and its applications.

9 781643 681283

ISBN 978-1-64368-128-3 (print)
ISBN 978-1-64368-129-0 (online)
ISSN 0922-6389 (print)
ISSN 1879-8314 (online)

FORMAL ONTOLOGY IN INFORMATION SYSTEMS

Frontiers in Artificial Intelligence and Applications

The book series Frontiers in Artificial Intelligence and Applications (FAIA) covers all aspects of theoretical and applied Artificial Intelligence research in the form of monographs, selected doctoral dissertations, 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 European Conference on Artificial Intelligence (ECAI) proceedings volumes, and other EurAI (European Association for Artificial Intelligence, formerly ECCAI) sponsored publications. The series has become a highly visible platform for the publication and dissemination of original research in this field. Volumes are selected for inclusion by an international editorial board of well-known scholars in the field of AI. All contributions to the volumes in the series have been peer reviewed.
The FAIA series is indexed in ACM Digital Library; DBLP; EI Compendex; Google Scholar; Scopus; Web of Science: Conference Proceedings Citation Index – Science (CPCI-S) and Book Citation Index – Science (BKCI-S); Zentralblatt MATH.

Volume 330

Recently published in this series

Vol. 329. A.J. Tallón-Ballesteros (Eds.), Modern Management based on Big Data I – Proceedings of MMBD 2020
Vol. 328. A. Utka, J. Vaičenonienė, J. Kovalevskaitė and D. Kalinauskaitė (Eds.), Human Language Technologies – The Baltic Perspective – Proceedings of the Ninth International Conference Baltic HLT 2020
Vol. 327. H. Fujita, A. Selamat and S. Omatu (Eds.), Knowledge Innovation Through Intelligent Software Methodologies, Tools and Techniques – Proceedings of the 19th International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques (SoMeT_20)
Vol. 326. H. Prakken, S. Bistarelli, F. Santini and C. Taticchi (Eds.), Computational Models of Argument – Proceedings of COMMA 2020
Vol. 325. G. De Giacomo, A. Catala, B. Dilkina, M. Milano, S. Barro, A. Bugarín and J. Lang (Eds.), ECAI 2020 – 24th European Conference on Artificial Intelligence – 29 August–8 September 2020, Santiago de Compostela, Spain
Vol. 324. C. Troussas and C. Sgouropoulou, Innovative Trends in Personalized Software Engineering and Information Systems – The Case of Intelligent and Adaptive E-learning Systems

ISSN 0922-6389 (print)
ISSN 1879-8314 (online)

Formal Ontology in Information Systems

Proceedings of the 11th International Conference (FOIS 2020)

Edited by

Boyan Brodaric

Geological Survey of Canada, Canada

and

Fabian Neuhaus

Otto-von-Guericke University of Magdeburg, Germany

IOS Press

Amsterdam • Berlin • Washington, DC

ISBN 978-1-64368-128-3 (print)
ISBN 978-1-64368-129-0 (online)
Library of Congress Control Number: 2020947181
doi: 10.3233/FAIA330

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

For book sales in the USA and Canada:
IOS Press, Inc.
6751 Tepper Drive
Clifton, VA 20124
USA
Tel.: +1 703 830 6300
Fax: +1 703 830 2300
sales@iospress.com

PRINTED IN THE NETHERLANDS

Preface

This volume contains the papers accepted for the 11th edition of the Formal Ontology in Information Systems conference, FOIS 2020, intended to occur in Bolzano, Italy, 14th-17th September 2020. FOIS 2020 was to be an integral part of the Bolzano Summer of Knowledge, which included a broad range of conferences, workshops and summer schools pertaining to knowledge representation.

Sadly, FOIS 2020 was not able to occur as planned. The year 2020 was inimically shaped by the COVID-19 virus, which – at the time of this writing – has infected millions and killed hundreds of thousands. As both long-distance travel and large-group meetings contributed to the spread of the pandemic, it became quickly apparent that it would be impossible to organize FOIS as a physical event in Bolzano in 2020. For these reasons the FOIS 2020 physical event was cancelled.

However, in consultation with the International Association of Ontology and its Applications (IAOA), the professional association governing FOIS, the decision was made to publish the FOIS 2020 papers as a high-quality conference proceeding in 2020 (this volume), roughly at the originally planned time of the conference.

Furthermore, in recognition of FOIS' important social function within the applied ontology community, which, for example, a virtual event could not satisfy, it was also decided to run a FOIS edition in September 2021. FOIS 2021 will happen in Bolzano, and it will include all the events originally planned for FOIS 2020 including the *Early Career Symposium*, the *Ontology Show and Tell*, as well as the *Demo and Industrial Track*. And, of course, there will be another call for research papers in early 2021. Authors of papers accepted for FOIS 2020 will also have the option to present their work at FOIS 2021.

Overview of Accepted Papers

For FOIS 2020 we accepted 17 of 42 research paper submissions, which is an acceptance rate of 40.4%. As usual in FOIS, the papers cover a broad range of topics. For the purpose of organizing this volume we grouped them into the following categories:

- Foundations
- Social Entities
- Intentionality and Embodiment
- Parts and Wholes
- Methods

These categories reflect traditional FOIS themes, with perhaps a greater emphasis in this edition on social and agent aspects. They also reflect a decline in consideration of other topics, such as physical or abstract entities, as well as a reduction in applied contributions.

Foundations

This first section is dedicated to the general ontological decisions providing a foundation for any ontology, with the opening two papers being more philosophically oriented. In *An analysis of the debate over structural universals* Garbacz provides an overview of the debate on structural universals, and classifies the various theories of structural universals by their main facets. Next, Toyoshima provides an overview of criteria for the distinction between 3D- and 4D-entities in *Foundations for Ontology of Persistence: Beyond Talk of Temporal Parts*. In the light of these criteria he compares the ways BFO, DOLCE and GFO distinguish between 3D- and 4D-entities. The remaining two papers in this section concern progress on a foundational ontology and an analysis of ontology languages. In preparation for their release of GFO 2.0, Burek, Loebe, and Herre summarise important research results in *Towards GFO 2.0: Architecture, Modules and Applications*, and also discuss how GFO 2.0 will shift to a modular architecture. In *An analysis of commitments in ontology language design*, Fillottrani and Keet compare popular ontology languages along various criteria, and focus on the ontological commitments embedded in an ontology language.

Social Entities

This section contains four papers providing an ontological analysis and formalization of various social entities, including secrets, legal theories and decisions, kinship, and cultural heritage. The section begins with the Best Paper award winner *A Commonsense Theory of Secrets*, by Ismail and Shafie, in which a secret is a 5-ary relation consisting of a proposition hidden by one group from another group while some condition is met at a time. In *Legal Theories and Judicial Decision-Making: An Ontological Analysis*, Griffo, Almeida and Guizzardi show how different legal theories underpinning two distinct ontologies can support judicial decisions. In *An Ontology for Formal Models of Kinship*, Chui, Gruninger and Wong develop a kinship ontology in first-order logic, one inspired by anthropological models as well as related algebraic structures, and capable of representing a variety of family relations. The final submission in this section, by Sanfilippo, Pittet and Markhoff, *Ontological analysis and modularization of CIDOC-CRM*, carries out a formal analysis of the CIDOC standard ontology for culture heritage data modelling and proposes a modularization of the ontology.

Intentionality and Embodiment

The group of papers in this section analyze aspects of an agent's intentions, beliefs and desires, as well as the embodiment of functional relations. An ontological analysis of needs is developed by Biccheri, Ferrario and Porello in *Needs and intentionality – An ontological analysis and an application to public services*, which draws on philosphical work in intentionality and is represented in the DOLCE ontology. *Foundations for an Ontology of Belief, Desire and Intention*, by Toyoshima, Barton and Grenier, develops an ontological framework and formalism in which beliefs and desires have dispositional and occurrent aspects, while intentions are dispositional. Turning from intentions to embodiment in the final contribution in this section, Pomarlan and Bateman propose a formalism for embodied functional relations, such as containment and support, in *Embodied* *functional relations: a formal account combining abstract logical theory with grounding in simulation*. The formalism combines high level abstraction with simulation to help address relevant questions faced by agents.

Parts and Wholes

In *A Mereology for Connected Structures* Grüninger, Chui, Ru, and Thai argue that in some domains – contrary to classical mereology – some underlapping objects do not have a sum. They propose a mereotopology (and a corresponding mereology), which requires sums of underlapping objects to be connected. The other three papers in this section are not about mereology per se, but are about the composition of certain types of entities, with mereological analysis of these entities at their core. In *Collectives, Composites and Pluralities* Masolo, Vieu, Ferrario, Borgo, and Porello analyse the difference between composites (e.g., a car), collectives (e.g., an orchestra), and the pluralities (i.e., non-atomic objects) that constitute them. *The mereological structure of informational entities* by Barton, Toyoshima, Vieu, Fabry and Ethier presents an axiomatization of a mereology for information entities; in particular for information entities like official forms, which are structured by information slots and information fillers. A particular kind of information entity, namely the computer program, is the subject of Keet's *The computer program as a functional whole*. She argues that a computer program, which consists of many individual source files, is not a collection (or set) of artifacts, but a functional whole.

Methods

The two papers in this section deal with ontology evaluation and use. The first paper, *A Study of Two Spatial Ontologies* by Stephen and Hahmann, analyzes the key impediments to verifying first-order logic ontologies via model-finding, develops an approach to minimize the impediments, and demonstrates benefits of the approach with tests on two spatial ontologies. The second and last paper in the section, and in this FOIS volume, *Ontology-Driven Cross-Domain Transfer Learning* by Fumagalli, Bella, Conti and Giunchiglia, develops and tests an approach to leverage ontologies within machine-learning, to facilitate re-use (transfer) of models across different machine-learning tasks.

Best Paper

The FOIS 2020 Best Paper award is given to Haythem Ismail and Merna Shafie for their contribution entitled *A Commonsense Theory of Secrets*. This paper, which presents a new ontological analysis and formal representation of secrets, received the highest scores from reviewers and garnered the most award nominations. In addition, while it fits neatly into a major theme for this FOIS – the ontological analysis of social entities – the topic of secrets is relatively novel and thus quite interesting, and the paper's clear explanations should make it accessible to a wide audience. We congratulate the authors on their winning contribution.

Possible Future Directions

For FOIS 2020 we introduced a rebuttal phase during the reviewing process. It allowed authors to respond to initial reviews prior to their finalization and before a decision was made on a paper. This change was received positively overall, and is likely to be continued in FOIS 2021.

The large majority of both submissions and the accepted papers for FOIS 2020 are theoretical in nature. Thus, this volume is strong on *Formal Ontology*, while the *Information Systems* aspect of FOIS is somewhat underrepresented. Future editions of FOIS might attempt to address this imbalance: by soliciting more papers that focus on methods and tools to build and use ontologies, as well as descriptions, evaluations, and implementations of non-foundational ontologies.

Acknowledgements

Authors of all submitted papers, accepted or not, are sincerely thanked for their submissions. These not only enable the conference program to be built, but also serve to keep the conference series robust and current, while bolstering the applied ontology community.

Conferences such as FOIS also rely heavily on the diligent work of the organizing committee, who are especially thanked for their exceptional efforts during the special circumstances of the COVID pandemic. This includes both the general chair (Roberta Ferrario), as well as the chairs of various tracks, some of whom were deep into track organization only to have their track postponed until 2021. It also includes members of the program committee, who collectively reviewed all paper submissions in concert with a small number of external reviewers. A special round of thanks is also owed to the publicity chairs and the local organizers, who were perhaps most impacted by the special circumstances and made considerable adjustments on-the-fly to enable this FOIS to occur. Last, but not least, we would like to thank Megan Katsumi, the proceedings chair, whose aid was instrumental in the creation of this volume. A full listing of the organizing committee is included after this preface.

FOIS 2020, like its recent predecessors, is organized under the auspices of the IAOA. IAOA not only provides a governance framework for FOIS, but is a source of invaluable guidance during all stages of the conference. We also thank IOS Press for its continued support in the publication of the FOIS proceedings. The following sponsors are also gratefully acknowledged: the Free University of Bozen-Bolzano as well as its KRDB Research Centre for Knowledge and Data, and the Italian National Lab for Artificial Intelligence and Intelligent Systems.

Boyan Brodaric
Fabian Neuhaus

FOIS 2020 Conference Organization

General Chair

Roberta Ferrario	ISTC-CNR Trento, Italy

Program Chairs

Boyan Brodaric	Geological Survey of Canada, Canada
Fabian Neuhaus	Otto-von-Guericke University of Magdeburg, Germany

Local Chairs

Oliver Kutz	Free University of Bozen-Bolzano, Italy
Nicolas Troquard	Free University of Bozen-Bolzano, Italy

Publicity Chairs

Maria M. Hedblom	Free University of Bozen-Bolzano, Italy
Guendalina Righetti	Free University of Bozen-Bolzano, Italy

Proceedings Chair

Megan Katsumi	University of Toronto, Canada

Workshop and Tutorial Chairs

Torsten Hahmann	University of Maine, USA
Claudio Masolo	ISTC-CNR Trento, Italy

Early Career Symposium Chairs

Pawel Garbacz	Catholic University of Lublin, Poland
Selja Seppälä	University College Cork, Ireland

Ontology Show and Tell Chairs

Michael Grüninger	University of Toronto, Canada
Frank Loebe	University of Leipzig, Germany

Demo and Industrial Track Chairs

Robert Hoehndorf	King Abdullah University of Science and Technology, Saudi Arabia
Amanda Vizedom	Credit Suisse, USA

Programme Committee

Joao Paulo Almeida	Federal University of Espirito Santo, Brazil
Mauricio Almeida	UFMG, Brazil
Nathalie Aussenac-Gilles	IRIT CNRS, France
Adrien Barton	IRIT CNRS, France
Colin Batchelor	Royal Society of Chemistry, UK
John Bateman	University of Bremen, Germany
Thomas Bittner	SUNY Buffalo, USA
Olivier Bodenreider	US National Library of Medicine, USA
Stefano Borgo	ISTC-CNR Trento, Italy
Boyan Brodaric	Geological Survey of Canada, Canada
Vinay Chaudhri	Stanford University, USA
Richard Dapoigny	LISTIC / Polytech'Savoie, France
Sergio de Cesare	University of Westminster, UK
Maureen Donnelly	SUNY Buffalo, USA
Antony Galton	University of Exeter, UK
Aldo Gangemi	Università di Bologna & ISTC-CNR, Italy
Pawel Garbacz	Catholic University of Lublin, Poland
Pierdaniele Giaretta	University of Padova, Italy
Michael Grüninger	University of Toronto, Canada
Nicola Guarino	ISTC-CNR Trento, Italy
Giancarlo Guizzardi	Free University of Bozen-Bolzano, Italy
Torsten Hahmann	University of Maine, USA
Janna Hastings	Otto-von-Guericke University of Magdeburg, Germany
Robert Hoehndorf	King Abdullah University of Science and Technology, Saudi Arabia
Krzysztof Janowicz	University of California, Santa Barbara, USA
Ludger Jansen	University of Rostock, Germany
Gilles Kassel	University of Picardie Jules Verne, France
C. Maria Keet	University of Cape Town, South Africa
Oliver Kutz	Free University of Bozen-Bolzano, Italy
Christoph Lange	Fraunhofer Institute for Intelligent Analysis and Information Systems, Germany
Frank Loebe	University of Leipzig, Germany
Claudio Masolo	ISTC-CNR Trento, Italy

Riichiro Mizoguchi	Japan Advanced Institute of Science and Technology, Japan
Till Mossakowski	Otto-von-Guericke University of Magdeburg, Germany
Fabian Neuhaus	Otto-von-Guericke University of Magdeburg, Germany
Leo Obrst	MITRE, USA
Jeffrey Parsons	Memorial University of Newfoundland, Canada
Oscar Pastor Lopez	Universitat Politènica de València, Spain
Simon Schneider	Utrecht University, The Netherlands
Stefan Schlobach	Vrije Universiteit Amsterdam, The Netherlands
Marco Schorlemmer	Artificial Intelligence Research Institute, IIIA, CSIC, Spain
Selja Seppälä	University College Cork, Ireland
Cogan Shimizu	Kansas State University, USA
Gaurav Sinha	Ohio University, USA
Barry Smith	SUNY Buffalo, USA
Gem Stapleton	University of Cambridge, UK
Mari Carmen Suárez-Figueroa	Universidad Politécnica de Madrid, Spain
Nicolas Troquard	Free University of Bozen-Bolzano, Italy
Guido Vetere	IBM, Italy
Laure Vieu	IRIT CNRS, France
Amanda Vizedom	Credit Suisse, USA
Frank Wolter	University of Liverpool, UK
Antoine Zimmermann	École des Mines de Saint-Étienne, France

Additional Reviewers

Enrico Franconi	Free University of Bozen-Bolzano, Italy
Jona Thai	University of Toronto, Canada
Enkhbold Nyamsuren	Open Universiteit, The Netherlands
Mihai Codescu	Otto-von-Guericke University of Magdeburg, Germany

This page intentionally left blank

Contents

IV. Parts and Wholes

V. Methods

I. Foundations

This page intentionally left blank

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200656

An Analysis of the Debate over Structural Universals

Pawel GARBACZ
Department of Philosophy, John Paul II Catholic University of Lublin, Poland

Abstract. The paper outlines a conceptual framework to identify all ontological and logical aspects relevant for the debate over structural universals. The framework allows for a multi-facetted classifications of various accounts of the latter and facilitates their comparison in a systematic way. To show the framework in action I use it to classify all major theoretical positions in this debate.

Keywords. structural universal, mereology, instantiation, metaontology

Introduction

The concept of universals is a key explanatory device in philosophy and in applied ontology. The distinction between universals and individuals is one of the most fundamental distinctions one can make in any discourse or over any domain. Evoking universals one may want to explain why two individual entities are similar, why we observe certain regularities among different, although similar individuals, etc. Universals can also play more specific theoretical roles, e.g., they may be possible worlds in ersatz modal realism. Thus universals come in handy, but at a price since they seem to be different in many respects from the individual entities and these differences may give rise to various paradoxes or antinomies. One of the less known objections questions their theoretical value on the basis of the following argument, which was probably first explicitly stated in [1]:

1. If there are universals, then, given what we know about the universe, it is possible that *all* universals are structural, i.e., that all universals have parts[1] or some kind of structure built out of other universals.
2. The concept of structural universals is not intelligible.

[1] focusses mainly on the second premise arguing that one cannot provide a consistent and meaningful account of the mereological structure of universals. So, *prima facie*, it casts a shadow of doubt on structural universals. Still, the argument can be seen as a conditional refutation of all kinds of universals, where the condition in question describes a possibility of the infinite complexity, i.e., the possibility that every universal is composed of other universals. Given this provision, the argument builds a case against the concept of universals questioning the intelligibility of just one kind of universals, namely, structural universals. Structural universals are claimed to be faulty devices because one

[1] I assume that 'part' means always 'proper part' throughout this paper.

cannot rationally account for their mereological structure, i.e., one cannot explain what it means that one universal is part of another. And this is all what structural universals are about. Therefore, if the universe turns out to be infinitely complex and all universals are structural, then no universal is intelligible and as such cannot serve any explanatory role. As a matter of fact, this particular type of argument, if conclusive, effectively undermines all views that posit any type of universals other than nominalistic (i.e. linguistic) constructs. In other words, both realism and conceptualism about universals, of any kind, must go.

Needless to say such sweeping objection gave rise to a number of accounts that attempted to salvage universals as *bona fide* theoretical devices. These attempts usually aim to rebut Lewis's criticism by explaining what it means that one universal is part of another and by explaining away the issues he raised in [1].

A careful analysis of the ensuing discussion reveals that the problem is, so to speak, multi-facetted, i.e., that it involves a number of more fundamental issues. In this paper I attempt to identify these aspects by means of a conceptual framework in which all these different accounts of structural universals can be accommodated so that one can classify and compare them. In particular my goal is to mine the elementary points of contention therebetween so that it will be more transparent what it is at stake in this debate.

Let me start the analysis with fleshing out the reasons because of which [1] questions the intelligibility of structural universals.[2]

1. Problems

A *structural universal* is a universal that is complex, i.e., it is somehow composed of other universals.[3] The controversy over structural universals concerns this very qualification 'somehow': what does it actually mean that a universal is composed out of other universals.

As usual, it is easier to say what it does *not* mean. [2] states upfront that structural universals are *not* conjunctive universals, where the latter are defined as below:

Definition 1 (Conjunctive Universals) *Universal* U *is* conjunctive *iff given that* U *is composed of universal* V *(*$U \neq V$*), then for every individual x, if x instantiates* U*, then x instantiates* V *as well.[2]*

So, for example, universal RED AND ROUND is conjunctive provided that we assume that it is composed of universal RED and universal ROUND.[4]

Given that structural universals don't boil down to conjunctive universals, what are they then? Consider an ethane (C_2H_6) molecule, i.e., an individual that is composed of six hydrogen (individual) atoms and two (individual) carbon atoms as depicted in fig. 1. If we grant that universals exist, this molecule instantiates one of them: ETHANE. But then also all its hydrogen atoms and the two carbon atoms instantiate some universals,

[2]I would like to thank the anonymous reviewers of this paper whose comments allowed me to improve its quality.

[3]For the sake of necessary simplifications this paper ignores the temporal aspect of parthood or any akin relationship, both in the domain of universals and in the domain of individuals.

[4]As usual, some would disagree: the theory outlined in [3] describes all structural universals as conjunctive universals.

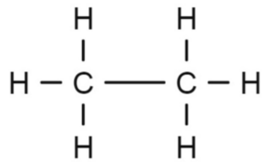


Figure 1. Ethane

scilicet, HYDROGEN and CARBON, respectively. Now each (individual) hydrogen atom is part of the ethane molecule. Is HYDROGEN part of METHANE as well? Obviously, there is some relation, at least akin to parthood, which relates these two, but if it is a species of parthood, then it is a rather peculiar one.

First, different structural universals may have the same universals as parts. For example, ETHANE has two conformational isomers – depicted in fig. 2:

1. STAGGERED ETHANE
2. ECLIPSED ETHANE

which have the same universals as parts and which are different due to the different relative positions of those parts. Thus, this type of parthood is not extensional.[5]

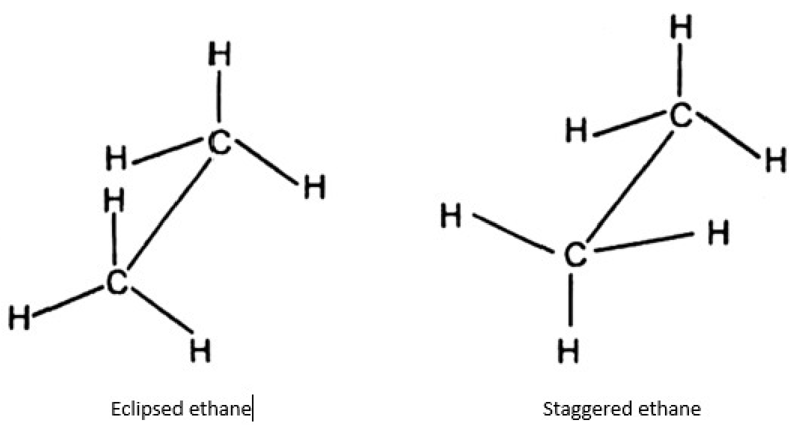


Figure 2. Ethane conformations

Secondly, compare ETHANE to METHANE (CH_4).

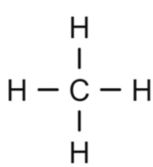


Figure 3. Methane

[5]Relation R is *extensional* if and only if it is the case that $\forall x,y,z\ \{[R(z,x) \equiv R(z,y)] \rightarrow x = y\}$.

One may wish to say METHANE, in the same way as ETHANE, has HYDROGEN and CARBON universals as parts, i.e., that HYDROGEN and CARBON are parts of both ETHANE and METHANE. At the same time ETHANE is different from METHANE, but, it seems to me, in a different way than STAGGERED ETHANE is different from ECLIPSED ETHANE. They are different not because their parts are arranged in different ways but because they have different parts. One may say that ETHANE is different from METHANE because (i) CARBON occurs twice in the former and only once in the latter and (ii) HYDROGEN occurs six times in the former and four times in the latter. Or one may say that the difference between ETHANE to METHANE is due to the fact that (i) the former has two CARBON parts and the latter has just one and that (ii) the former has six HYDROGEN parts and the latter has four of them. In other words, CARBON is part of ETHANE twice over and is part of METHANE only once (and similarly for HYDROGEN). Now this problem may be seen as more substantial than the first one: what can it mean for something to have a part twice over and how is this different from having it only once or from having it, say, three times over?

Thirdly, note that if structural universals do have parts, then we may need to recognise a number of unexpected mereological consequences. For example, if HYDROGEN is part both of METHANE and ETHANE, then the universals of METHANE and ETHANE overlap although none of their instances do.

2. Solutions

These issues, well at least the first two, were originally stated in [1], where D. Lewis also outlines a number of possible solutions. His outlines are rather sketchy and as such require a certain amount of rational reconstruction in the sense of R. Carnap.

Let me start with the so-called pictorial theory of structural universals. The main tenet of this theory is that individuals are isomorphic (with respect to the relation of parthood) to the universals they instantiate in the sense due to which representations like fig. 1 or fig. 3, construed as "pictures", illustrate the mereological structure of both the universal and its instances. As a matter of fact, the details of the pictorial theory of structural universals may require a more convoluted description:

Solution 1 (Pictorial Theory) *If individual x instantiates structural universal* U, *then there is a set X of x's (proper) parts such that*

1. *x is the mereological sum of X;*
2. *there exists a surjective map f from X to the multiset of* U*'s parts such that*
   (a) *for all $y,z \in X$ if y is part of z, then $f(y)$ is part of $f(z)$;*
   (b) *f corresponds to the relation of instantiation, i.e., for all $x \in X$, $f(x)$ is the universal x instantiates.*[6]

Clause 1, or rather the whole idea of taking some, i.e. not necessarily all, parts of individuals into account when we consider the morphism between a universal and its instances, is justified by the need of being able to represent a broad range of universals and

[6] For the sake of simplicity, I assume here that the relation of instantiation is a function, i.e., that no individual can instantiate more than one universal.

all kinds of morphisms. Consider the universal of METHANE. *Prima facie* it contains universals of HYDROGEN and CARBON atoms but it does not contain universals corresponding to their subatomic parts, e.g., to electrons or protons. If this, somewhat superficial, interpretation is right, then the universal of METHANE specifies, so to speak, its parts just down to the level of atoms and the structural similarity between an individual molecule of methane and METHANE does not range over all parts of the former, i.e., an instance of a universal may have parts that are not mapped onto any part of the universal. The clause in question is to allow for such cases – as it allows also for all cases where all parts of an individual are mapped onto its universal's parts. In other words, this clause is to make room for structures being homo- or isomorphic down to a certain level of granularity.

Now Pictorial Theory may occur in two versions:

1. weak: where there exists no bijective map that satisfies clause 2 in solution 1;
2. strong: where there exists a bijective map that satisfies clause 2 in solution 1.

Given a weak version one may infer, along the lines outlined in the previous section, all three issues mentioned there:

1. parthood among universals is not extensional;
2. one universal can be part of another multiple times;
3. many structural universals overlap.

That is to say, one may infer these consequences within the context of the weak version the pictorial theory, i.e., the arguments present previously are consistent with this version.

I reckon that the theory developed in [4] could be classified as an example of such account although the main body of this approach is focused on how to represent mereological structure of individuals.

Given the strong version of the pictorial theory corollary 2 drops out because in this version there are as many parts of a structural universal as there are parts of its instances and no such part need to be part of the universal more than once. An example of such approach is described in [5], where each universal that is part of a structural universal is claimed to be a universal that is a *subtype* of one common or global universal. The idea is that each occurrence of a universal in a structural universal is actually a universal, which is a kind of local species of one common genus. This account would, for instance, have it that (i) there exists the HYDROGEN universal, (ii) the METHANE structural universal would have as its parts six universals that are subtypes of HYDROGEN, and (iii) each individual hydrogen atom in a methane molecule is an instance of one of these six universals. In this particular exemplification of the strong pictorial theory even corollary 3 no longer holds because [5] has it that each structural universal has its "own", local subtypes, so ETHANE and METHANE have no part in common. The non-extensionality of parthood still remains an issue here given the existence of conformational isomers. But the proponent of the strong theory will have to address the more serious objection, mainly the problem of egregious proliferation of universals.

Another type of solutions questions the claim that structural universals have parts. Now it is argued that a non-mereological account of structural universals is needed – a version of such solution was discussed in [1] as the magical theory. Such solutions take it for granted that if the notion of structural universal is to be different from the more general notion of universal *simpliciter*, there should exist some *differentia specifica*, which would separate structural from non-structural universals. This difference needs to be rendered in terms of a certain relation between structural universals and universals so that a universal is structural if there are one or more universals that bear this relation to it. For example, [1] tentatively uses the term ”involvement”, i.e., a universal is structural if it involves other universals. If we subscribe to a non-mereological account of this type, we need to be able to identify a certain relation by means of which structural universals may be defined. Let R be the appropriate relation. This type of solution, which incidentally generalises the pictorial theory, would require that we adapt clause 2 in solution 1 as follows:

Solution 2 (Magical Theory) *If individual x instantiates structural universal* U*, then there is a set X of x's (proper) parts such that*

1. *x is the mereological sum of X;*
2. *there exists a surjective map f from X to the multiset of* U*'s* R*-chunks such that*
   (a) *for all $y, z \in X$ if y is part of z, then $\mathrm{R}(f(y), f(z))$;*
   (b) *f corresponds to the relation of instantiation, i.e., for all $x \in X$, $f(x)$ is the universal x instantiates.*[7]

As before this account may come in two versions: weak and strong. Given the weak version of the magical theory the three aforementioned observations still hold:

1. R is not extensional;
2. one universal can be R-chunk of another multiple times;
3. many structural universals R-overlap.

However this time given a proper account of R one can probably dismiss 1 and 3 as implicit features of R. Still 2 should be seen as an issue because no matter what kind of relation is at stake if you think about relations in the usual sense, it makes no sense to say that universal U_1 is R-ed to universal U_2 multiple times. But, as in the case of strong pictorial theory, 2 may drop out for the strong version of the magical theory if, for example, R-chunks of universals are construed as in [5].

Thirdly, one may raise a doubt whether parts of universals are themselves universals. Given the conceptual framework adopted in this section this amounts to dropping clause 2b in solution 1. The resulting theory may be expressed in the homogeneous or heterogeneous sense: one may require that the mapping function never coincides with the relation of instantiation (see assumption 3) or require that it coincides with it only sometimes (assumption 4):

Solution 3 (Homogeneous Amphibian Theory) *If individual x instantiates structural universal* U*, then there is a set X of x's (proper) parts such that*

1. *x is the mereological sum of X;*
2. *there exists a surjective map f from X to the multiset of* U*'s parts such that*
   (a) *for all $y, z \in X$ if y is part of z, then $f(y)$ is part of $f(z)$;*
   (b) *for all $x \in X$, $f(x)$ is not the universal x instantiates.*

[7] The notion of chunk is a generalisation of the notion of part: x is an R-chunk of y if $\mathrm{R}(x, y)$. Conversely, one may define R-lumps: x is an R-lump of y if $\mathrm{R}(y, x)$.

Solution 4 (Heterogeneous Amphibian Theory) *If individual x instantiates structural universal* U*, then there is a set X of x's (proper) parts such that*

1. *x is the mereological sum of X;*
2. *there exists a surjective map f from X to the multiset of* U*'s parts such that*
   (a) *for all* $y, z \in X$ *if y is part of z, then* $f(y)$ *is part of* $f(z)$*;*
   (b) *for some* $x \in X$*,* $f(x)$ *is not the universal x instantiates and for some* $x \in X$*,* $f(x)$ *is the universal x instantiates.*

In both versions the amphibian theory has it that some parts of universals are not universals themselves. So on this approach four HYDROGEN entities in ETHANE universal are not universals themselves. Obviously, the main problem now is what these entities actually are. One may *say* that they are occurrences of universals, e.g., four items marked by H in fig. 3 are four different occurrences of the HYDROGEN universal. Again it is easier to say what these occurrences are not: i.e., occurrences of universals are neither individuals nor universals. [1]'s dismissively dubs them "amphibians" because he argues that they do not fit the standard ontological landscape where all entities are disjointly and exhaustively divided into individuals and universals. Also, given solutions like [5], one may ask how a "local universal" is different from an occurrence of the global universal. In any case I am not aware of any specific example of this kind of solution and that whether the three issues mentioned in the previous section apply now as well depends on its details.

Finally I should note that [1] mentions also the linguistic account of structural universals, where structural universals are (unspecified) set-theoretic constructions of words, or more specifically speaking, of predicates, which are simple, i.e., non-structural, universals. These constructions are said to follow the recursion patterns we know from the model theory, e.g., the relation of instantiation becomes now the relation of satisfaction. [1] does not provide us with the details of this account, e.g., we don't know whether words or predicates he mentions are word-tokens or word-types and we are not informed about the type(s) of the set-theoretical constructions in question, so we don't know whether simple universals can be construed as parts of the structural universals.[8] As a result, it is not possible to unambiguously render this solution in terms similar to the other three solutions.

3. Facets of Structural Universals

All parties to the debate on structural universals seem to presuppose a number of relatively obvious assumptions:

1. There are two disjoint domains of entities: individuals and universals, where the former instantiate the latter.

[8] On the other hand, we do know why [1] does not accept this solution: it requires that all structural universals are ultimately grounded in simple universals and this excludes the possibility of the infinite complexity.

2. Universals are posited, rather than discovered, to serve some theoretical goal(s), for example they can be deployed to explain various kinds of similarity between individuals, i.e., two individuals are similar to one another in a given respect if they both instantiate the universal relevant for this respect, e.g., they are similar as chemical molecules if they instantiate the same chemical formula.
3. The relation of parthood is primarily defined on individuals, i.e., there are certain mereological facts that can be asserted about them. These assertions can be deployed to explain *some* similarities among individuals, i.e., one may explain that two individuals are similar pointing out to the fact they have isomorphic or homomorphic mereological structures.
4. The relation of parthood or rather a counterpart thereof can be derivatively defined on universals, well at least on some of them. Namely, we saw in section 1 that there are universals that are posited to explain mereological similarities between their instances, i.e., the fact that one individual is mereologically similar to another individual is ontologically grounded in the universals these two individuals instantiate. Such similarities are not just accidental facts but that they hold between the individuals of necessity and that this kind of necessity is somehow grounded in the respective universals. The interplay between mereology and necessity can be described as follows: if individual x is part of individual y and if this mereological fact has to do with the universals x and y instantiate, say, respectively, V and U, then necessarily, every instance of V is part of some instance of U.[9]
5. If this account adequately represents the common assumptions of the debate in question, we are now in the position to provide a definition of structural universals, Here U is defined as a *structural universal* if and only if there exists (at least) one universal V such that the following two principles are satisfied by U and V:

   Principle 1 *It is possible that some instance of universal* U *has a part that is an instance of* V.[10]

   Principle 2 *If some instance of universal* U *has a part that is an instance of universal V, then necessarily, every instance of* U *has a part that is an instance of V*.[11]

   If universals U and V satisfy these principles, one can say that V is part, or better chunk, of U.

Having the scope of agreement outlined let me elaborate the contestable aspects of the debate. The previous section showed that the issues in question go well beyond the initial Lewis' worries about the extensionality of parthood. In fact, it involves a system of more basic issues. Let me now untangle them by identifying and elaborating on the main aspects involved:

[9] This interpretation allows for mereological similarities between individuals that are not ontologically grounded in the universals these individuals instantiate.

[10] The modal qualifier is to allow for the case of uninstantiated structural universals.

[11] This principle is a slightly modified version of the CO-INT principle from [2].

A. General structurality: this aspect concerns the issue whether the domain of universals can be seen as a structure, i.e., whether universals, including structural universals, are related to one another by some relation R that is relevant for explaining principle 2.

- All accounts of structural universals I am aware of imply that structural universals are related to one another in some way, but generally speaking it is possible to hold an extreme view where principles of the form principle 2 are explained without a reference to any relationships between universals. One may hold a sort of brutalism (with respect to structural universals) according to which principle 2 reports just brute facts.[12] Alternatively, one may refer to the specific natures, i.e., intrinsic properties, of these universals. Be it as may, all other facets mentioned below are meaningful only if the general structurality question is answered in the positive.
- A structure in which a structural universal U occurs will be called a *universal structure*. Similarly, the mereological structure of an individual will be called *individual structure*.

B. Structure multiplicity: this concerns the issue whether a structural universal has a single structure or multiple structures.

- Most accounts of structural universals presuppose that a structural universal has the unique structure, be it mereological or otherwise, which is referred to when principle 2 is explained. However [7] points out to a theoretical possibility where a structural universal is described by means of two relations to other universals: (non-extensional) parthood and composition.[13] Also [8] mentions two kind of relations: "slothood" ('. . . is a slot in (universal) . . . ') and slot occupancy ('(universal) . . . occupies (slot) . . . ').
- I assume that each structure in which a structural universal occurs is defined by one relation. Thus, multiple universal structures mean multiple relations R1, R2, . . . etc.

C. Structure type: this concerns the issue whether R is the relation of parthood or not.

- Following [1] I assume that the relation of parthood is extensional, so all non-extensional relations, even if conceptually close to parthood, will be classified here as other relations. I take this assumption as a terminological convention without any ontological or logical commitment.

D. Structural universal's role in structure: this concerns the issue whether structural universals play the role of R-lumps in these structures or play the role of R-chunks or neither.

- This aspect is to make room for all kinds of general non-mereological solutions in which structural universals play the role that generalises the mereological role of the part, or the role that generalises the role of the whole or neither.

[12] I have in mind here a theory similar, *mutatis mutandis*, to the theory of mereological composition exposed in [6].

[13] I assume here the interpretation of [7] given in [2].

- I am not aware of any account where structural universals play the role of R-chunks, but [3] implies that structural universals play neither the role of R-chunks nor R-lumps.

E. Structural similarity: this concerns the issue whether universal structures are structurally similar to the individual structures of their instances or not.

- Structural similarity is given by a map f from the set of instance's parts to the set of universal's R-chunks such that if individual x is part of individual y, then $f(x)$ is R-chunk of $f(y)$. I take it that this map is a genuine ontological relation (or a natural relation in the sense of [9]), e.g., the relation of instantiation, and not just a set (of ordered couples).
- Again all known accounts of structural universals imply the affirmative answer to this question, i.e., they endorse the existence of some similarity map, but a view that universal structures are not similar to instance structures probably can be worked out. For example, if we interpret formal parts in the sense of [10] as such parts (of individuals) that either (i) are individuals that do not instantiate any universals or (ii) are individuals that instantiate universals that are not parts of any other universals, then the individual structures will not be similar to the respective universal structures. As before, all facets mentioned below are meaningful only if the structural similarity question is answered in the positive.
- If universal structures are structurally similar to the individual structures, then two other questions are relevant:
  (a) Similarity ontological interpretation: this concerns the issue whether the similarity map coincides with the instantiation relation.
  (b) Similarity formal interpretations:
    i. Similarity scope: this concerns the issue whether the similarity map is a partial or (total) function – see the discussion of clause 1 of solution 1 above.
    ii. Similarity formal properties:
      A. one aspect is whether the similarity map is surjective or not;
      B. another aspect is whether the similarity map is injective or not.

The aspects B., C., D., and E. are orthogonal to one another, i.e., an answer to any of these does not imply or exclude any answer to all others. Similarly, the aspects: E.a, E.b and the aspects: E.(b)iiA, E.(b)iiB, are, respectively, independent from one another. The visual summary of all these facets is shown in fig. 4.

4. Solutions Classified

Having the above conceptual framework in place we can now classify the major accounts of structural universals. I show in table 1 an attempt towards this end – obviously a rough and ready one because the proper classification would require a detailed conceptual analysis of each of these theories – which most likely would reveal alternative interpretations for at least some of them – and this task goes well beyond the scope of this paper. For the sake of illustration of possible intricacies let me note that [11]'s account seems to as-

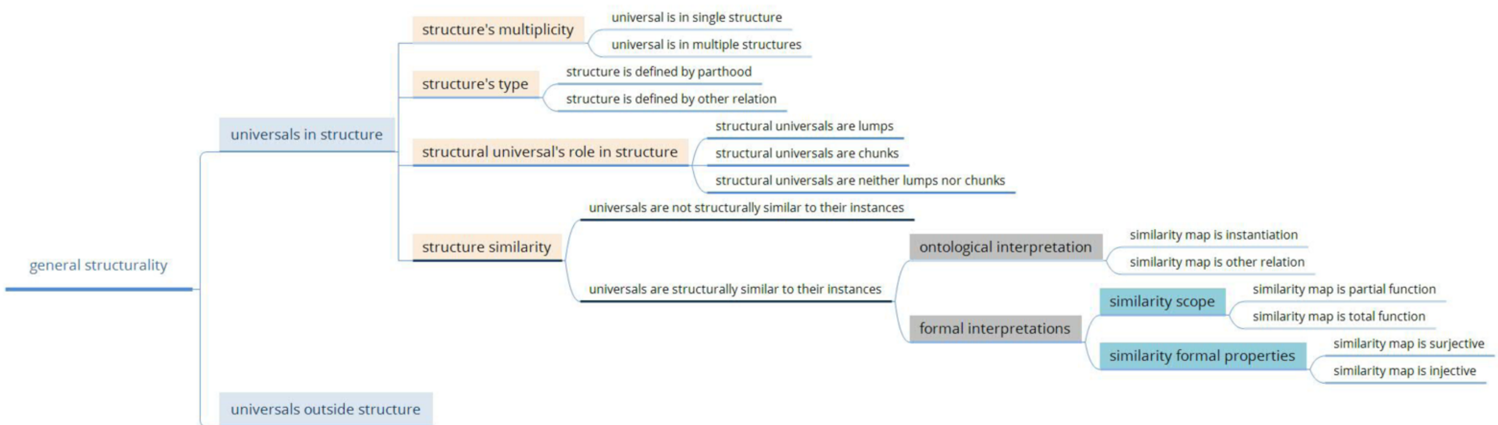


Figure 4. Main facets of the debate over structural universals

Source	**Structure Multiplicity**	**Structure Type**	**Universal Role**	**Similarity Interpretation**	**Similarity Formal Properties**
[1] (Pictorial)	Single	Parthood	Lump	Instantiation	Surjection and Non-Injection
[4]	Single	Parthood	Lump	Instantiation	Surjection and Non-Injection
[5]	Single	Parthood	Lump	Instantiation	Surjection and Injection
[1] (Magical)	Single	Other Relation	Lump	Instantiation	Surjection and Non-Injection
[11]	Single	Non-extensional Parthood	Lump	Instantiation	Surjection and Injection
[3]	Single	Constitution	Neither	Instantiation	Surjection and Non-Injection
[7]	Multiple	Non-extensional Parthood Composition	Lump	Instantiation	Surjection and Non-Injection
[8]	Multiple	slothood Slot occupancy	Lump	Instantiation	Non-Surjection and Non-Injection
[1] (Amphibian)	Single	Parthood	Lump	Other Relation	Surjection and Non-Injection
[13]	Single	Parthood	Lump	Instantiation	Non-Surjection and Non-Injection
[14]	Single	Parthood	Lump	Instantiation	Non-Surjection and Non-Injection

Table 1. Classification of theories of structural universals

sume a rather subtle distinction between universals that are state-of-affairs types ("…is F") and universals that are components of state-of-affairs types ("F"), which distinction may involve two kinds of instantiation relationship or may hide the fact that the former universals are more like amphibians than like universals proper.

The classification does not include two of the aforementioned aspects:

1. general structurality: all theories in scope assume that structural universals have structures of some kind.
2. similarity scope: all theories in scope do not explicitly specify the scope, but implicitly they do assume partial scope for the examples they mention.

Let me note that the framework is capable to accommodate not just the existing solutions but it also allows for possible, yet-not-formulated, accounts of structural universals. On the other hand, it does not range over all solutions for the issues raised in [1], in particular all theories that "explain away" structural universals, like [12], are outside its scope.

5. Conclusions

The above analysis of the debate over structural universals showed a number of dependent and independent issues (of the ontological and formal provenance) that are relevant when one needs to formulate an adequate account of structural universals. These aspects go well beyond the initial problem of the extensionality of parthood raised by D. Lewis. All of them seem to be pertinent for the comprehensive evaluation of any theory of universals. Having them in place we can classify and compare different solutions in a more systematic way, although obviously not every detail of every account is captured in the framework. Finally the identified aspects may be part of a requirement specification for a logical framework that is capable to represent structural universals of any theoretical provenance.

References

[1] Lewis D. Against structural universals. Australasian Journal of Philosophy. 1986;64(1):25–46.
[2] Fisher ARJ. Structural universals. Philosophy Compass. 2018;13(10):e12518.
[3] Bigelow J, Pargetter R. A theory of structural universals. Australasian Journal of Philosophy. 1987;67(1):1–11.
[4] Masolo C, Vieu L. Graph-Based Approaches to Structural Universals and Complex States of Affairs. In: Formal Ontology in Information Systems - Proceedings of the 10th International Conference, FOIS 2018, Cape Town, South Africa, 19-21 September 2018; 2018. p. 69–82.
[5] Davis WA. On Occurrences of Types in Types. Australasian Journal of Philosophy. 2014;92(2):349–363.
[6] Marksonian N. Brutal Composition. Philosophical Studies. 1998;92:211–49.
[7] Hawley K. Mereology, Modality and Magic. Australasian Journal of Philosophy. 2010;88(1):117–133.

[8] Bennett K. Having a Part Twice Over. Australasian Journal of Philosophy. 2013;91(1):83–103.

[9] Lewis D. New Work For a Theory of Universals. Australasian Journal of Philosophy. 1983;61:343–377.

[10] Koslicki K. The structure of objects. Oxford University Press; 2008.

[11] Armstrong DM. A world of states of affairs. Cambridge University Press; 2008.

[12] Mormann T. On the mereological structure of complex states of affairs. Synthese. 2012;187(2):403–418.

[13] Kalhat J. Structural Universals and the Principle of Uniqueness of Composition. Grazer Philosophische Studien. 2008;76:57–77.

[14] McFarland A. Causal powers and isomeric chemical kinds. Synthese. 2018 Apr;195(4):1441–1457.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200657

Foundations for Ontology of Persistence: Beyond Talk of Temporal Parts

Fumiaki TOYOSHIMA[1]
GRIIS, Université de Sherbrooke, Sherbrooke, Quebec, Canada

Abstract. Persistence is about how things behave across time. It is generally discussed in terms of endurantism (three-dimensionalism) and perdurantism (four-dimensionalism). Despite the relevance of persistence to ontological modeling, however, there is no clear consensus over how to characterize precisely those two theories of persistence. This paper takes the initial steps towards a foundation for ontology of persistence. In particular, I examine by employing recent findings from philosophy of persistence how some major upper ontologies conceptualize endurantism and perdurantism. My resulting modest suggestion is that formal-ontological discussion on persistence should be updated by expanding its perspective beyond the topic of whether objects have proper temporal parts or not.

Keywords. persistence, temporal part, time, spacetime, upper ontology

1. Introduction

Persistence is about how things behave across time. It is usually discussed in terms of two contrasting theories of persistence: endurantism (aka three-dimensionalism) and perdurantism (aka four-dimensionalism). To see the relevance of persistence to ontological modeling, it will suffice to consider its vital role in upper ontologies. An upper ontology (aka foundational ontology), by nature, "deals with general domain-independent categories only" and "has been built and motivated by the upfront and explicit choice of its core principles" [1, p. 3]. A classical example of choices of upper ontologies is an ontological choice [2]: a choice as to whether a certain ontological category or relation is adopted. A relatively widespread ontological choice is the categorical distinction between continuants (objects) and occurrents (events, processes). There is also a growing acknowledgement of meta-ontological choices [3] of upper ontologies: choices that characterize categories and relations foundationally. As de Cesare et al. [3] say, a choice between endurantism and perdurantism can be seen as a meta-ontological choice because it determines the nature of continuants (and occurrents). As a matter of fact, the terms 'endurant' and 'continuant' (resp. 'perdurant' and 'occurrent') are often employed interchangeably in formal ontology.

It is nonetheless controversial what endurantism and perdurantism are supposed to be. As we will see below, they would seem to be characterized differently in different

[1] E-mail: toyo.fumming@gmail.com, fumiakit@buffalo.edu. I am indebted to Antony Galton for his highly valuable feedback on this work in its early stage. I appreciate three anonymous reviewers' constructive reviews. I also thank Jean-François Ethier and Christina Khnaisser (in alphabetic order) for some helpful comments on the occasion of the finalization of this paper. Finally, I acknowledge financial support by the SPOR Canadian Data Platform (CIHR).

upper ontologies. Because these two accounts of persistence and a choice between them constitute the crux of ontology of persistence, it will be useful to provide a systematic exploration of hitherto implicit divergent views of persistence in formal ontology.

This paper aims to investigate ontology of persistence from a foundational perspective from which to clarify meta-ontological choices as to persistence. To achieve this goal, I begin by presenting some preliminaries and explaining four existing desiderata for classifying endurantism and perdurantism (Section 2). Then I examine, according to those four criteria, what three major upper ontologies take ontology of persistence to be primarily about: Basic Formal Ontology (BFO), the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE), and General Formal Ontology (GFO) (Section 3). Based on this case-study, I consider which standard for characterizing endurantism and perdurantism would be appropriate in formal ontology by employing recent insights from philosophy of persistence; and I also briefly discuss some implications of my argument on persistence for ontology of time (Section 4). Finally, I conclude the paper with a small suggestion on a foundational ontology of persistence and some brief remarks on future work (Section 5).

2. Ontology of persistence: A general overview

2.1. Preliminaries

The scope of my inquiry will be specified before I present four existing criteria for characterizing endurantism and perdurantism. In the first place, I consider the persistence of continuants, or precisely so-called ordinary material objects: e.g. molecules, people, tables, and planets. Relatedly, I leave aside abstract objects (e.g. sets and numbers) and also “lower-dimensional” objects (e.g. surfaces and boundaries). The term ‘object’ will be henceforth used in this sense unless otherwise stated. Indeed, one may think that occurrents also persist, as the terms ‘occurrent’ and ‘perdurant’ are often used synonymously. However, the usage of the term ‘perdurant’ is based on a conceptual analogy between occurrents and objects as conceived in the perdurantist’s fashion. Thus, persistence is primarily about persistence of continuants (objects) and talk of “persistence of occurrents” would at best convey a perdurantist understanding of objects.[2] In addition, I take it as the very starting point that an object persists if and only if it exists at one time and also exists at another different time. To take one example, my table persists because it existed yesterday, exists today, and will probably exist tomorrow. I also take for granted that some persisting objects gain and lose parts over time: for instance, my table still continues to exist after one of its edges is chipped off.[3]

Presumably, ontology of persistence is a challenging task for at least two reasons. For one thing, discussion on endurantism and perdurantism (or three- and four-dimensionalism) is generally complicated by significantly different usages of the terms ‘endurantism’, ‘perdurantism’, ‘three-dimensionalism’, and ‘four-dimensionalism’.

---

[2] One may think that some version of perdurantism (e.g. [4]) would not distinguish between objects and occurrents (not as sharply as endurantism does or can do, at least) by lumping them together under the heading of something four-dimensional. While this might be a proposed view, it would be still important to clarify how the intuitive difference between objects and occurrents could be represented in this framework

[3] Therefore, mereological essentialism [5] (“Objects have their parts essentially”) is off the table, since it implies that objects cannot gain or lose parts over time.

Insofar as my literature survey is concerned, the latter two terms tend to be used more often than the former two in formal ontology. In philosophy, by contrast, three- (resp. four-) dimensionalism may sometimes mean a specific version of endurantism (resp. perdurantism) or a combination of endurantism (resp. perdurantism) with a particular view of time. For instance, four-dimensionalism is frequently taken, within perdurantism, as the worm theory as compared to the stage theory (see Section 4 for details on these two perdurantist accounts). For the sake of my argumentation, I will employ primarily the terms 'endurantism' and 'perdurantism' to refer loosely to two *most general* theories of persistence, and only secondarily the terms 'three-dimensionalism' and 'four-dimensionalism' (e.g. in a survey of upper ontologies to be given in Section 3).

For another, persistence is closely intertwined with time, since persistence is in nature about how objects behave across time. However, the fundamental nature of time has been scarcely addressed by prominent upper ontologies [6] or by domain ontologies (see e.g. [7]). To simplify the matter, I make two basic presuppositions on time (but see Section 4 for a brief discussion on persistence under other temporal assumptions).[4] First, I postulate the classical (non-relativistic) view of (space and) time or spacetime according to which so-called absolute simultaneity holds.[5] Second, I stipulate the eternalist view of time: the past, the present, and the future exist, or more specifically, past and future times, objects, and occurrents are as real as the present ones [11].[6]

2.2. Theories of persistence: Four desiderata

There are at least four existing desiderata for characterizing the endurantist and perdurantist accounts of persistence [13,14]. These criteria may not exhaust all the aspects of preceding discussion on persistence (see e.g. [15]). As we will see in Section 3, they are still together general and helpful enough to compare some prominent upper ontologies with respect to persistence.

First, one of the most traditional standards for classifying endurantism and perdurantism is (TP) whether objects lack or have proper *t*emporal *p*arts [16,17]:

> (E-TP) Objects lack proper temporal parts.
> (P-TP) Objects have proper temporal parts.[7]

[4] In this paper I remain neutral about the issue of time instants and time intervals, especially of which are ontologically prior to the other (see Galton's [6] detailed discussion, in particular with respect to BFO, DOLCE, and GFO) and also about the debate over time or spacetime between substantivalism and relationalism. As for the former, this might involve the ambiguity of the term 'time' at some points in my explanation (e.g. of the definition of persistence of objects). As for the latter, one may suspect that I am assuming substantivalism because I take for granted the existence of time or spacetime. According to North's [8] reformulation of this substantival/relational dispute, however, both substantivalism and relationalism can be ontologically committed to spatiotemporal structure while disagreeing over whether (facts about) spatiotemporal structure is "grounded" in (facts about) material bodies. Given her proposal, my argument below can be open to both substantivalists and relationalists.

[5] See Maudlin's [9] introductory guide for space, time, and spacetime in classical physics and also Bittner's [10] formal ontology of classical physics.

[6] Another motivation behind this assumption than simplicity is the recent argument [12] that eternalism is entailed even by some approaches to quantum gravity (such as loop quantum gravity and string theory) that suggest the non-fundamentality of spacetime, especially of time.

[7] The notation 'E-*X*' (resp. 'P-*X*'), where *X* is a variable term, means the characterization of *e*ndurantism (resp. *p*erdurantism) in terms of *X*.

As for (P-TP), for instance, my table has its "temporal part at a time t_1" and its "temporal part at a time t_2", just as it has a flat top and a leg as its spatial parts (see Section 4 for detailed discussion on the definition of temporal parts).

Second, an equally well-known desideratum is (PART) whether objects stand only in time-relative *part*hood relations [17-19]:

> (E-PART) Objects stand only in time-relative parthood relations.
> (P-PART) Objects stand in atemporal parthood relations.

To illustrate, the endurantist insists by (E-PART) that the part-whole relation between a leg and my table should be relativized to time such as t_1. The perdurantist counters by (P-PART) that this leg bears the parthood relation to my table in a timeless way.

Third, (SL) *s*patiotemporal *l*ocation (i.e. how objects are located in spacetime) attracts growing attention in contemporary philosophy of persistence [13,14,20-22]:

> (E-SL) Objects are three-dimensional.
> (P-SL) Objects are four-dimensional.

We will see Donnelly's [13] more rigorous formulation of (SL) based on some auxiliary assumptions about spacetime in Section 4. It should be noted that, as I will illustrate with upper ontologies in Section 3, talk of three- and four-dimensionalism does not *ipso facto* entail any commitment to (E-SL) and (P-SL), respectively.

Fourth and finally, a comparatively minor criterion is (EXEM) whether objects *exem*plify properties only in a time-relative way [14,19][8]:

> (E-EXEM) Objects exemplify properties at a time in virtue of the fact that they exemplify those properties in a way that is relative to the time.
> (P-EXEM) Objects exemplify properties at a time in virtue of the fact that their temporal parts at the time exemplify those properties atemporally.[9]

Suppose that my table is white at a time t_1. This is the case, as the endurantist says by (E-EXEM), in virtue of the fact that either my table exemplifies the property of being white that is relativized to t_1 or my table bears the exemplification relation (that is relativized to t_1) towards the property of being white. Contrariwise, the perdurantist ascribes, based on (P-EXEM), the same state of affairs to the fact that the "temporal part at t_1" of my table exemplifies the property of being white in a timeless manner.

3. Persistence in upper ontologies: A case-study

I presented above four standards for specifying endurantism and perdurantism: (TP), (PART), (SL), and (EXEM). Given those desiderata, we will look at those formulations of theories of persistence which are provided by three well-known upper ontologies: BFO, DOLCE, and GFO. The focus is upon persistence of objects, but I will sometimes discuss occurrents to know about a perdurantist conception of objects (see Section 2.1).

---

[8] Throughout this paper I use the term 'property' in its most general sense to refer to characteristics of entities, especially of objects.

[9] I present a simplified version of Suzuki's [14] exposition of (EXEM). In particular, (P-EXEM) can be formulated more accurately as follows: "(i) Necessarily, every object has an instantaneous temporal part at every time at which it exists and (ii) objects exemplify properties at a time in virtue of the fact that their temporal parts at the time exemplify those properties atemporally" [14]. Note that the item (i) therein corresponds to (P-TP).

3.1. Basic Formal Ontology (BFO)

Basic Formal Ontology (BFO) [23,24] discusses three- and four-dimensionalism in connection with BFO:processes (as a subtype of occurrents) as follows:

> One can think of each process as a temporally extended continuum, a spacetime *worm* (…) this view of spacetime worms is distinct from popular four-dimensionalist views according to which objects (…) would themselves be extended in time and would have temporal parts. BFO does indeed embrace a four-dimensionalist perspective; but it combines this with a three-dimensionalist perspective for continuants (…). [23, p. 124]

BFO would take the core of persistence to be about (SL) ("extended in time") and (TP) ("have temporal parts"). At the same time, BFO accepts (E-SL) and (E-TP) since it subscribes to "a three-dimensionalist perspective for continuants". As evidence of this: "The continuant portion of BFO consists of representations of entities that (1) persist, *endure*, or continue to exist through time while maintaining their identity, and (2) *have no temporal parts*" [23, p. 89, emphasis added].

3.2. The Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)

The Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [2,25] delineates three- and four-dimensionalism as follows:

> In general a 3D option claims that objects are: a) extended in a three-dimensional space; b) wholly present at each instant of their life; c) changing entities, in the sense that at different times they can instantiate different properties (indeed, one could say *When I was out in the balcony my hands were colder than now*). On the contrary a four-dimensional perspective states that objects are: a) space-time worms; b) only partially present at each instant; c) changing entities, in the sense that at different phases they can have different properties (*My hands during the time spent out in the balcony, were colder than now*). [25, p. 10]

Using Masolo et al.'s [25] notation, a) and c) would correspond approximately to (SL) and (EXEM), respectively. By comparison, b) would be open to interpretation because it is generally contentious what the phrase 'wholly present' (and 'partially present') generally amounts to [26,27]. At least insofar as four criteria under consideration are concerned, however, b) could be also taken to be about (SL) (see also Section 4 for a discussion on a close relationship between (SL) and the expression 'wholly/partially present'). Therefore, I submit that the DOLCE view of persistence is primarily about (SL) and (EXEM) and DOLCE endorses (E-SL) and (E-EXEM), as is shown by the DOLCE category of *endurant* (continuant). Furthermore, Masolo et al. [25] would seem to describe (PART) as "quite illuminating for our purposes" in discussing the distinction between endurants and perdurants (p. 11, 16). We could therefore recognize the DOLCE commitment to (PART) and (E-PART) as well.

3.3. General Formal Ontology (GFO)

Finally, General Formal Ontology (GFO) [28] discusses persisting objects as follows: "An entity perdures if it persists by having different temporal parts, or stages, at different times, whereas an entity endures if it persists by being wholly present at any time of its existence" [29, p. 180]. GFO would thus seem to think that a general ontology of persistence can be characterized by (TP) ("having different temporal parts, or stages, at different times") and (SL) ("wholly present at any time of its existence").

In striking contrast with BFO and DOLCE, GFO claims that continuants are the creation of the mind and so is their persistence [29,30]. Taken literally, the GFO cognitive view of persistence could imply that there is no persisting object in ontological parlance. It can be also thought however that GFO endorses its original formulation of perdurantism. GFO characterizes continuants in terms of two GFO categories. One is a presential: "an individual which is entirely present at a time-point" [30, p. 309]. The other is GFO:processes, which "have a temporal extension thus cannot be wholly present at a timepoint" [30, p. 310]. GFO asserts principally that, for every continuant *Con*, there exists a GFO:process *Proc(Con)* such that the presentials exhibited by *Con* equal the GFO:process boundaries of *Proc(Con)*. According to this "principle of object-process integration", the alleged change of continuants is explicable in terms of presentials, which are in turn explainable in terms of GFO:processes. Granted that GFO:processes are "perduring" entities, GFO may well be said to provide a perdurantist account of persistence, as evidenced by Herre's [28,30] claim that GFO is the only four-dimensional upper ontology that is used in applications.[10] Table 1 briefly summarizes my survey of BFO, DOLCE, and GFO with respect to persistence.

Table 1. A survey of BFO, DOLCE, and GFO upper ontologies with respect to persistence

Commitment	**BFO**	**DOLCE**	**GFO**
w.r.t. (TP)	X		X
w.r.t. (PART)		X	
w.r.t. (SL)	X	X	X
w.r.t. (EXEM)		X	
w.r.t. its own theory of persistence	Endurantism I.e. (E-TP) and (E-SL)	Endurantism I.e. (E-PART), (E-SL), and (E-EXEM)	1) No persisting object 2) Perdurantism

4. Discussion

4.1. Relevance of spatiotemporal location

According to my survey, BFO, DOLCE, and GFO all share the conviction that (SL) is a suitable standard for characterizing endurantism and perdurantism. To consider this point more carefully, I present Donnelly's [13] formalization of (SL). This requires, first of all, the idea of absolute timeslice.[11] Assuming that spacetime is a four-dimensional manifold

[10] In more detail, Herre [28,30] would seem to employ the term 'three- (resp. four-)dimensionalism' to refer to the thesis that continuants (resp. occurrents) are fundamental to occurrents (resp. continuants). I would prefer to call those two doctrines 'substantialism' and 'processualism', respectively, though [31].

[11] Donnelly [13] assumes Galilean (Neo-Newtonian) spacetime (refer to Maudlin [9] for details). Besides, she claims to presuppose substantivism based on her ontological commitment to spacetime [13, p. 29]. Given

of points that in no way changes or grows, every object is eternally located in this "block spacetime". A spacetime region can be then identified with a non-empty set of spacetime points. Given classical physics, an absolute timeslice is a region which is a maximal set of pairwise (absolutely) simultaneous spacetime points. For the sake of simplicity, we can think of absolute timeslices as time *instants* (rather than time *intervals*) in our common understanding of time.

Donnelly also introduces a primitive relation between an object and a spacetime regions: (exact) occupation or (exact) location. Intuitively, an object exactly occupies a spacetime region just in case the object has precisely the same shape, size, and position as the spacetime region does. Roughly speaking, (E-SL) says that a persisting object exactly occupies multiple three-dimensional spacetime regions. (P-SL) says, by contrast, that a persisting object exactly occupies a single four-dimensional spacetime region. Let it be that a spacetime region r_1 *is included in* a spacetime region r_2 if and only if is r_1 a subset of r_2.[12] Donnelly [13, p. 31] characterizes (E-SL) and (P-SL) as follows:

> (E-SL) Each persisting object x exactly occupies multiple three-dimensional regions r_{xt} where: (i) each r_{xt} is included in some time and (ii) for each time t_0 through which x persists, exactly one of r_{xt} is included in t_0. If there are any non-persisting objects, each of these objects exactly occupies just one region and this region lies within a single timeslice.
>
> (P-SL) Each persisting object x exactly occupies a unique four-dimensional regions r_x which spans x's entire life (in particular, r_x crosses each time through which x persists). If there are any non-persisting objects, each of these objects exactly occupies just one region and this region lies within a single timeslice.

I argue, in line with philosophy of persistence and upper ontologies, that (SL) is an appropriate desideratum for understanding persisting objects (if not fully; see Section 4.3). Characteristically, as Donnelly [13] says, (E-SL) (resp. (P-SL)) might mesh with the endurantist's (resp. perdurantist's) truism that a persisting object is "wholly present" (resp. "partially present") at different times (at which it exists). For, (E-SL) means that a persisting object exactly occupies different (temporally unextended) spacetime regions lying within different timeslices, while (P-SL) means that a persisting object exactly occupies no spacetime region that is "wholly contained" in any timeslice.

4.1.1. Spatiotemporal location in BFO

Donnelly's [13] formulation of (SL) would help to deepen our understanding of (SL) in the three upper ontologies above discussed. First of all, BFO as such would seem to be silent on the relationship between objects and spacetime regions. BFO indeed has the category of *spatiotemporal region* [23, pp. 123-124]. However, it is based on the BFO theory of the dynamism of reality according to which a purely spatial ("SNAP") view of the world should be joint together with a purely spatiotemporal ("SPAN") view of the world [32]. The SNAP and SPAN perspectives on the world represent a series of instantaneous *snap*shots of reality and changes within time *span*s in reality, respectively.

North's [8] reconstruction of the substantival/relational debate (see Footnote 4), however, Donnelly's proposal may be taken to be neutral as to whether substantivalism or relationalism is adopted.

[12] One may prefer to take the inclusion relation between spacetime regions to be primitive by denying Donnelly's [13] identification of spacetime regions as a non-empty set of points. In this sense, her version of (SL) may not be universally accepted, although it would be arguably one of the most explicit formulations of (SL). See also Section 4.1.3 for another limitation.

In BFO, this SNAP/SPAN divide is well reflected in its top-level categorical distinction between continuants and occurrents.[13]

Consequently, a BFO:spatiotemporal region is "an occurrent entity at or in which occurrent entities can be located" [23, p. 123].[14] It is, so to speak, a container within which BFO:processes unfold in the SPAN realm, just as space is a container within which objects and their properties exist in the SNAP realm. Thus, BFO introduces the **occupies_spatiotemporal_region** relation which has as domain a BFO:process or a BFO:process boundary (i.e. an instantaneous temporal boundary of a BFO:process) and as range a BFO:spatiotemporal region.[15] This relation cannot be directly used to formalize a BFO- and (E-SL)-based theory of persistence *of objects*. It could be nevertheless utilized to constrain formally the here introduced "(exact) occupation relation" (**OCU**) between an BFO:object and a BFO:spatiotemporal region as follows:

OCU (x,y) → ∃z,t **participates_in** (x,z,t) ∧ **occupies_spatiotemporal_region** (z,y) ∧ **temporally_projects_onto** (y,t)[16]

Further inquiry along this line is left for the future, such as careful consideration of the conceptual legitimacy of this relation *vis-à-vis* the SNAP/SPAN view of reality.

4.1.2. Spatiotemporal location in DOLCE

Unlike BFO, DOLCE does not explicitly have the category of spacetime region. It is nonetheless natural to think that the DOLCE specification of spatiotemporal locationality of objects would follow its existing "quality-based" treatment of their spatial and temporal locations: "In our ontology, space and time locations are considered as individual qualities like colors, weights, etc." [25, p. 18]. The DOLCE theory of qualities introduces several technical terms: '(individual) quality', 'quality type', 'quality space', and 'quale' [2,25]. Qualities (e.g. the color of this rose) are property particulars that depend specifically on particular entities. Quality types are "partitioners" of qualities: for instance, the color quality type provides the whole color spectrum. Quality spaces (e.g. the "color space") are "classifiers" of qualities of the same quality type, or more precisely mereological sums of all the DOLCE:quality regions (which are a subtype of DOLCE:abstract entities) related to a certain quality type. Qualia (e.g. a particular shade of red) are "values" that qualities have in virtue of their position within a certain quality space. As DOLCE goes, spatial (resp. temporal) locations are qualities that belong to the space (resp. time) quality type and that have as their qualia DOLCE:space (resp. temporal) regions (which are a subtype of DOLCE:quality regions) in the corresponding quality space, namely in the geometric (resp. temporal) space.

In this direction, DOLCE can analyze spatiotemporal locations as qualities that belong to the "spacetime quality type" and that have as their qualia spacetime regions in the "four-dimensional coordinated quality space", i.e. a mereological sum of all the

---

[13] While retaining its foremost continuant/occurrent distinction, the latest version of BFO [24] would seem to be somewhat distant from this SNAP/SPAN worldview, though.

[14] See Bittner's [10] criticism of the BFO occurrent conception of spacetime regions and also Galton's [6] related critical examination of the BFO category of *temporal region*.

[15] See Axiom (15) found in: https://standards.iso.org/iso-iec/21838/-2/ed-1/en/pdf/spatiotemporal.pdf.

[16] For details on the **participates_in** and **temporally_projects_onto** relations in BFO, see: https://standards.iso.org/iso-iec/21838/-2/ed-1/en/pdf/participation.pdf and https://standards.iso.org/iso-iec/21838/-2/ed-1/en/pdf/spatiotemporal.pdf, respectively.

spacetime-related quality regions. A worry could nevertheless arise as to how well this approach represents the idea of exact occupation that we currently seek. Certainly, DOLCE aims "to capture the intuitive and cognitive bias underlying commonsense" [2, p. 279] and the modeling of spacetime regions within DOLCE should be understood "from the mesoscopic and conceptual level" [2, p. 280] of reality.[17] Still, spacetime (as well as space and time) and persistence are so fundamental to upper ontologies in general that the DOLCE quality-based view of spatiotemporal location (and perhaps also of spatial and temporal locations) might merit further consideration.

4.1.3. Spatiotemporal location in GFO

We will look finally at GFO. While the GFO theories of time [29] and space [34,35] have been recently formalized, the GFO underlying view of spacetime may hinder us from considering persistence in GFO by means of Donnelly's [13] formulation of (SL). GFO conceptualizes space and time as abstractions of the continuum that can be accessed through introspection, hence "phenomenal space" and "phenomenal time". Notably: "space and time cannot be conflated into a homogeneous, four-dimensional system" [34, p. 55]. This contradicts the temporal assumption on which Donnelly's proposal is based: spacetime is a four-dimensional manifold of points. As GFO says, space and time are nonetheless naturally integrated by the object-process integration principle (see Section 3.3). I will return to the topic of persistence in GFO in Section 4.3.

4.2. Irrelevance of the mere possession of temporal parts

Let us turn to the (TP) criterion for persistence, on which BFO and GFO would agree. To this date, (TP) remains a fairly common characterization of theories of persistence in philosophy [36] and in formal ontology [31,37]. There is however a growing recognition that (TP) may be an unsuitable standard for understanding endurantism and perdurantism [13,14]. Here I present Suzuki's [14] simple argument for this thesis.[18] First of all, one of the most widespread definitions of temporal parts is arguably Sider's [17] based on the ternary time-indexed parthood relation[19]:

> *x* is an instantaneous temporal part of *y* at time instant $t =_{\text{def.}}$ (1) *x* exists at, but only at, *t*; (2) *x* is part of y at *t*; and (3) *x* overlaps at *t* everything that is part of *y* at *t*. [17, p. 59]

For instance, a temporal part of my table when it is white is such that (1) it exists at and only at that time; (2) it is part of my table at that time; and (3) for every part *z* of my table at that time, there is something that is part of the temporal part at that time and that is part of *z* at that time. Using this definition of temporal parts, Sider explains perdurantism as follows:

> (P-TP) Necessarily, every object has an instantaneous temporal part at every time at which it exists.

[17] There may be nonetheless a more nuanced approach to spatiotemporal location in compliance with DOLCE. For a pointer to this line of research, see Brodaric et al.'s [33] DOLCE-based formalization of spatial location that is not necessarily based on its quality-based treatment above explained.

[18] Donnelly [13] develops a more complex argument by introducing her notion of "temporal segment".

[19] Refer to Donnelly [13] and Olson [19] for other alternative definitions of temporal parts.

It is rather obscure, by contrast, how to offer a (TP)-based formulation of endurantism, i.e. (E-TP), so that (E-TP) can be reasonably compared to (P-TP). A naïve attempt to provide (E-TP) is to deny (P-TP) as follows:

> (E-TP*) Possibly, some objects do not have any instantaneous temporal part at some time at which they exist.

Not unnaturally, (E-TP*) would be explanatorily impotent with respect to an accurate understanding of endurantism. A more promising explanation of (E-TP) would be the following, relatively popular idea concerning endurantism:

> (E-TP**) Objects do not have any temporal parts (at any time at which they exist).

(E-TP**) would be nonetheless untenable either, because the endurantist theory of *persistence of objects* (rather than of *the structure of objects*) would cohere with the existence of temporal parts of objects. To take Sider's [17, p. 64] example, the endurantist would be ontologically committed to (the possibility of) instantaneous objects (objects that exist only at one time instant), which turn out to be (improper) temporal parts given his definition of temporal parts presented above. Sider [17, pp. 64-65] further argues, on some auxiliary assumptions that the endurantist can accept, that persisting objects can have (proper) temporal parts. This means that endurantism can be ontologically committed to temporal parts of persisting objects as well.

Furthermore, it is unclear what perduring objects are like according to (P-TP), as says Suzuki [14]. More specifically, it would be hardly understandable why perdurantism compels objects to have so many instantaneous temporal parts during the whole course of their lives. Briefly, (TP) is inappropriate for specifying the endurantist and perdurantist theories of persistence. As Donnelly [13] points out, (TP) may be about what "short-lived" objects there are, but not about how objects persist.

4.3. Possible relevance of property exemplification

4.3.1. More on perdurantism: The worm theory and the stage theory

As said, my survey would seem to indicate the apparently contradictory GFO conception of persistence: GFO proposes a kind of perdurantism while offering a cognitive interpretation of persistence. To delve into this point, I introduce two prominent variants of perdurantism (mentioned in Section 2.1): the worm theory and the stage theory. The worm theory holds that a persisting object is a four-dimensional "worm" that stretches out through time and that is constructed out of those temporal parts of the object which are connected in some relevant way [16].[20] The stage theory maintains that a persisting object consists of "stages" (which would correspond approximately to the worm theorist's temporal parts) in such a way that the object name refers to different stages

[20] Popular answers to the question of linking temporal parts of perduring objects include spatiotemporal continuity [38] and sortal continuity [39]. Sortals are, broadly speaking, are a kind of linguistic terms (or of concepts) that take numerical modifiers, that is, can be associated with numerical adjectives [40]. For instance, the word 'cat' is a sortal because it is a linguistic term that takes numerical modifiers, as is observed by the fact that we can say "two cats". I myself think, following Williams [41], that they are at best necessary conditions for perdurance and only the right kind of causal connection is sufficient to solve this problem, though [31].

located within different times [17,18]. In this regard, the stage theorist denies diachronic identity of objects because she thinks that objects do not persist in the literal sense of the term. As she adds, objects are said to persist only in the sense that their different stages at different times stand in "temporal counterpart relations" (which would be likened to the worm theorist's connections between temporal parts).

The GFO view of persistence would be stage-theoretic in the sense of coupling a perdurantist ontological view of persistence with an epistemic stand on persistence. In fact, while BFO [23] focuses mainly on the worm theory and DOLCE [25] briefly mentions the stage theory when they discuss perdurantism, the GFO theory of persistence [29] reflects detailed analysis of the problems with the stage theory. It should not be classified as a stage theory in a full-fledged sense of the term, though.[21]

4.3.2. Complementing spatiotemporal location with property exemplification

The stage-theoreticity of the GFO view of persistence may help to explain (at least partially) why (SL) might be by itself an inadequate desideratum for understanding persistence in GFO. It is a prevailing orthodoxy that the stage theory is grouped together with the worm theory under the heading of perdurantism [15]. Suzuki [14] argues however that Donnelly's [13] version of (SL) *per se* would not classify the stage theory uniquely as perdurantism. (SL) admits of two readings, strict and moderate, depending on the construal of the term '(non-)persisting' therein. On the strict reading, the stage theory entails the non-existence of persisting objects and it is ontologically committed only to non-persisting stages, or instantaneous objects. Therefore, the stage theory can satisfy both (E-SL) and (P-SL) and it is categorized either as endurantism or as perdurantism. On the moderate reading (which Suzuki attributes to Donnelly), the term 'persisting' can accommodate the ordinary intuition that objects (e.g. my table) persist and an ontology of the stage theory comprises persisting objects. According to Donnelly's interpretation of the stage theory, a persisting object may exactly occupy a *unique three-dimensional* spacetime region. Suppose that my table exists at a time t_1 and at another time t_2. Then, its t_1-stage (resp. its t_2-stage) *is* my table: namely, an object that is, intuitively, spatially present at and only at t_1 (resp. t_2). The stage theory would be thus regarded neither as endurantism nor as perdurantism, because it satisfies neither (E-SL) ("multiple three-dimensional regions") nor (P-SL) ("a unique four-dimensional region"). In either case, (SL) does not categorize the stage theory only as perdurantism.[22]

This observation leads Suzuki [14] to suggest that endurantism and perdurantism should be characterized not only in terms of (SL) but also in terms of (EXEM), so that the stage theory can be analyzed uniquely as perdurantism based on the strict (and ontologically more foundational) reading of (SL). This is because the stage theory meets (P-EXEM), but not (E-EXEM). Assuming that my table is white at a time t_1, this is the case in virtue of the fact that the t_1-stage of my table exemplifies the property of being white. By (SL) and (EXEM), the stage theory is perdurantism because it satisfies both (P-SL) and (P-EXEM), whereas it is not endurantism because it meets (E-SL) but not (E-

[21] "Let us emphasize that (…) [the GFO] approach [to persistence] differs from the stage theory (…). For example, in stage theory processes are considered as mereological sums of stages, temporally extended entities (…). In contrast, processes in GFO do not have such stages as smallest parts." [29, p. 185]

[22] The same criticism can apply to other interpretations of (SL) than Donnelly's [13]. For instance, Balashov [22] provides a (SL)-based formulation of three theories of persistence which he calls 'endurantism', 'perdurantism', and 'exdurantism'. They correspond to endurantism, the worm theory, and the stage theory in the terminology of this paper. He does not give a unifying perdurantist framework for the latter two accounts.

EXEM). At present, it is not an established approach to classify theories of persistence according to (SL) and (EXEM). Still, this idea would be useful in clarifying the GFO stage-theoretic view of persistence. Lastly, note that DOLCE would espouse (EXEM) (see Section 3.2) and (EXEM) could be generally formalized using DOLCE-CORE [2]: the kind of modular ontology for upper-level entities that is inspired by DOLCE.

4.4. Persistence and time

Since persistence and time are intimately related [15], close scrutiny of ontology of persistence may serve to elucidate formal ontologies with respect to their temporal aspect. A foundational view on time can be generally characterized by theory choices concerning two issues [7]. One is the dispute over what we may call 'NOW' between the A-theory (aka the tensed theory) and B-theory (aka the tenseless theory).[23] We (directly) experience only the present time, but not any past or future time. NOW seems to move in one direction and the irreversible movement of NOW appears to be the single most important factor of our experience of the "passage" or "flow" of time. The question is whether NOW, the passage of time, and the distinction between the past ("before NOW"), the present ("contemporaneous with NOW"), and the future ("after NOW") are the objective (mind- and language-independent) characteristics of the real world or not. The A-theory says yes: the movement of NOW creates the passage of time from the past through the present towards the future [43]. The B-theory says no: NOW, the passage of time, and the purported past-present-future distinction are nothing more than the features of our experience of time, but not those of fundamental reality of time [11].

The other topic is the controversy over temporal ontology mainly between eternalism (see Section 2.1) and presentism (but see Section 5 for other temporal ontologies). Presentism says that only the present exists, or that only the present times, objects, and occurrents exist [44]. Imagine that one asks: "Does Socrates exist?" and "Does the 5 billion-year-old-earth exist?" The eternalist says yes to both questions, but the presentist says no to them. Because the presentist acknowledges the ontological specialty of the present and argues invariably for the A-theory, I will use the term 'presentism' to refer to a pair of the A-theory with the presentist temporal ontology. In addition, the B-theorist unexceptionally espouses eternalism and their couple is more often than not called 'block universe theory' [17]. While most eternalists are block universe theorists, some eternalists adopt the A-theory and endorse the moving spotlight theory [45]: the view that all the times, objects, and occurrents exist but the presentness is still privileged. A variety of theories of time are briefly summarized in Table 2.

Table 2. Theories of time

	Presentism ("Only the present exists")	Eternalism ("The past, present, and future exist")
A-theory	presentism	moving spotlight theory
B-theory	? (unspecified in the literature)	block universe theory

The first thing to note is that, contrary to e.g. Merricks [46], endurantism would be compatible with eternalism, at least insofar as either (SL) or (EXEM) is concerned.[24]

[23] The terms 'A-theory' and 'B-theory' are usually attributed to McTaggart's [42] terms 'A-series' and 'B-series' of time in his argument for the unreality of time, respectively.

[24] By comparison, perdurantism would be hardly consistent with presentism, regardless of whether (SL) or (EXEM) is adopted [14]. I omit to detail this point owing to spatial limitations.

For one thing, (E-SL) can be generalized enough to be acceptable for both the eternalist and presentist endurantists in such a way that each object exactly occupies a three-dimensional spacetime region at each time when it is present. The presentist endurantist would interpret the phrase therein 'at each time when it is present' as saying "whenever it exists", because she thinks that to exist presently is to exist *simpliciter* [13]. For another, (E-EXEM) is adaptable to eternalism, notably because it can apply to property exemplification regarding diachronic change ("… exemplify properties *at a time* in virtue of …" therein; emphasis added) while allowing persisting object to exemplify properties otherwise in the eternalist way that to exist in time is to exist *simpliciter* [14].

Furthermore, it has been recently argued that, according to (SL), the shift from classical to relativistic spacetime would favor perdurantism (the worm theory, in particular) over endurantism [13,22,47].[25] Based on Donnelly's [13] construal of (SL), for instance, (P-SL) can still be efficacious for specifying perdurantism in the relativistic worldview. In contrast, it is unclear how (E-SL) characterizes endurantism under the relativistic assumptions because (E-SL) is deeply rooted in the idea of absolute timeslice, which is incompatible with relativity. This does not *ipso facto* mean the inconsistency between endurantism and relativistic modern physics, but it is equally true that some substantive work is required for a relativistic reconstruction of (E-SL) [20,48].[26]

All those considerations as to the relationship between persistence and time could help to provide a better understanding of time in upper ontologies [6]. Let me focus only on BFO for space reasons. First of all, BFO adopts a realist approach to ontology development which prescribes that ontologies should represent entities in reality [51]. Since BFO endorses endurantism (see Section 3.1) and endurantism turns out to be compatible with eternalism (as well as the presentist temporal ontology), BFO may be willing to embrace the moving spotlight theory or the block universe theory, rather than presentism. For one thing, eternalism would serve better the general purpose of simpler ontological modeling, given the BFO principle of ontological realism.[27] Notably, Galton [6] focuses on the BFO category of *zero-dimensional temporal region* as illustrated by "right now" [23, pp. 124-125].[28] He states that, if it is to be taken seriously, this "right now" would entail an ontological commitment to the A-theory, granted the BFO realist methodology.[29] Following his advice to avoid the potentially problematic A-theoretic property of being "right now", BFO may be further motivated to embrace the B-theory, above all the block universe theory. Finally, it remains to be seen whether and how the BFO endurantist framework can hold in the relativistic setting (see also [10]).

---

[25] See also Galton's [6] detailed discussion on special relativity, especially in relation with BFO, DOLCE, and GFO.

[26] For persistence under another physical assumption, see Pashby's [49] argument that a (SL)-based characterization of persistence in general (which is not restricted to Donnelly's [13]) can be problematic for both endurantism and perdurantism in quantum mechanics. Additionally, (P-EXEM) may deserve further consideration in quantum mechanics, given his other argument [50] that persisting quantum objects do not have temporal parts.

[27] The latest version of BFO [24] elucidates the term 'entity' as "anything that exists or has existed or will exist" (see Section A.1.2.1 of the Excel file named "bfo-2020-terms.xlsx" found in: https://standards.iso.org/iso-iec/21838/-2/ed-1/en/). This might be possibly construed as a commitment to eternalism. In practice, however, realist ontologies would represent entities only in the past and the present, but not in the future, since we are presently never certain about (entities in) the future.

[28] The latest version of BFO [24] would seem to drop out this example, though. See Section A.1.2.75 "zero-dimensional temporal region" of the Excel file named "bfo-2020-terms.xlsx" found in: https://standards.iso.org/iso-iec/21838/-2/ed-1/en/.

[29] See also Bittner's [10] criticism of the more general BFO category of *temporal region*.

5. Concluding remarks

This paper aimed at initial elucidation of a foundation for ontology of persistence. For this purpose, I specified four existing desiderata (viz. (TP), (PART), (SL), and (EXEM)) for characterizing endurantism and perdurantism, examined persistence in three upper ontologies (BFO, DOLCE, and GFO) by those criteria, and discussed some key topics emerging from this survey. My overall contention is the modest proposal that formal-ontological investigation into persistence should be updated by expanding its perspective beyond the issue of whether objects have or lack proper temporal parts.[30]

I conclude with three brief remarks on future work. First, given the relevance of (SL) to persistence, spatiotemporal locationality of objects will require further exploration (see Section 4.1.1 for a pointer to this direction of inquiry in BFO). Second, careful investigation is warranted into the nature of time [6], especially into other temporal ontologies than discussed above: e.g. the growing block theory [52,53] ("The past and the present exist, but the future does not"; see also [7,54]) and the shrinking block theory [55] ("The present and the future exist, but the past does not"). Third, it is well worth considering the relationship between persistence and scientifically important notions of "natural necessity" [56]: e.g. causation, dispositions, laws of nature, and counterfactuals.

References

[1] Borgo S, Hitzler P. Some open issues after twenty years of formal ontology. In: Proceedings of FOIS2018. Amsterdam: IOS Press; c2018. p. 1-9.
[2] Borgo S, Masolo C. Ontological foundations of DOLCE. In: Poli R, Healy M, Kameas A, editors. Theory and applications of ontology: computer applications. Springer; 2010. p. 279-95.
[3] de Cesare S, Gailly F, Guizzardi G, Lycett M, Partridge C, Pastor O. 4th international workshop on ontologies and conceptual modeling (Onto.Com). In: Proceedings of JOWO2016. CEUR Workshop Proceedings, vol. 1660; c2016. p. 1-3.
[4] Quine WV. Identity, ostension, and hypostasis. J. Philos. 1950 Oct;47(22):621-33.
[5] Chisholm RM. Person and object: a metaphysical study. La Salle (IL): Open Court; 1976. 230 p.
[6] Galton A. The treatment of time in upper ontologies. In: Proceedings of FOIS2018. Amsterdam: IOS Press; c2018. p. 33-46.
[7] Toyoshima F. Ontology of time for the digital humanities: a foundational view. In: Proceedings of JOWO2019. CEUR Workshop Proceedings, vol. 2518; c2019. p. 1-6.
[8] North J. A new approach to the relational-substantival debate. In: Bennett K, Zimmerman DW, editors. Oxford studies in metaphysics, vol. 11. Oxford: Oxford University Press; 2018. p. 3-43.
[9] Maudlin T. Philosophy of physics: space and time. Princeton University Press; 2012. 200 p.
[10] Bittner T. Formal ontology of space, time, and physical entities in classical mechanics. Appl. Ontol. 2018 May;13(2):135-79.
[11] Mellor H. Real time II. London: Routledge; 1998. 160 p.
[12] Le Bihan B. String theory, loop quantum gravity and eternalism. Eur. J. Philos. Sci. 2020 Mar;10:17. https://doi.org/10.1007/s13194-020-0275-3
[13] Donnelly M. Endurantist and perdurantist accounts of persistence. Philos. Stud. 2011 May;154(1):27-51.
[14] Suzuki I. What do four-dimensionalism and three-dimensionalism disagree about? J. Jpn. Assoc. Philos. Sci. 2017;44(1-2):15-33. [in Japanese]
[15] Haslanger S, Kurtz RM, editors. Persistence: contemporary readings. Bradford; 2006. 483 p.
[16] Lewis DK. On the plurality of worlds. Oxford: Blackwell; 1986. 276 p.
[17] Sider T. Four-dimensionalism. Oxford: Oxford University Press; 2001. 288 p.
[18] Hawley K. How things persist. Oxford: Oxford University Press; 2002. 232 p.

[30] I did not discuss (PART) in detail for the sake of space, but see Donnelly's [13] argument (which Suzuki [14] and I myself find plausible) that (PART) is an unsuitable criterion for specifying theories of persistence. It is interesting to note that Merricks [46] would seem to employ (PART) in his argument for the incompatibility between endurantism and eternalism.

[19] Olson ET. Temporal parts and timeless parthood. Noûs. 2006 Dec;40(4):738-52.
[20] Gilmore C. Where in the relativistic world are we? In: Hawthorne J, editors. Philosophical perspectives, vol. 20, metaphysics. Oxford: Blackwell; 2006. p. 199-236.
[21] Sattig T. The language and reality of time. Oxford: Clarendon Press; 2006. 240 p.
[22] Balashov Y. Persistence and spacetime. Oxford: Oxford University Press; 2010. 240 p.
[23] Arp R, Smith B, Spear AD. Building ontologies with Basic Formal Ontology. MIT Press; 2015. 248 p.
[24] ISO/IEC 21838-2 [internet]. Information technology - Top-level ontologies (TLO) - Part 2: Basic Formal Ontology (BFO); 2020 Mar. Available from: https://www.easybib.com/reference/guide/apa/website
[25] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. Wonderweb deliverable D18 - ontology library (final). LOA-NCR-ISTC; 2003. Available from: http://wonderweb.man.ac.uk/deliverables/D18.shtml
[26] Simons P. Location. Dialectica. 2004 Jun;58(3):341-47.
[27] Crisp TM, Smith DP. 'Wholly present' defined. Philos. Phenomenol. Res. 2005 Sep; 71(2):318-44.
[28] Herre H. General Formal Ontology (GFO): A foundational ontology for conceptual modelling. In: Poli R, Healy M, Kameas A, editors. Theory and applications of ontology: computer applications. Springer; 2010. p. 297-345.
[29] Baumann R, Loebe F, Herre H. Axiomatic theories of the ontology of time in GFO. Appl. Ontol. 2014; 9(3-4):171-215.
[30] Herre H. Persistence, change and the integration of objects and processes in the framework of the General Formal Ontology. In Petrov V, Scarfe AC, editors. Dynamic being: essays in process-relational ontology. Cambridge Scholar Publishing; 2015. p. 337-57.
[31] Toyoshima F. Linking temporal parts in processual biological ontology. In: Proceedings of JOWO2019. CEUR Workshop Proceedings, vol. 2518; c2019. p. 1-8.
[32] Grenon P, Smith B. SNAP and SPAN: towards dynamic spatial ontology. Spat. Cogn. Comput. 2004 Mar;4(1):69-103.
[33] Brodaric B, Hahmann T, Grüninger M. Water features and their parts. Appl. Ontol. 2019 Feb;14(1):1-42.
[34] Baumann R, Loebe F, Herre H. Towards an ontology of space for GFO. In: Proceedings of FOIS2016. Amsterdam: IOS Press; c2016. p. 53-66.
[35] Baumann R, Loebe F, Herre H. Properties defined on the basis of coincidence in GFO-space. In: Proceedings of JOWO2019. CEUR Workshop Proceedings, vol. 2518; c2019. p. 1-10.
[36] Wasserman R. Theories of persistence. Philos. Stud. 2016 Apr;173(1):243-50.
[37] Verdonck M, Sales TP, Gailly F. A comparative illustration of foundational ontologies: BORO and UFO. In: Proceedings of JOWO2018. CEUR Workshop Proceedings, vol. 2205; c2018. p. 1-12.
[38] Hirsch E. The concept of identity. Oxford: Oxford University Press; 1982. 336 p.
[39] Wiggins D. Sameness and substance renewed. Cambridge: Cambridge University Press; 2001. 274 p.
[40] Grandy RE. Sortals. In: Zalta EN, editor. The Stanford encyclopedia of philosophy. Winter 2016 ed. Available from: https://plato.stanford.edu/archives/win2016/entries/sortals/
[41] Williams N. Powerful perdurance: linking parts with powers. In Jacobs JD, editor. Causal Powers. Oxford: Oxford University Press; 2017. p. 139-64.
[42] McTaggart JME. The unreality of time. Mind. 1908 Oct;17(68):457-74.
[43] Prior AN. Thank goodness that's over. Philos. 1959 Jan;34(128):12-17.
[44] Ciuni R, Torrengo G, Miller K, editors. New papers on the present: focus on presentism. Philosophia Verlag; 2013. 340 p.
[45] Cameron RP. The moving spotlight: an essay on time and ontology. Oxford: Oxford University Press, 2015. 240 p.
[46] Merricks T. On the incompatibility of enduring and perduring entities. Mind. 1995 Jul;104(415):521-31.
[47] Gilmore C. Persistence and location in relativistic spacetime. Philos. Compass. 2008 Nov;6(3):1224-54.
[48] Gibson I, Pooley O. Relativistic persistence. In: Hawthorne J, editor. Philosophical perspectives, vol. 20, Metaphysics. Oxford: Blackwell; 2006. p. 157-98.
[49] Pashby T. How do things persist? Location relations in physics and the metaphysics of persistence. Dialectica. 2016 Sep;70(3):269-309.
[50] Pashby T. Do quantum objects have temporal parts? Philos. Sci. Dec 2013; 80(5):1137-47.
[51] Smith B, Ceusters W. Ontological realism: A methodology for coordinated evolution of scientific ontologies. Appl. Ontol. 2010 Nov;5(3-4):139-88.
[52] Broad CD. Scientific thought. London: Kegan Paul; 1923. 555 p.
[53] Correia F, Rosenkranz S. Nothing to come: a defence of the growing block theory of time. Springer Verlag; 2018. 197 p.
[54] Galton A. Guarino's possibilism. In Borgo S, Ferrario R, Masolo C, Vieu L, editors. Ontology makes sense: essays in honor of Nicola Guarino. Amsterdam: IOS Press; 2019. p. 167-76.
[55] Casati R, Torrengo G. The not so incredible shrinking future. Analysis. 2011 Apr;71(2):240-4.
[56] Toyoshima F. Natural necessity: an introductory guide for ontologists. Appl. Ontol. 2020 Feb;15(1):61-89.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200658

Towards GFO 2.0: Architecture, Modules and Applications

Patryk BUREK[a,1], Frank LOEBE[b], and Heinrich HERRE[c]
[a] *Institute of Computer Science, Faculty of Mathematics, Physics and Computer Science, Marii Curie-Sklodowskiej University, Lublin, Poland*
[b] *Computer Science Institute, University of Leipzig, Germany*
[c] *Institute for Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, Germany*

Abstract. The General Formal Ontology (GFO) is a top-level ontology that has been developed by the Onto-Med Research Group since the early 2000s. Since that time several new theoretical results have been achieved as well as numerous projects have utilized the ontology, especially in complex domains such as bioinformatics and medical computer science. This leads to the need for an up-to-date overview of GFO and access to its applications.
This paper represents the first step towards introducing the GFO 2.0 framework, which aims at the integration of the work that is already present, but scattered in various publications, and its provision as a ready-to-use and reusable framework. For this purpose we summarize key features of GFO so far, outline a novel modular architecture and survey first modules for GFO 2.0, linking to applications. Finally, a rigorous and systematic development process is indicated.

Keywords. top-level ontology, GFO, module

1. Introduction

The General Formal Ontology (GFO) is a top-level ontology that originated as a component from the broader GOL project (General Ontological Language), commenced in 1999 by the Onto-Med Research Group [1] at the University of Leipzig. Version 1.0 of the GFO specification was released in 2006 [2], followed by a minor revision 1.0.1 (draft) [3] in 2007. Since that time much new theoretical research on GFO has been pursued and presented as well as numerous applications have been realized.

The GFO is a broad conceptual framework partially formalized in first order logic (FOL) and serialized in the Web Ontology Language (OWL) [4]. The projects in which GFO was or is utilized, especially in complex domains such as bioinformatics and medical computer science, cf. e.g. [5-6], have demonstrated a wide spectrum of applications, including ontological and conceptual modeling, the development of domain ontologies and ontology design patterns as well as of Unified Modeling Language (UML) profiles [7].

[1] Corresponding Author, University address: pl. Marii Curie-Sklodowskiej 5, 20-031 Lublin, Poland; E-mail: patryk.burek@poczta.umcs.lublin.pl .

On the other hand, we recognize the need for a consistent and systematic overview of all new results. This relates further to the aim of providing GFO as a ready-to-use and reusable framework that is suitable for further applications in both research and industrial projects. Currently especially the readiness for use is hampered by the fact that many results remain hidden in discussion papers and reusable artifacts are missing.

Therefore, we aim at taking next steps. The actual artifacts of GFO, comprising of ontology files and/or UML profiles and accompanying material (guides/manuals, etc.), need to be integrated with new theoretical findings as well as with the results obtained during the application of the framework.

The current paper represents the first step towards introducing the GFO 2.0 framework. With the aim of eventually integrating all of the work, here we survey and develop a kind of a map of the existing bits and pieces. We expect such a collection to be already useful for future applications as well as for the further development of GFO itself, first and foremost as a new starting point towards releasing GFO 2.0 in a production-ready form. This release will cover new results, yet it will likewise comprise additional artifacts necessary for applying GFO in projects. Hence, our objective is not only to establish and elaborate theoretical foundations, but also to make them and GFO as a whole much more easily available for practitioners. Enhancing the development process with more rigor than during the last years, we will rely on an iterative and modular approach, as successfully adopted in software engineering. Accordingly, GFO should less be understood as a single, monolithic ontology. Instead, we will outline a conceptual framework and an ontology, the modules of which will form a coherent whole, on the one hand, but can be used flexibly and selectively to the extent required, on the other hand. Altogether, the current paper is a primer preparing the roadmap for the GFO 2.0 project.

The remainder of the paper is organized as follows. In the next section we give a short overview on key features and ontological choices of GFO and relate that with further state-of-the-art top-level ontologies. In section 3 we discuss the main architectural principles underlying the GFO 2.0 release, including the adopted modular approach to ontology development. Section 4 provides an overview of several main modules of GFO and serves therefore as a foundation for the roadmap of the whole project. Finally, section 5 concludes the paper with a summary and future work, in particular an outline of the roadmap and the development process.

2. State of the Art

An ontology is to be understood as a formalized system of categories and relations that describe knowledge about a domain of reality. Top-level or foundational ontologies contain those categories that can be applied to a very broad spectrum of domains. Nowadays, top-level ontologies (TLOs) play various roles: They serve as a framework for organizing domain-specific knowledge, in many cases of complex fields; they can be applied to the ontological analysis and the foundation of areas of knowledge about a domain; and they can be used to create new theories. Furthermore, TLOs provide a method to transcend the conceptual boundaries of a field of research. Meanwhile efforts are under way to offer TLOs as standards, not at least in order to provide more stability and reliability when referring to them or using them. In particular, the recent standard ISO 21838 [8] is devoted to defining requirements for TLOs.

2.1. Ontological Choices in GFO

In general, GFO is intended to become a scientific theory which covers empirical areas, such as physics, chemistry and biology, and likewise non-empirical domains, such as mathematics and set theory, but also the humanities. Subsequently, we summarize six special features that constitute the hallmarks of GFO.

(1) *Ontological regions and levels of reality.* In GFO, the world is subdivided into four ontological regions: the material region, associated to the natural sciences; the psychological region, which is related to the phenomena of the mind, such as intentionality; the ontological region of socio-systemic entities, including communication between subjects, social phenomena like social roles, economy, the role of labor for creating artifacts, and the inter-relation between man and nature. Finally, GFO covers the ontological region of ideas and abstract entities.

The conception of ontological regions is partially borrowed from the philosophy of N. Hartmann [9], cf. also [10], though there are differences, in particular in treating set theory as a particular ontological sub-region of the abstract region. Pursuing an integrative approach, we argue that important ideas of influential figures of history cannot be neglected,[2] such that a broad classification of modes of being is necessary to achieve a comprehensive picture of the world.

(2) *Ontology of the material region*. One main contribution of GFO in this field consists of a new theory of space and time [12], based on ideas of Franz Brentano [13]. Another contribution is the postulation of a basic law, called integration axiom. It says that for any material object there exists a process that corresponds to the object in a particular way. The axiom utilizes a new type of individuals, called presentials. Moreover, it is the fundament for a new classification of properties of spatio-temporal material entities [14] and it allows for a new interpretation of the particle-wave duality in quantum mechanics.

(3) *Ontology of categories and the multi-categorial approach*. GFO admits various kinds of categories, which are classified into universals (Aristotelian, Platonic), concepts, and symbolic structures. The individual-category dichotomy is a basic feature of formal ontology, however, it must be refined by taking the various kinds of categories into consideration. None of the mentioned types of categories can be neglected. Otherwise important phenomena related to the meaning of terms and to the communication between humans cannot be captured. Sets are included among the categories, yet they play a particular role, because they are based on the membership-relation, which differs from instantiation. An important feature of GFO is the introduction of types of higher order for concepts and for sets. This yields a powerful mechanism to represent concepts the instances of which are themselves concepts.

(4) *The principle of integrative realism*. GFO postulates the existence of a reality that is independent of the mind. Integrative realism is determined by two features. First, the subject has access to real and independent entities of the world only through concepts, being a part of the mind. Secondly, there is a law-like correspondence between the subjective phenomena of the mind and the independent reality. This approach leads to a new understanding of the relation between ontology and epistemology, because both cannot be separated. Ontology is directed at the independent objects of the world, though, these appear to the subject through the knowledge (involving perception and concepts)

---

[2] For example, Kurt Gödel defends a rigorous mathematical Platonism, where mathematical entities such as numbers and sets have an objective existence that is independent of the mind and of the material world [11].

that the subject has about them. This leads, we believe, to a new interpretation of works by G. Frege [15], B. Russell [16], and K. Twardowski [17]. Further, GFO pursues a logical approach to the denotation problem. One basic idea is that an entity of reality is reflected as a (logical) individual within a model-structure, which models a part of reality. The description of this individual can then be given against background knowledge in the form of a formal logical theory.

(5) *The onto-axiomatic method.* This newly established method combines formal ontology with the axiomatic method and the model theory of mathematical logic, with A. Tarski's work as an important pillar [18]. We postulate that the onto-axiomatic method provides an overarching principle for the rational reconstruction of existing theories as well as for the development of new scientific theories. The semantics of formalized theories must be further developed into an ontological semantics, cf. [19]. The application of the method combines and unifies top-down with bottom-up principles.

(6) *Top-level ontologies in an open evolutionary system.* The approach of a coordinated evolution [20] is rejected, if such an approach assumes some overarching coordinating top-level ontology within which ontologies evolve. P. Feyerabend is a strong critic of a similar principle applied to the development of science in general [21]. For GFO, the future development of ontologies is deliberately assumed to be open. This notwithstanding, there are some basic principles about the organisation and structuring of a network of top-level ontologies. Briefly indicated, these ontologies are presented in a formal language and there are various relations connecting them, among which we consider the interpretability relation to be of utmost importance. If one ontology can be interpreted in another ontology, then the latter is at least as expressive as the former. It can be expected that there will never be a single system covering all other systems, since the future is – so to say – open. The interpretability relation creates a partial ordering between ontologies, the investigation of which constitutes a research field of its own.

2.2. Comparison with other Top-Level Ontologies

Within limited space, we relate GFO selectively to some further representatives of top-level ontologies. The process ontologies of J. Seibt [22] and of M. West [23] are 4D-ontologies, where objects are special processes. These ontologies lack the deep duality between objects and processes, and a particular feature of objects disappears, the phenomenon of being wholly present at a time point. Hence, the notion of presential cannot be explicated, because a process cannot be wholly present at a time point. Furthermore, both ontologies do not provide any account of categories.

Turning to pure 3D-ontologies, BFO (Basic Formal Ontology) [24] analyzes the distinction between objects and processes differently. In BFO processes are – so to say – properties of objects, they depend on objects. Furthermore, BFO does not allow for an ontology of categories, in particular concepts are excluded from BFO. Universals in BFO resemble Aristotelian universals, while there is no place for ideal entities, such as platonic universals or sets. The treatment of boundaries in BFO is a relevant contribution. The topic has been adopted and advanced by GFO, up to a complete axiomatization in connection with the analysis of time [12] and solving a problem of touching entities.

DOLCE [25, 26] exhibits some commonalities with GFO, yet also relevant differences. DOLCE contains a classification of properties, which can be understood as categories. But a basic distinction between concepts, universals, and symbolic structures appears to be missing, likewise a full type-system (with concepts of higher order).

DOLCE is another pure 3D-ontology, where we are not aware of an integration axiom like that of GFO. For a more detailed comparative study on DOLCE and GFO, see [14].

The Unified Foundational Ontology (UFO) [27, 28] incorporates early developments from DOLCE, the Ontology of Universals underlying OntoClean [29] and aspects of GFO into a single coherent foundational ontology. UFO is essentially a 3D-Ontology. The notion of type as used in UFO, corresponds to the general notion of category in GFO, though we see no distinction between concepts, universals and symbolic structures. Similarly, sets do not seem to play any relevant role in UFO.

In summary, the top-level ontologies mentioned above do not satisfy all characteristics (1)-(6) of GFO in sect. 2.1. While we believe in a high level of expressiveness of GFO, it remains an interesting project to study the mutual interpretability among the ontologies.

3. Architecture for GFO 2.0

Concerning the architecture of GFO, we note first that very early formalizations already comprised a few hundred formulas in first-order logic. Developing, managing and maintaining such a theory as a monolithic artifact is hardly feasible and can prevent or at least hinder its usage. Since then and to some extent inspired by software engineering, the modularization of ontologies has been studied from several angles, cf. [30, 31], even though there is no universal way to modularize an ontology.

Over the years, GFO has been continuously elaborated, revised and applied. This includes the formation of modules to a certain extent, but in a rather ad-hoc way. In working on GFO 2.0 we pursue an explicitly modular approach with some novel aspects.

3.1. Modules

First of all and at its core, by a *module* we understand basically a theory of some (typically limited) modeling problem or area. Moreover, a module constitutes a coherent, ideally self-contained part of a larger system (in our case, of GFO). At least within that system each module is expected to be extensible in the sense that it is well geared to other modules and can be used in consistent combination with them.

'Module' and 'theory' are here to be understood in a conceptual sense. Insofar a module can also be described as a conceptual framework of the problem or area that it addresses. Accordingly, a module is not language-specific, neither regarding any natural, nor any formal language. Of course, in order to capture and provide a module we need to rely on language(s).

This yields a distinction between, on the one hand, 'module' in the conceptual reading described so far – from now on referred to as the *content of a module* – and, on the other hand, artifacts intended to represent that content – *representational artifacts of a module*. Examples of such artifacts are specifications in formal or semi-formal languages, such as axiomatizations in first-order logic, OWL patterns and UML profiles. Note the semantic shift, in that now the term 'module' incorporates both aspects, a content side and a representational side. For simplicity,[3] we use 'module' to refer to either side or jointly to both sides.

Not each artifact of a module must cover all of its content. It is further rarely the case that a single representation in a given modeling area is the best option in all

[3] and as is common for dot types, cf. [32]

application or implementation scenarios. Therefore, a single module may contain many patterns. Similarly, there may be more than one axiomatization, e.g. due to using different formal languages or formalizing the module content by means of the same language in different ways.

In understanding an ontology as a collection of modules in the sense of conceptual frameworks we summarize the following benefits:

(1) The module (its content) is agnostic with respect to a particular formalism or implementation language, i.e., there is no implementation bias. Instead, a module may be provided in the form of multiple implementations for different application contexts.

(2) Modules are intended to fit and function together, but should be designed to be maximally independent from one another. Consequently, they can be used separately, but with built-in extensibility if other modules may need to be included.

(3) While modules as conceptual frameworks appear similar to ontology design patterns, the latter – at least in all cases we are aware of – appear attuned to a certain level of granularity, and some of them have an application-bias. The notion of module as described above responds to such bias through the offer of its content in (potentially) several languages. Moreover, it is intended to cover a whole range of module sizes, from very small modules to possibly large and complex frameworks.

3.2. Further Organization Principles

From an architectural point of view, three aspects account for further dimensions in the organization of GFO. The first draws on the notion of *ontological regions* as discussed in sect. 2.1. Secondly, there are several *top-level distinctions*, i.e., very generic and widely applicable distinctions that re-occur across different ontological regions. Prime examples are “classical” distinctions, such as objects vs processes, or attributes vs their bearers vs facts and situations. Thirdly, *meta-level abstraction* distinguishes ordinary categories from those categories involved in analyzing ontologies themselves, as well as from categories of formal entities (such as sets) that are employed in connection with the formalization of ontologies.

Let us consider those three aspects and their manifestations in the context of GFO and its modular structure more closely, starting with the four major ontological regions. The material region is the one among those four that has been developed furthest within GFO. This applies also with regard to the treatment of top-level distinctions, which has frequently been tested and applied in connection with material entities. Ontological regions may also be viewed as modules, notwithstanding their huge extent as theories. This applies in the sense of the contents of modules, whereas, at the current stage, GFO comprises no representational artifacts that correspond to regions as a whole.

Instead, the modules presented in the next section – for example, (i) categories and individuals, and (ii) attributes – largely result from the study of top-level distinctions and associated kinds of entities. These modules are much more limited in size than ontological regions, for which reason they are more manageable. Furthermore, those distinctions have been analyzed for decades in information sciences and for centuries in philosophy. This has yield many inspirations when establishing these theories.

Finally, certain modules (or fragments of them) dealing with top-level distinctions lend themselves to the analysis of ontologies and their constituents (and the formalization of ontologies). First and foremost this is the case for the theory of categories and individuals. In the present paper we do not elaborate on the self-analytic nature of these modules and their value for meta-level considerations, but we refer to [33] for that.

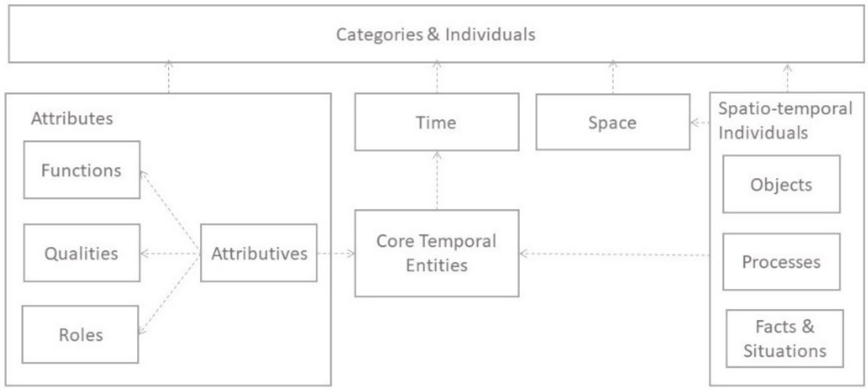


Figure 1. UML-like diagram depicting the modules of the GFO and their dependencies. Each rectangle represents a module with optionally embedded submodules. Dashed directed arrows indicate dependencies.

4. Overview on Selected GFO Modules

The current section provides an overview of the main modules of GFO 2.0. The findings are organized into modules together with applications. A UML-like components diagram in Figure 1 depicts the organization of the modules.

4.1. Categories and Individuals

A basic distinction in formal ontology is that between individuals and categories. Individuals are uniquely determined entities which cannot be instantiated. In contrast, categories are instantiable and can be predicated of other entities. We postulate that this basic distinction holds in any of the ontological regions. Categories themselves are classified into universals (subdivided into platonic and immanent universals), concepts and symbolic structures.[4] Sets are entities that behave in some respect similar to categories, but on the other hand, they differ from them because they cannot be predicated of other entities. Platonic universals are independent of the subject[5] and of the material world, they have an ideal, mind-independent existence. They belong to the ontological region of ideal entities. Typical entities of this kind are mathematical entities, for example as occurring in pure geometry and number theory. Immanent universals (which correspond to Aristotelian universals) are in the real material things; they are independent of the subject, though they depend on the real things. Concepts are in the mind, they are creations of the subject. Since the subject is the bearer of any intellectual activity, concepts play the central role in any process of contemplation, perception, and theory formation. We acquire the world through our concepts and therefore we consider them as the most important kind of categories. Concepts provide the basic means to access any other entity. Insofar they realize an interface between the subject and the

---

[4] This distinction draws partly on the theory of J. Gracia in [34].

[5] The term *subject* has the meaning of *recognizing self* and is synonymously used for the term *mind*. The relation between subject and object is a central topic of the European philosophy since 1500, culminating in the classical German philosophy of the 19th century.

entities outside of and independent of the mind.[6] For any of these kinds of categories (concepts, immanent universals, platonic universals) we introduce a basic relation: an entity e is an instance of the concept C, formally: instanceOf(e, C). We further distinguish the extension and the intension of a category. The extension of a category is defined as the set of its instances, whereas its intension refers to its meaning or content. We assume that the intension of a concept has parts, which we call *categorial parts*.

There remain two further kinds of categories, namely symbolic structures and sets. Symbolic structures have tokens as instances, which are material entities and therefore individuals. Furthermore, for symbolic structures (and their instances) we introduce another basic relation, the relation of denotation with the meaning den(s, S, e) := 'the token s (being an instance of the symbolic structure S) denotes the entity e'. The relation den(X, Y, Z) can be iterated, which may lead to complex denotation systems. Further, we consider sets as a particular kind of category. Sets exhibit a full simple type system, which can similarly be established for concepts. However, sets clearly differ from concepts, a.o. because their basic relation is the membership relation, introduced as 'the entity e is an element of the set S'.

A few examples of axioms follow, regarding the signature of which we explain Int(·) for intension, int(x, y) for x being the intension of y, ext(y, x) for y being the extension of x, and cpart(x, y) for x being a categorial part of y.

1. $\text{Int}(C) = \{c \mid \text{cpart}(c, C)\}$
2. $\neg \exists x\, (\text{Cat}(x) \wedge \text{Ind}(x))$
3. $\forall x\, (\text{Cat}(x) \rightarrow \exists\, y\, (\text{Set}(y) \wedge \text{ext}(y, x))$
4. $\forall x\, y\, (\text{Cat}(x) \wedge \text{Cat}(y) \wedge \text{int}(x, y) \rightarrow \text{ext}(y, x))$

Note that axiom (1) represents a simplified approach, for which we assume in addition that cpart satisfies the conditions of a partial ordering with least upper bound for any two of its elements, as well as that any categorial part contains (or is) an atomic part.

Aspects of this module were applied in the analysis of the notion of core ontology and the development of GFO-Bio [35], a core ontology for biology, as well as in an ontological theory of the notion of data element [36] resulting from the MDR project on a metadata repository for clinical and epidemiological research. The method of search ontologies as introduced and employed in the projects Ontovigilance and OntoPMS [5] relies particularly on the relation between symbolic structures and denoted concepts.

4.2. Time and Space

Space and time are basic categories which account for fundamental assumptions behind those individuals that are said to be in space and time. We introduce two modules *Space* and *Time* in GFO 2.0 for covering notions related to spatial and temporal phenomena.

The module of time has a high impact on information systems and the understanding of time is fundamental for numerous domains, since the representation of a domain typically involves entities related to time, such as events, changes and processes. The significance of the conception and representation of time entities and reasoning about

---

[6] We emphasize that GFO defends a realist philosophy, introduced as *integrative realism*, expounded in some detail in [14]. We reject Kant's artificial separation between *thing as such* and its *appearance*. The main arguments against such a separation were already explicated by Hegel in his *Phenomenology of the Spirit*.

temporal data and knowledge is supported by the presence of the time concept in most top-level ontologies as well as by dedicated time ontologies, e.g. OWL-Time [37].

The Time module consists of two submodules, *Time Core* and *Time Region*, and is already backed by a rich axiomatization in first-order logic. The latter has been analyzed metalogically, proving consistency, completeness and decidability as presented in [12]. The mature formalization of the module goes hand in hand with low ontological commitment, inspired by ideas of Franz Brentano [13]. The Time Core submodule introduces two basic temporal categories, namely intervals (also called chronoids) and time points (aka time boundaries). This basic distinction is a foundation for further modules of GFO, especially for the core temporal module. There is the assumption that the temporal continuum can be introspectively accessed without any metrics, and that it cannot be understood and grasped only in terms of sets of time points. Equally, neither can intervals be simply reduced to the sequence of time points, nor should they be identified with intervals (sets) of real numbers. Central basic relations of the module are temporal part-of, the temporal coincidence of time boundaries, and relations linking time boundaries with chronoids. The Time Region submodule extends Time Core by addressing mereological sums of chronoids, referred to as time regions.

The Space module in turn accounts for basic notions for representing spatial entities, also axiomatized in FOL. The theory is likewise inspired by the ideas of Franz Brentano on space, time and the continuum and it starts from four primitives: the category of space regions, the relations of being a spatial part and being a spatial boundary, as well as the relation of spatial coincidence.

Both modules of Space and Time are fundamental elements of GFO 2.0 and they supply basic notions utilized in further modules of the framework. Especially, Core Time is used in the Core Temporal Entities module and Space in Spatio-Temporal Individuals. Moreover, they are directly relevant in connection with projects on the navigation in surgical interventions, cf. BioPass and COMPASS [38].

4.3. Core Temporal Entities

The module *Core Temporal Entities* is responsible for representing all entities located in time. As such it depends on the Time module and more specifically on the distinction between time intervals and time points introduced there. On that basis we introduce two notions, *presentials* and *time extended entities* (TEEs). The former are immutable entities, which exist and are fully present at exactly one time point, therefore having no temporal extension. Time extended entities exist over a time period and thus can undergo changes. Presentials and TEEs are glued together by the snapshot relation, denoted snapshot(x, y), 'presential x is a snapshot of time extended entity y'. For each presential there exists a time extended entity that that presential is a snapshot of. From a modeling point of view presentials can be interpreted as reified temporal snapshots of TEE.

In its current state the module comes together with the OWL Temporal Entities Pattern [39] which supports the modeling of an interplay between presentials and TEEs as it is needed in common cases where data on presentic observations is aggregated into time extended entities. The pattern is applied in connection with the development of the Cell Tracking Ontology (CTO) [39], there for gluing raw observation data to more complex data structures such as cells and cellular genealogies.

4.4. Spatio-Temporal Entities

Material entities are classified into objects, processes and entities that depend on them. *Objects* occupy space and endure through their lifetime, which means that they are the same through their life-time and "are" wholly present (exhibit a presential) at any time point of their life-time. Processes evolve through time and have a temporal extension.

Subsequently, we collect some exemplary axioms on objects.[7]

5. $\forall x\ (\mathrm{Obj}(x) \rightarrow \exists y\ (\mathrm{Chron}(y) \wedge \mathrm{lifetime}(x, y))$
6. $\forall x\ (\mathrm{Obj}(x) \rightarrow \exists y\ (\mathrm{SReg}(y) \wedge \mathrm{occ}(x, y))$
7. $\forall x\ y\ (\mathrm{Obj}(x) \wedge \mathrm{mpart}(y, x) \rightarrow \exists u\ (\mathrm{occ}(y, u) \wedge \mathrm{SReg}(u)))$
8. $\forall x\ y\ t\ (\mathrm{Obj}(x) \wedge \mathrm{lifetime}(y, x) \wedge \mathrm{tp}(t, y) \rightarrow \exists z\ (\mathrm{exhib}(x, z, t)))$
9. $\forall x\ y\ t\ (\mathrm{exhib}(x, y, t) \rightarrow \mathrm{Pres}(y))$

Processes evolve through time, they cannot be wholly present at time points. Since we restrict considerations in this section to material processes, we stipulate that any process boundary contains a material presentic object as a part. This presentic object possesses object qualities, though further properties can be associated with this object that have their origin in a process, for example, the velocity of a moving object at a time point.

Objects and processes are not isolated from each other, they are integrated in a particular way. This integration is postulated by the integration law, a principal axiom of GFO.

10. $\forall x\ (\mathrm{Obj}(x) \rightarrow \exists y\ (\mathrm{Proc}(y) \wedge \forall z\ t\ (\mathrm{exhib}(x, z, t) \leftrightarrow \mathrm{procbd}(y, t, z))$
    Integration Law: For any object there is a process such that its process boundaries (procbd) coincide with the presentials exhibited by the object.

Situations are responsible for representing aggregates of objects comprehensible as wholes. In GFO, situations are built upon objects and individualized relations connecting them, as for instance in the case of a cup standing on a table, in which the cup and the table are the objects, stands_on is a relation connecting them and standing_object and supporting_object are the roles of the objects in the context of that relation. All those entities taken together and perceived as a whole is considered as a situation. For every situation *S* there exists an object participating in *S*. If an object as such can be understood as a whole composed of individual qualities connected with their bearer by the inherence relation, then even a single object may be considered as a situation. That interpretation resembles the ontology of Tegtmeier [40], where situations (aka states of affairs) are the basic entities and all other entities (object, processes etc.) are their special cases.

The module is currently equipped with the OWL Situation Pattern which has been applied to the CTO [39]. It enables reconstructing from the raw data of cell tracking experiments the more complex structures called cellular genealogies, which consist of multiple cells linked by relations such as e.g. cell divisions. Other applications of the module are devoted to medical computer science, among them BISON, an ontology of minimally-invasive surgical procedures based on situational information extracted from endoscopic, procedural and sensory data [41].

---

[7] using this signature: Obj(·) for object, Pres(·) – presential, Chron(·) – chronoid, SReg(·) – space region, exhib(x, y, t) – 'object x exhibits the presential y at time point t', lifetime(x, y) – 'x is the lifetime of y', mpart(x, y) – 'x is a material part of the object y', occ(x, y) – 'object x occupies space region y', tp(t, y) – 't is a time point within y'.

4.5. Attributes

The two major generic notions of the *Attributes* module are termed attributives and properties. In contrast to the individuals discussed so far, *attributives* are individuals that characterize other individuals, e.g. the function of a molecule or the color of an apple. Attributives are instances of particular categories that we call *properties*. The module supports modeling both, properties and attributives, and it consists of four submodules. Three of them deal with representing different kinds of properties: qualities, roles and functions. The fourth concerns the assignment of attributives to time related entities.

4.5.1. Qualities

The Qualities submodule covers *qualities* and *quality values*. The former are properties "which are typically expressed in natural and artificial languages by means of syntactic elements such as adjectives/adverbs or attributes/properties, respectively" [2], such as weight, color, speed or temperature. Values in turn are quantities used in measurement, observation or calculation of qualities, e.g. 10kg, green. Often they are scalars or vectors. Both qualities and values are considered as abstracts having no relation to time and being independent from the entities which are characterized by them.

An assignment of a quality and a value to an individual is called an *individual quality*. In case of time-related quality bearers the assignment itself is also of some temporal extent and may involve the change of quality values over time. Therefore, in order to model qualities which change values in time we introduced the OWL Temporal Qualities Pattern which is motivated by the observation that a straightforward approach to modeling qualities in OWL cannot represent the change of quality values adequately [42]. The pattern has been applied to the CTO and it supports consistent modeling of (1) the qualities observed at a single time point, (2) static, non-changing qualities of enduring entities and (3) dynamic, changing qualities of enduring entities [39]. In connection with the Leipzig Health Atlas (LHA) the module forms the basis for specifying complex phenotypes [43].

4.5.2. Roles and Relations

The notion of *roles* is pervasive in many domains, because many categories can be revealed as roles. Roles depend on the contexts that "define" them. This leads to the difficulty that the nature of distinct kinds of roles varies greatly. Loebe [44] distinguishes three major kinds: relational, processual and social roles.

There is a two-fold approach regarding roles in the module structure for GFO 2.0. On the one hand, a very generic module covers the notion of role and neighboring categories and relations with a weak theory. On the other hand, more specific notions of roles are dealt with in connection with other modules, typically in modules where the contexts of a role type are covered.

Relations in GFO are understood as categories of particular entities that "glue their arguments together", where the latter are called *relators*. This differs clearly from formal approaches of viewing relations as mere sets of *n*-tuples of the corresponding arguments. Relations are one of the common kinds of contexts for roles. Indeed, relators are composed of (relational) roles as well as relations are associated with a number of role categories, cf. [45] for more details on this theory.

4.5.3. Functions

The Functions module is responsible for representing functions as commonly used for characterizing entities across many domains, including engineering (e.g. the function of an engine), the natural sciences (e.g. the function of blood cells) and the social sciences (e.g. the function of a manager). Despite many works on functions over the last decades, consensus on a single reading of the term has not been reached. For GFO we rely on a popular account of function formulated in terms of the notion of role, where a *function* is seen as a category that captures a role in the context of some goal achievement.

The Functions module provides patterns for representing function specifications as well as for modeling interdependencies between functions, e.g. function decomposition. Attributive assignments from the Attributes module support the modeling of the assignment of a function to an entity, in the sense of an entity having a function.

The specification of a single function in GFO (aka the structure of function) grasps two aspects: (i) the input-output perspective (traditional for function modeling), which includes modeling goals and side effects, and (ii) the participant perspective, covering entities involved in the realization of a function, such as ‘doer’, ‘contributor’ and ‘instrument’. Moreover, functions are rarely modeled in isolation, but usually interrelated in complex networks. The Functions module introduces several distinct relations typically hidden behind functional decomposition [46, 47], among them *operand-part* and *function-subsumption*, the latter with subdivisions such as *specialization of mode of realization* and *specialization of function operand*.

The notions of the Functions module have been used for the development of FueL, a UML profile suitable for functional modeling with UML-like graphical notation [48]. The case study of applying FueL to the Gene Ontology revealed several refactoring options [49]. Additional uses in the context of the biomedical sciences include representing knowledge about biological functions [50], dispositions and functional abnormalities [51].

5. Conclusions and Future Work

We present the top-level ontology General Formal Ontology (GFO), originally released in version 1.0 in 2006, focusing on recent results and its future development. Those results include basic research on formal ontology as well as numerous practical applications, notably in the field of biomedical computer science, but also in the areas of semantic methods and knowledge representation.

The overview of existing work serves as the starting point for the establishment of an integrated framework and architecture for GFO 2.0. Based on this framework, GFO is less to be seen as a single, monolithic ontology, but rather as a system of coherent modules, which can be flexibly used together, depending on a chosen domain and context. GFO 2.0 will thus provide integration principles which should support the composition of various modules to larger unities, depending on the domain to be studied and the level of abstraction of the considered application.

The present paper addresses some of these basic topics. Architectural considerations are outlined, including an explication of the employed understanding of ‘module’ and further organization principles. In the second part of the paper several selected modules are surveyed and partially linked to existing formalizations and applications.

Our goal for the future is to approach the GFO 2.0 development as a software product, following state of the art industry standards. This will include an explicit roadmap and release planning, following an iterative development approach based on architectural principles for software engineering and working with a backlog. The main architectural principles adopted for GFO 2.0 are (i) a modular approach and, on that basis, (ii) a release plan adhering to an iterative methodology. Accordingly, the modules will each be released separately and in connection with rigorous versioning (major and minor revisions, snapshots, releases) and maturity level tracking for the overall project as well as for the individual modules. For each release version corresponding artifacts shall be published, e.g., on GitHub and as a paper. The present paper is a primer for the roadmap of the GFO 2.0 project.

This overall approach differs from the course of development that led to version 1.0 of GFO (and from that of initial versions of other top-level ontologies) and significantly strengthens earlier, but less systematic steps, e.g., towards a more modular structure. A TLO developed along these lines should no longer be conceived as a large, monolithic theory. In the modular approach, it is rather to be seen as a coherent framework composed of easily replaceable parts that use standardized interfaces, such that those parts constitute building blocks readily applicable to specific engineering problems.

But what will those parts, what will a module look like, given that we aim at very diverse application contexts such as conceptual modeling and the Semantic Web? Several kinds of artifacts are already foreseen or do already exist, among them (i) textual descriptions (research papers, reports, use cases, tutorials, etc.), (ii) axiomatizations in FOL, Common Logic and/or OWL, (iii) ontology patterns, e.g., in OWL and (iv) UML specifications and profiles for applications in conceptual modeling.

Returning to ontological content once more, this project will not only comprise of the modules presented herein. For example, two longstanding open issues are a dedicated ontology of generic social entities, and an extension and refinement of the theory of symbol structures. Overall, clearly and certainly we envisage new modules in the future, not at least inspired by application needs.

References

[1] Onto-Med Research Group. Home [Internet]. IMISE, University of Leipzig, Germany; [cited 2020 Aug 15]. Available from: https://www.onto-med.de/.

[2] Herre H, Heller B, Burek P, Hoehndorf R, Loebe F, Michalek H. General Formal Ontology (GFO) [Version 1.0]. Onto-Med Report 8. IMISE, University of Leipzig, Germany; 2006.

[3] Herre H, Heller B, Burek P, Hoehndorf R, Loebe F, Michalek H. General Formal Ontology (GFO) [Version 1.0.1]. Draft. IMISE, University of Leipzig, Germany; 2007.

[4] Onto-Med Research Group. General Formal Ontology (GFO): Axiomatizations and Implementations [Internet]. IMISE, University of Leipzig, Germany; [cited 2020 Aug 15]. Available from: https://www.onto-med.de/ontologies/gfo#impl.

[5] Uciteli A, Kropf S, Weiland T, Meese S, Graef K, Rohrer S, et al. Ontology-based specification and generation of search queries for post-market surveillance. Journal of Biomedical Semantics. 2019;10(1):9.

[6] Uciteli A, Neumann J, Tahar K, Saleh K, Stucke S, Faulbrück-Röhr S, et al. Ontology-based specification, identification and analysis of perioperative risks. Journal of Biomedical Semantics. 2017;8(1):36.

[7] Rumbaugh J, Jacobson I, Booch G. The UML Reference Manual. 2nd ed. Addison Wesley; 2005.

[8] ISO/IEC 21838-1. Information Technology — Top level ontologies TLO — Part 1: Requirements. 2020.

[9] Hartmann, N. Ontologie. De Gruyter; 1935–1950 (4 Vol.).

[10] Poli R. The Basic Problem of the Theory of Levels of Reality. Axiomathes. 2001;12(3-4):261–283.

[11] Gödel K. Collected Works III. Oxford University Press; 1995.

[12] Baumann R, Loebe F, Herre H. Axiomatic theories of the ontology of time in GFO. Applied Ontology. 2014;9(3-4):171–215.

[13] Brentano F. Philosophische Untersuchungen zu Raum, Zeit und Kontinuum. Felix-Meiner Verlag; 1976.
[14] Herre H: Towards a New Foundational Ontology of Properties, Attributives and Data. In: Ontology Makes Sense. IOS Press; 2019. p. 194-210.
[15] Frege G. Über Sinn und Bedeutung. Zeitschrift für Philosophie und philosoph. Kritik. 1892;100:25–50.
[16] Russell B. On Denoting. Mind. 1905;14:479–493.
[17] Twardowski K. On the Content and Object of Presentations. Martinus Nijhoff; 1977.
[18] Tarski A. Logic, Semantics, Metamathematics: Papers from 1923 to 1938. Clarendon Press; 1956.
[19] Loebe F. Ontological Semantics: An Attempt at Foundations of Ontology Representation [PhD Thesis]. University of Leipzig, Germany; 2015. http://nbn-resolving.de/urn:nbn:de:bsz:15-qucosa-166326.
[20] Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nature Biotechnology. 2007;25(11): 1251–1255.
[21] Feyerabend P. Against Method. New Left Books; 1975.
[22] Seibt J. General Processes [Habilitation Thesis]. University of Konstanz, Germany; 2004.
[23] Stell JG, West M. A four-dimensionalist mereotopology. In: FOIS 2004. IOS Press; 2004. p. 261–272.
[24] Arp R, Smith B, Spear AD. Building Ontologies with Basic Formal Ontology. MIT Press; 2015.
[25] Borgo S, Masolo C. Ontological Foundations of DOLCE. In: Theory and Applications of Ontology: Computer Applications. Springer; 2010. p. 279–295.
[26] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. WonderWeb Deliverable D18. LOA; 2003.
[27] Guizzardi G, Wagner G. Using the Unified Foundational Ontology (UFO) as a Foundation for General Conceptual Modeling Languages. In: Theory and Applications of Ontology: Computer Applications. Springer; 2010. p. 175–196.
[28] Guizzardi G, Herre H, Wagner G. On the General Ontological Foundations of Conceptual Modeling. In: ER 2002. LNCS 2503. Springer; 2002. p. 65–78.
[29] Guarino N, Welty C. An Overview of OntoClean. In: HB on Ontologies. Springer; 2004. p. 151–159.
[30] Stuckenschmidt H, Parent C, Spaccapietra S, editors. Modular Ontologies. LNCS 5445. Springer; 2009.
[31] Loebe F. Organization and Management of Large Categorical Systems. In: Theory and Applications of Ontology: Computer Applications. Springer; 2010. p. 67–100.
[32] Arapinis A, Vieu L. A plea for complex categories in ontologies. Applied Ontol. 2015;10(3-4):285–296.
[33] Herre H, Loebe F. A Meta-ontological Architecture for Foundational Ontologies. In: OTM/ODBASE 2005 (Part II). LNCS 3761. Springer; 2005. p. 1398–1415.
[34] Gracia JJE. Metaphysics and Its Tasks. State University of New York Press; 1999.
[35] Hoehndorf R, Loebe F, Poli R, Herre H, Kelso J. GFO-Bio. Applied Ontology. 2008;3(4):219–227.
[36] Uciteli A, Groß S, Kireyev S, Herre H. An ontologically founded architecture for information systems in clinical and epidemiological research. Journal of Biomedical Semantics. 2011;2(Suppl 4):S1.1–22.
[37] Hobbs J, Pan F. Time Ontology in OWL. Recommendation. W3C; 2006.
[38] Siemoleit S, Uciteli A, Bieck R, Herre H. Processual Reasoning over Sequences of Situations in Endoscopic Surgery. In: GMDS 2017. IOS Press; 2017. p. 222–226.
[39] Burek P, Scherf N, Herre H. A pattern-based approach to a cell tracking ontology. Procedia Computer Science. 2019;159:784–793.
[40] Tegtmeier E. Grundzüge einer kategorialen Ontologie. Alber Verlag; 1992.
[41] Siemoleit S, Uciteli A, Bieck R, Herre H. Ontological Modelling of Situational Awareness in Surgical Interventions. In: JOWO 2017. CEUR 2050. CEUR-WS.org; 2017.
[42] Burek P, Scherf N, Herre H. OWL Patterns for Modeling the Change over Time exemplified by the Cell Tracking Ontology. In: ODLS 2014. University of Leipzig, Germany; 2014.
[43] Uciteli A, Beger C, Kirsten T, Meineke FA, Herre H. Ontological Modelling and Reasoning of Phenotypes. In: JOWO 2019. CEUR 2518. CEUR-WS.org; 2019.
[44] Loebe F. Abstract vs. social roles. Applied Ontology. 2007;2(2):127–158.
[45] Loebe F. Towards an Ontology of Categories and Relations. In: JOWO 2018. CEUR 2205. CEUR-WS.org; 2018.
[46] Suh NP. Axiomatic design: advances and applications. Oxford University Press, USA; 2001.
[47] Burek P, Herre H, Loebe F. Ontological Analysis of Functional Decomposition. In: SOMET 2009. IOS Press; 2009. p. 428–439.
[48] Burek P, Loebe F, Herre H. FueL: Representing function structure and function dependencies with a UML profile for function modeling. Applied Ontology. 2016;11(2):155–203.
[49] Burek P, Loebe F, Herre H. Towards refactoring the Molecular Function Ontology with a UML profile for function modeling. Journal of Biomedical Semantics. 2017;8(1):48.
[50] Burek P, Hoehndorf R, Loebe F, Visagie J, Herre H, Kelso J. A top-level ontology of functions and its application in the Open Biomedical Ontologies. Bioinformatics. 2006;22(14):e66–e73.
[51] Hoehndorf R, Ngonga Ngomo AC, Kelso J. Applying the functional abnormality ontology pattern to anatomical functions. Journal of Biomedical Semantics. 2010;1(1):4.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200659

An Analysis of Commitments in Ontology Language Design

Pablo R. FILLOTTRANI [a,b] and C. Maria KEET [c,1]
[a] *Department of Computer Science and Engineering, Universidad Nacional del Sur, Argentina*
[b] *Comisión de Investigaciones Científicas de la Provincia de Buenos Aires, Argentina*
[c] *Department of Computer Science, University of Cape Town, South Africa*

Abstract. Multiple ontology languages have been developed over the years, which brings afore two key components: how to select the appropriate language for the task at hand and language design itself. This engineering step entails examining the ontological 'commitments' embedded into the language, which, in turn, demands for an insight into what the effects of philosophical viewpoints may be on the design of a representation language. But what are the sort of commitments one should be able to choose from that have an underlying philosophical point of view, and which philosophical stances have a knock-on effect on the specification or selection of an ontology language? In this paper, we provide a first step towards answering these questions. We identify and analyse ontological commitments embedded in logics, or that could be, and show that they have been taken in well-known ontology languages. This contributes to reflecting on the language as enabler or inhibitor to formally characterising an ontology or an ontological investigation, as well as the design of new ontology languages following the proposed design process.

Keywords. Ontology Language Design, Ontology Engineering, Conceptual Modelling, Ontological Foundations

1. Introduction

Ontology engineering aims to study and develop methodologies, tools, and languages that support building ontologies to be used in information systems [1,2,3]. Determining which formal ontology language to use to represent the intended meaning is, or ought to be, a key decision in this process [4]. This because, firstly, it helps in understanding the nuances of the interpretation, facilitates addressing possible ambiguities, and enables capturing the intended meaning as precise as possible. Secondly, the ontology language interacts with the tools so that such semantic descriptions can be automatically processed, including services such as consistency checking, and be integrated with other ontologies or become part of an information system.

Several ontology languages have been used for this purpose, which were either forked from previous knowledge representation languages and their applications or specifically designed for ontologies. This brings afore two key components: selecting the

[1]Corresponding Author E-mail: mkeet@cs.uct.ac.za

appropriate language for the task at hand and language design itself. Making the right decision for language design or its use requires gathering information, specifying requirements, identifying alternative solutions, and weighting pros and cons. This entails one has to examine the 'commitments' each language has embedded in it, so as to make it clear what the choices for existing languages are and to eventually justify a possible need for designing a new ontology language. Examples of philosophical commitments are whether the language should have a separate type of element for attributes (e.g., alike the OWL data properties), whether parthood should be a primitive, and whether the world is crisp or inherently vague. Such choices in the design of an ontology language, or which have been made for the modeller already, are fundamental in the way the language enables or obstructs representing reality. Ignorance of limitations assures that no change towards proper correction is possible. Two common examples that have been investigated and shown to affect modelling ability and precision are the improved understanding by the modeller with look-here vs. look-across syntax notation for n-ary relationships [5] and the increased use and disambiguation of part-whole relations when it is a primitive in the language [6].

This raises the question: *what are the sort of commitments one should be able to choose from in language design or selection, which have an underlying philosophical point of view?* And, taking a step back: *which philosophical stances have a knock-on effect on the specification or selection of an ontology language?* In this paper, we provide an initial step towards answering these questions. We identify and analyse ontological commitments embedded in logics, or that could be, and show that they have been taken in well-known ontology languages. They include, among others, crisp vs. vague, 3- or 4-dimensionalist, the 'fundamental furniture' (basic building blocks, also called "epistemological primitives" [7]) and typical possible refinements, and a logic's interaction with natural language. To position this in an overall engineering process and take a step to make it actionable for ontologists, we framed it in a process of language design and devised a requirements catalogue aimed to help identify or select commitments for a language. We hope that this will provide the ontology engineer with better means to decide the language to represent an ontology, design a new one, or at least be aware of the commitments they have selected.

We start by describing the steps in the process of language design in Section 2, in order to acknowledge the different stages that are involved, since they also interact and affect each other. Then in Section 3, we identify the choices that are made and analyse their consequences. In Section 4 we assess several key ontology languages in view of these commitments, and present a use case of the the application of the design process for a new ontology language. Finally, we present some conclusions in Section 5.

2. Language Design for Ontologies

The design of a logic (representation language) to develop ontologies or to formally characterise the ontological nature of the topic under investigation may be seen as an engineering task, just like the development of an artefact. That is, a process with several steps from start to finish, such as requirements engineering and testing. For instance, requirements may be that the logic has to be decidable in subsumption reasoning, that it needs to have n-ary relationships (with $n \geq 2$), and that it must have support for modularity;

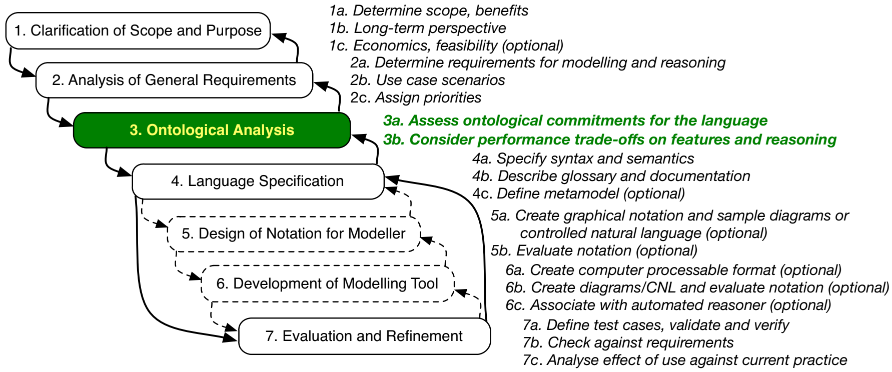


Figure 1. Development steps for the creation of a logic for ontologies, where the focus of this paper is highlighted in bold and shaded (green); the two dashed stages are optional, depending on what has been specified as scope in step 1.

see, e.g., the set of requirements and goals for OWL [8] and the use cases and feature set of DOL [9]. To the best of our knowledge, there is no existing methodology to design a logic. A design pipeline does exist for domain-specific languages (DSLs), notably as outlined by Frank [10], and there are a few works that focus on one aspect of the process, such as the requirements engineering step [11], or that propose guidelines tailored to one class of modelling language specifically (e.g., DSLs [12]). For the purpose of designing a logic for ontologies in particular, we modified Frank's "macro-process" (an iterative waterfall) for DSLs in the way as shown in Fig. 1, with the specific changes being a modified requirements step (mainly also including reasoning), addition of "3. Ontological Analysis", having made steps 5 and 6 optional, and including CNLs as optional syntactic sugar cf. only diagrammatic notations. Acknowledging that in practice the final version of the language typically is not obtained in a single sequential pass through these steps, we added the possibility of backtracking between each step and the previous one, and between Step 7 and Step 4. The highlighted Step 3 is the focus of this paper, which we shall turn to after illustrating briefly how each step applies to ontology language design.

For step 1, the *scope* is ontology languages (rather than, say, DSLs, Business Process Modeling languages etc.), whereas the *goals* and *purpose* may already vary for the case at hand. An example of a scope specification can be found in the DOL standard [9]. A typical goal for ontologies may be *to have the necessary constructs to be able to represent that what needs to be formalised as precise as possible* in order to match the intended models as closely as possible with the models that the theory admits, to assist with developing 'good' ontologies [13]. In most ontology research (cf. their applications in industry), economics does not play a role, whereas a long-term perspective is assumed.

For step 2, Frank refers to a "requirements catalogue" to consult and choose one's requirements from [10], which does not exist for ontology languages. Perhaps a 'modelling features on offer' list, alike in the appendix of the Description Logics handbook [14], could be seen as contributing toward a catalogue, but a features-on-offer list is distinct from a broad requirements list. Broader requirements for a logic may encompass not just the modelling features (e.g., qualified cardinality, transitivity, etc.), but also concern the language as a whole, such as *must be usable in a multilingual setting*, and its context, such as *must be able to work with ontology modules*; examples of sets of requirements are those for OWL [8] and CL [15, section 5]. The requirements for automated reasoning are, typically, inconsistency checking, subsumption reasoning, instance checking, and querying the ontology and data in ontology-based data access. It may be that one reasoning service is deemed more important than the others, hence, priorities can be, or have been, assigned (item 2c) implicitly or explicitly. This is the case with the OWL family: for the OWL profiles [16], querying instances was deemed much more important for the OWL 2 QL developers, and size trumped inconsistency checking for the OWL 2 EL developers. A few trade-off tables are also available, such as for logics for conceptual models [17] and for representing various fragments of the KGEMT mereotopology in multiple logics [18]. There are few explicit use cases designed to inform ontology language development beyond the toy examples for a particular features (e.g., 'need to be able to model narcissist' to motivate reflexive object properties [19]), with the exception of the 12 use cases for DOL [9].

Step 3 may seem closely related to both requirements and language design, so why then a separate step? It turns out that the jump from requirements to language design leaves implicit many decisions of an ontological nature, where meeting a requirement entails committing to one philosophical stance or another. This may have to be deliberated first, or else at least be a conscious decision to document. For instance, if there is a requirement *track the entities through time*, then does that mean 3D objects+time or 4D objects, and if the former, does that have to come with discrete or dense time, and with linear time or trees, or if the requirement states *with attributes*, then does that mean attributions with a universalist stance or one with tropes, and as separate element in the language to support the representation of either, or not? More generally, this raises the question *what is the ontological analysis to conduct, to the extent that it will have a bearing on the logic?* We shall delve into answering this question in the next section.

The language specification step (no. 4) is comparatively well-known, with Description Logics probably the most popular by number of papers, and OWL by number of ontologies represented in that language. Most of these logics do not have a glossary, documentation (beyond the scientific paper), or a metamodel. Regarding the latter, this is likely thanks to the formal semantics (which is different for DSLs), although the UML diagrams in the OWL standard [20] amount to a metamodel. The DOL and CL standards do include a "terms and definitions" section, functioning as a glossary [9,15].

Steps 5 and 6 in the pipeline are optional, since an ontology language may well be a fully paper-based logic without any syntactic sugar or tooling support. Some logics for ontologies do have such interfaces and tools. For instance, there are multiple controlled natural language (CNL) interfaces to several OWL fragments, as well as graphical renderers, and a relatively comprehensive tooling infrastructure for manipulating and reasoning over OWL files. There are a few tools for Common Logic [15] and OBO [21].

The last step in the process is to evaluate the artefact, being the logic for representing the ontology: does it meet the requirements? Does it solve the problems described in the use cases? Does it adhere to the priorities? The answers probably will not be a threefold "yes", but with a systematic procedure in place, it should at least have become clear where concessions have been made, how, and why, which at least contributes to an understanding of why that logic (ontology language) is the way it is.

3. Design choices for ontology languages

We will now zoom in on Step 3 of the design process (recall Fig. 1).

Assess ontological commitments for the language (Step 3a of the process) While the seminal paper on ontology-driven information systems [13] did note *ontological commitment* as "intensional interpretation" in addition to the standard extensional one in a model-theoretic semantics, it did not consider the affordances and features of the logic. The latter entails two components: 1) the ability to represent the conceptualisation more or less precisely with more or less constraints[2], and 2) whether the representation language contributes to support, or even shape, the conceptualisation and one's ontological analysis for the ontology, or embeds certain philosophical assumptions and positions. Regarding the latter, we identified four key decision points that each have multiple sub-questions and decisions each, which may not yet be exhaustive, but it is the first and most comprehensive collection to date for this approach. They are drawn from related work and our own, and are elaborated on afterward.

1. What belongs to the 'fundamental furniture of the universe', or: what are the 'epistemological primitives'? From a logic viewpoint: what are the building blocks in the logic? This includes answering questions such as:
    (a) Does the world have an abundance of 3-dimensional objects (with an optional time dimension), or are there 4-dimensional space-time worms and thus a language catering for that? Related to that: is the world made up of processes, actions, etc. that static entities participate in or are there static entities that may, or may not, participate in this dynamism?
    (b) Do two or more elements relate directly, or through the roles they play in the relation?
    (c) Related to, or perhaps underlying, the former two items: is there a prioritisation among the primitives, some being more or less relevant, and does that affect the notion of 'fundamental'?
2. Should one refine the kinds of general elements (from item 1) and promote them to have their own representation element in the logic, to possibly result in a different (better) ontology? For instance,
    (a) Refining Relationship with an pre-defined element for parthood;
    (b) Refining Class (or concept or universal) with, e.g., stereotypes or a many-sorted logic to indicate ontological distinctions between the kind of entities, such as between a rigid and non-rigid entities, or between sortals, quasi-types, and attributions etc;
    (c) Setting the arity of the relationship (or n-ary predicate with $n \geq 2$): if only binary relationships are allowed, then the modeller may assume there are only binary relations in the world, reifications of n-aries vs the existence of n-aries ($n \geq 2$) proper, and fixed arities vs. relationships with variable arity;
3. Should the logic be intertwined with natural language, or is natural language a layer 'on top of' the logic and thus separable from (and perhaps even independent of) the core knowledge or ontology? Related to this: What must be named?

[2] this is different from subject domain coverage; to be able to represent, say, "has part =2 legs" vs only "has part $\geq$ 1 legs" concerns *precision*, whereas omitting information about the legs concerns *coverage*.

4. Is the world crisp in the sense of something either being true or false, or may an entity be something to a degree and the world is thus inherently vague?

There is scant research on the effects of choosing one option over the other, with one notable observation in the related area of conceptual modelling: binaries vs. n-aries (Item 2c) and just plain relationship vs. also with aggregation (roughly parthood, Item 2a) do indeed make a difference at least for UML vs ER and ORM: UML class diagrams have significantly more aggregation associations than ER and ORM diagrams have, yet fewer n-aries [6]. The former is attributed to it being a separate element and the latter at least partially because there are obstructions to draw n-aries and to understand them due to its graphical notation [5], compared to ER and ORM that use the same notation for both binaries and n-aries and have no primitive for parthood. Parthood also features prominently in ontologies represented in the OBO format, where it was a primitive [22], whereas noting that absence of such a primitive in OWL might be an explanation for the well-known is-a/part-of confusion by novice ontologists. There is a 25-year old proposal to include parthood as a primitive in DLs [23], but this has never been pursued further. The lack of n-ary relationships—i.e., where $n \geq 2$ rather than just $n = 2$—in OWL has been a long-standing complaint, since it is possible to have them in DLs, notably the $\mathcal{DLR}$ [24] and $\mathcal{CFD}$ [25] families, whilst modellers are facing workarounds with ontology design patterns to approximate it by means of a reification with partial constraints. The n-aries problems do not exist in full FOL or HOL, but they do not have a primitive for parthood, since there there are just n-ary predicates ($n \geq 1$), not predicates + parthood, in the definition of the language. Surely, one can define a 'FOL with parthood', but that is different from the regular definition of full first-order predicate logic.

We will elaborate on the remainder of the items in the following paragraphs.

On the fundamental furniture of the universe (Item 1) Practically: what are the building blocks in the logic? More principled with respect to ontology, this includes assessing:
– Are there just predicates with $n \geq 1$, or are there entity types that are (necessarily) related by means of n-ary relationships (with $n \geq 2$), where that distinction between unaries and n-aries is ontologically meaningful?
– Are the roles that objects play in relation(ship) fundamental components, and therewith that a relationship does not have a directionality, no inverses, and is so-called *positionalist* (cf *standard view*)? (rewording of Item 1b)
– Within the predominant 3D scope: are stuffs distinct from objects, are they types of objects, or do they not exist?

Let us discuss Item 1b first, since different decisions have been taken. As illustration, assume there is some binary relationship called teach that holds between Professor and Course. In the "standard view" [26], there would be at least one predicate, teaches (or taught by), in which Professor and Course participate *in that specific order* (or in the reverse, respectively). The 'there are roles too'-option ("positionalism" [26]) would argue that Professor plays a role, say, [lecturer], in the relationship teach and Course plays the role [subject] in the relationship teach. The relationship has no 'direction', as the standard view has, and the roles thus do not have an ordering, since the participation is clear from the assignment of the object to a role in the relationship. Objects always play a role in the relations they participate in, and inherently so; hence, *role* will have to be an element in the language. Besides the philosophical arguments, positionalism is also deemed better for natural language interaction and expressing more types of constraints than with standard view relationships [26,27,28,29], therewith contributing to more precise representations of the universe of discourse. Most DLs, OWL, FOL, and HOL adhere to a standard view commitment, whereas the main conceptual data modelling languages (EER, UML, and ORM) and the $\mathcal{DLR}$ family of DLs is positionalist (having so-called DL role components). There is also an anti-positionalist stance, which is argued to be even better than positionalist [26,29] and a language specific for the anti-positionalist commitment is proposed in [30].

Stuffs are generally assumed to be distinct from countable objects (third item, above), yet there is no uniform approach to deal with it because it is not agreed upon how distinct they are. Are they different categories of things, or are they both universals or particulars, or is a stuff (or its particular amount of matter) just another type of endurant, or is each amount of matter an object? The answer may affect the ontology language. For instance, Donnelly and Bittner use a many-sorted logic for the portions to distinguish it from objects with parts [31], stereotyping in UML [32] or formalising it in a HOL [33], or merely a subclass of endurant [34] and therewith thus would not merit a separate element in the logic. Currently, most logics do not make that distinction.

3D vs. 4D (Item 1a) 3-dimensionalism assumes there are objects in space where the objects are wholly present at each point in time (i.e., do not have temporal parts) and statements are true at the 'present', whilst being ignorant of the object in the past and future. Time can be added as an orthogonal dimension, as in, e.g., $\mathcal{DLR}_{\mathcal{US}}$ [35], for which there are further choices, notably linear vs trees and dense vs discrete time. 4-dimensionalism, sometimes also called 'fluents', assumes entities exist in four dimensions, being in spacetime, entities unfold in time, and thus do have temporal parts, and statements can be about not only the present, but also the past and future; examples for ontologies and ontology-driven modelling include [36,37]. What does this mean for representing knowledge? Let us take as example a holding or supra-organisation [37], such as Alphabet and Nestlé: these companies exist for some time and keep their identity all the while they acquire and sell other (subsidiary) companies. In a 3D-only representation where there is only an atemporal 'current' snapshot, one would have a record of which companies they own now, but not whether they are the same ones as last year.

The predominant choice made by developers of popular ontology languages is 3D objects with optional temporal extensions. We could not find any scientific evidence that demonstrates explain why this is the case. From a computational viewpoint, the temporal extension is expensive, hence, is prohibitive for designing scalable systems.

Refining the core elements or not (Item 2) Principled ideas for refining the unaries (concepts, classes, universals) avail of notions such as rigidity that then leads to types of unaries [38], such as "type" for the kind of objects that supply an identity criterion (e.g., Person) and "role" that an object plays (e.g., Professor), as examples for Item 2b. Like aforementioned stuff, they could be defined in a HOL, or used in declaring a many-sorted logic, be it based on such metaproperties or, say, by taking the main entities from a foundational ontology.

Another possible refinement of core elements may be the aforementioned parthood relation (Item 2a), and one could do likewise with other common relations, such as participation, causation, and constitution. A refinement of binary relationship is *attribute* (OWL data property), which holds between a class and a data type. It is debatable whether attributes with their data types belong in an ontology language. For instance, KL-ONE is clear in stating that they are *not* among the "epistemological primitives" for they have "no semantically justifiable place in the epistemology" and therefore do not belong in the logic [7]. Common Logic does not have attributes either, but one can specify its inclusion through an extension [15]. Similarly, it is absent from most DLs, yet it made it into the OWL specification [20]. There are indeed no data types, hence, no attributes, in a world without humans having conceived of the construct. Whether an ontology and its language are permitted to contain 'unnatural' things is debatable. One also may argue that an attribution or quality such as length means the same thing regardless whether it is used for the length of, say, a Sofa or a Table and thus would be one thing, so rather than to split up into two or more types of attributes, alike $\mathsf{lengthS} \mapsto \mathsf{Sofa} \times \mathtt{Integer}$, $\mathsf{lengthT} \mapsto \mathsf{Table} \times \mathtt{Float}$, and $\mathsf{lengthT}' \mapsto \mathsf{Table} \times \mathtt{Int}$, it should be one datatype-independent property; hence, an argument from ontological parsimony vs abundance is also possible.

Another refinement is implication—assuming implication is a core notion. In FOL and HOL, there is only implication, but it has been argued that not all implications are the same. In particular, there is the notion of *inheritance* for classes/concepts/universals (unaries) and *subsetting* and *subtyping* for relationships (n-aries, with $n \geq 2$). For instance, while the DL axiom $\mathsf{Cat} \sqsubseteq \mathsf{Animal}$ gets translated into FOL as $\forall \mathsf{x}(\mathsf{Cat}(\mathsf{x}) \rightarrow \mathsf{Animal}(\mathsf{x}))$, the former has the embedded notion of *property inheritance along the taxonomy* where the properties of Animal are inherited by Cat. Logics with a syntactic sugar communicates this differently to modellers and domain experts yet again. For, instance, the aforementioned is verbalised as 'Each Cat is an Animal', whereas, say, $\forall \mathsf{x}(\mathsf{Cat}(\mathsf{x}) \rightarrow \exists \mathsf{y}(\mathsf{eats}(\mathsf{x},\mathsf{y}) \land \mathsf{Mouse}(\mathsf{y})))$ is not verbalised as 'Each Cat is a something that eats at least one Mouse' but as 'Each Cat eats at least one Mouse', in that there's a relationship eats between two entities, on par with each other, not the notion of subsumption or inheritance between an entity and a relation it participates in.

Naming things (Item 3) Naming something involves identifying it and acknowledging its relevance; conversely, the nameless may be redundant, irrelevant, or non-existent. The process of naming something involves the interaction between natural language and ontology. There are millennia-old philosophical debates on that interaction. Naming elements may come before or after determining the 'fundamental furniture', and differ by ontology language, as illustrated in Table 1; observe that none of the languages names all six types of elements. Moreover, those elements are given names, such as Prof and teaches, embedding natural language into the logic. There are alternatives to this approach. OBO uses identifiers for concept names, which each have one or more labels for the natural language name of that natural language-independent entity. This approach can be ported into the OWL world where, e.g., the Protégé tool then renders the labels in the interface. An attempt to systematically address that natural language $\leftrightarrow$ ontology interaction has been proposed in the context of the Semantic Web, where the natural language dimension has its own extension on top of an OWL file [39], by means of a W3C community-based annotation scheme [40]. For the related conceptual modelling languages, it is worth noting that ORM diagrams commonly have *reading labels* for a fact type (relationship), where modellers hardly add names for the roles or the fact type themselves [6] (modelling tools add those automatically in the serialisation), whereas for UML Class diagrams, association end/memberEnd (i.e., role) names are expected, but not association (relationship) names, as also illustrated in the UML standard.

Table 1. Terminology of some of the logics for ontologies (relevant selection).

General term	**KL-ONE**	**OWL**	**FOL**	**CL**
Relationship	RoleSet	Object Property	Predicate	Name
Role	Link	n/a	n/a	n/a
Entity type	Concept	Class	Predicate	Name
Attribute	n/a	Data Property	n/a	n/a
Datatype	n/a	Datatype	n/a	n/a
Function	n/a	n/a	Function	Functional term (that is a Name)

A crisp world or not (Item 4) While the debate on concepts vs universals have received ample attention, they both assume a crisp world: they have clear boundaries, specified by a set of properties of the universal or concept, and where some instance is either an instance of that universal or concept, or not. There are alternatives. In cognition and learning, the notions of *prototypes* and *exemplars* are well-known [41]. Prototypes have fuzzy boundaries since some object fits a category more or less well, and some exemplars may be better or not that good: they are more or less of a member of that set. For instance, Penguin is not a prototypical bird, or not a good exemplar of it, compared to, say, Raven. Also, entity types may have a rough boundary: it is clear what is in and what is out, but membership cannot be established around the border, be it not ever or not with the knowledge or data at one's disposal; i.e., the notion of a *rough set* [42]. Further, recent years has seen widespread use of statistics, such as in machine learning and deep learning, where non-crisp models are built. It remains to be seen whether adoption of these techniques entail an underlying difference in philosophical stance or a 'disagree and commit' because of the remarkable outcomes, and how, if at all, such outcomes would be connected to ontologies. Depending on one's philosophical inclination, the logic will either have only true/false, or some way to deal with the vagueness that may be inherent [43,44]. Logics with vagueness have been investigated and some tools and applications exist, including for DLs [45].

Engineering factors (Step 3b of the design process) How do the chosen primitives of Items 1 and 2 relate to the logical symbols to obtain the intended meaning? It is also necessary to analyse the role of the logic. Logic maps explicit extensional primitive elements (facts or relations, for example) into implicit intensional elements through the available logical symbols and inference rules. This mapping raises the point of determining which logical symbols and inference rules are adequate, which have to be together with the well known trade-off between *expressivity* and *efficiency* [46]. For instance, instead of mere n-ary predicates with $n \geq 1$, as in FOL, one could design a logic with a syntax where unaries denote universals, indicated with, e.g., camel case notations in serif (e.g., a universal Meerkat) and n-aries with $n \geq 2$ as relational properties denoted in the syntax with italics serif in all lowercase (e.g., *eating*), and roles in square brackets (e.g., [prey]), to enable declaring that there are eating events where meerkats are the prey of jackals, in a positionalist universe and underlying sets for semantics: $\exists \mathit{eating} \Rightarrow [\mathsf{prey}]\mathsf{Meerkat} \times [\mathsf{predator}]\mathsf{Jackal}$. Another example is the temporal operators in a temporal logic, such as 'until' U and 'at all times' $\Box$: while it indeed can be reconstructed in FOL, it is deemed essential so as to merit to be explicitly available in the language. One also could add alternative symbols to distinguish that generic 'mere' implication with subsumption between unaries, as alluded to with the Cat and Mouse, example, above; e.g., using different notation alike, say, a Cat $\Rightarrow$ Animal vs. a Cat $\rightarrow \exists$eats.Mouse. Such considerations forces a designer to indeed have taken a stance on Item 1c of the ontological commitments and deal with its consequences in this step b. Non-logical symbols, such as syntactic sugar, may afterwards be added but are not in the focus of the logical language; e.g., typical natural language renderings of "$\exists$" are 'there are', 'at least one', and 'some'.

Besides ontological commitments, there may be practical considerations in designing the representation language. These are mainly concerned with automated reasoning: 1) which reasoning services are needed, 2) how scalable it has to be, if at all, and 3) whether and to what extent can one avail of existing tooling infrastructure. Examining this step is not the focus of this paper.

4. Evaluation

We apply and evaluate the language design and ontological analysis step in two ways. First, we assess several key ontology languages that were, and are, used throughout the decades. Second, we step through the design process to see how it may work for a hypothetical 'FK Ontology Language', *FKOL*, that should meet our *ad hoc* requirements.

4.1. Assessment of popular ontology languages

The logics for the assessment were selected for the following reasons. OBO [21] has been widely popular with bio-ontologies since it use since 1998 for the Gene Ontology [22] and it helped popularising the use of lightweight domain ontologies especially in science. SKOS is also popular, notably for bringing thesauri and similar vocabularies into the Semantic Web [47] and for its subsequent use in bottom-up ontology development. KL-ONE is essentially the predecessor to DL and, unlike the DL documentation, did try to express its rationale for the design of the language [7]. $\mathcal{DLR}_{ifd}$ [48] is an unusual DL in that it has features that are commonly claimed as not doable and it is deemed the best fit for a logic-based reconstruction of conceptual data modelling languages such as UML Class Diagrams. OWL 2 DL [20] is included in the comparison because it has popularised ontologies in government and industry and was standardised by the W3C. Traditional FOL is included because it is a common logic foundation theoretically at least. HOL was added because it is the most expressive family of logics.

Their comparisons on ontological commitments and computational features are included in Tables 2 and 3, respectively. In Table 2, a feature is considered primitive ('yes') when there is a corresponding logical symbol in the language; it is 'possible' if there is a direct and simple reconstruction of the meaning of the feature in the language without altering the essential properties of the language (in terms of those defined in step 3). For example, subsumption is possible to be defined in terms of FOL primitives; in contrast, fuzzy extensions of OWL are possible, but they change significantly the designed properties of the language, so we deem this a 'no'. If the feature is not completely represented in the language, it is 'partial'; e.g., $\sqsubseteq$ in DLs does have property inheritance in the semantics for class subsumption, but there are no different symbols for properties of classes as illustrated with the Cat, Animal, and Mouse above.

FOL and HOL turned out to have the same answers in Table 2 and therefore were merged into one column; SKOS is not used for automated reasoning and therefore omit-

Table 2. Selection of logics for ontologies and vocabularies and their ontological commitments and features. 3D/4D: 3/4-dimensionalism; NL: natural language

Feature	OBO	SKOS	KL-ONE	$\mathcal{DLR}_{ifd}$	OWL2DL	FOL/HOL
Standard view	yes	yes	yes, with some positionalist	positionalist	yes	yes
3- or 4-dimensionalism	3D	3D	3D	3D	3D	3D or 4D
NL$\leftrightarrow$logic separation	yes	possible	no	no	possible	no
Type(s) of unaries	class	concept	generic, primitive, defined, and individual concepts	concept	class	unary predicate
Concept subsumption as primitive	yes	no‡	yes	partial	partial	possible
Parthood as primitive	yes*	possible	no	no	no	no
Relationship subsetting and sub-typing as primitives	no	subsetting possible	yes	subsetting	subsetting	no
Relationship definition	no	no	composition	no	limited chains	yes
n-aries (where $n \geq 3$)	no	no	no	yes	no	yes
Attributes with datatypes	no	partial	no	yes	yes	no
Functions or procedures	no	no	procedures may be attached	no	no	functions
Multi-valued (including fuzzy, rough)	no	no	no	no	no	possible
Semantics	graph	none / variable	variable	model-theoretic	model-theoretic	model-theoretic
Open World Assumption	no†	no†	yes	yes	yes	yes
Unique Name Assumption	yes†	yes	yes	no	no	no

* depends on the OBO file format version: v1.4 does not have it so as to be compatible with OWL.
† not explicitly stated in the documentation to be the case, but likely given other information.
‡ there is `skos:broader` and `skos:narrower`, but they do not equate sub-/super-concept.

ted from Table 3. As can be seen from the values in the columns, there are varied ontological and computational commitments, except for a clear predominance of a 3-dimensionalist stance and crisp logics, and to some extent also Open World Assumption, parthood not as primitive, and on not scoring well on scalability; or: lacking 4-dimensionalism, uncertainty, CWA, parthood, and scalability. Why this is so is a question that an experimental philosopher may wish to investigate.

4.2. Designing one's preferred language

Commencing the design process with *Step 1*, the scope is 'use case' of which the benefits are the evaluation of the applicability of the design approach to ontology language design. Then, to determine the requirements of our fictitious *FKOL* language (*Step 2*), we need to avail of a requirements catalogue. Since it did not exist, we first created a draft catalogue by combining the published requirements from OWL [8] and CL [15], added the ontology-motivated ones, and a few more that seemed possibly relevant for

Table 3. Selection of logics for ontologies and their computational features.

Feature	OBO	KL-ONE	$\mathcal{DLR}_{ifd}$	OWL2 DL	FOL	HOL
Complexity (sat., subsumtions)	depends*	undecidable	ExpTime-complete	N2ExpTime-complete	undecidable	highly undecidable
Scalability	yes	not scalable	± scalable TBox	± scalable TBox	not scalable	not scalable
Serialisation	yes	no current one	no	RDF/XML	yes, e.g., CLIF	yes, e.g., Isabelle
Reasoner	no	no current one	no	multiple	few	very few
Interoperability	yes	no	no	yes, ample	little	very little
Usable with DOL	no	n/a	no	yes	yes	yes
Modular ontologies	no	no	no	yes	possible with DOL	possible with DOL

* depends on the version and format (whether as graph or as stored in a database).

such a catalogue. This preliminary catalogue of 56 possible requirements is available online and open for comments and extensions[3]. After establishing the catalogue, the authors independently chose a subset of features they like most or would be most interesting to experiment with, which were then combined into a joint list. The joint list did not have conflicting requirements and is included in Fig. 2. Use case scenarios include specification of the subject domain knowledge that it has to be able to represent, such as that all canonical humans have exactly two legs as part (Case1), and crisp subsumption reasoning and classification, in that it should be able to compute that, e.g., elephants are herbivores (Case2). Due to space limitations, we assign equal priority to each selected requirement.

With respect to the ontological analysis (*Step 3*), the following. The reason for positionalism (O-5 in the requirements catalogue) is a combination of several arguments: 1) the assumption that when things relate, there is one relation in reality, not as many as humans have words for it, 2) it allows for more detailed constraints, so possibly resulting in a better ontology, and 3) it has a better link with popular conceptual data modelling languages. The reasons for parthood as primitive (O-6) are because of its pervasiveness throughout ontologies and when it is a separate primitive, like the aggregations association in UML class, it increases analysis and use and thus may improve ontology quality. Subsumption (O-6) goes hand-in-hand with the requirement for subsumption reasoning/taxonomic classification (UC-8) and that millennia-old notion. The choice for a crisp world (O-7) is based on the argument that any vagueness can be attributed to language or lack of knowledge, rather an inherent vagueness. One can, of course, easily argue differently, but these are our—i.e., the language designers'—choices for the purpose of the evaluation.

The performance trade-offs (*Step 3b*) for *FKOL* will not be good. If we take parthood to mean including the axioms of Ground Mereology, reasoning over a partonomy would be undecidable due to antisymmetry of parthood, so let us ignore antisymmetry. Then the complexity trade-offs depend largely on the list of constraints in E-2, which are fairly straight-forward and can be obtained with languages that are, at most, ExpTime-complete and probably less, since, say, $\mathcal{ALCQI}$ is ExpTime-complete. The other require-

[3]`https://keet.wordpress.com/2020/04/10/a-draft-requirements-catalogue-for-ontology-languages/`

Req-E Expressiveness/constructs/modelling features:
- E-1 Equipped with basic language elements: n-ary predicates ($n \geq 1$, as classes and relationships), roles, and individuals.
- E-2 Equipped with language features: domain & range axioms and cardinality constraints.

Req-F Features of the language as a whole:
- F-9 Should not make arbitrary assumptions about semantics.
- F-11 Extendable (e.g., regarding adding more axioms to same ontology, add more vocabulary, and/or in the sense of importing other ontologies).
- F-16 Use Unique Name Assumption.

Req-UC Usability by computer:
- UC-1 Be an (identifiable) object on the Web.
- UC-8 Able to be used by tools that can do subsumption reasoning/taxonomic classification.
- UC-9 Able to be used by tools that can detect inconsistency.

Req-HU Human usability:
- HU-5 Such that a modeller is free to invent new names and use them in published content.
- HU-6 Have clearly defined syntactic sugar (CNL or diagrammatic)

Req-I Interaction with outside:
- I-3 Compatible with existing standards (e.g., RDF, OWL, XML, URIs, Unicode).

Req-O Ontological decisions:
- O-1 3-Dimensionalist commitment.
- O-5 Positionalist relations and relationships.
- O-6 Have additional primitives: parthood, subsumption.
- O-7 Statements are either true or false.

Figure 2. Combined requirements selected from the requirements catalogue by the authors, for which a logic would have to be designed.

ments can be met without making the language less well-behaved computationally, but it is actually incomplete, still. Notably, it does not say which semantics should be chosen in F-9 and which existing standards in I-3, but let us assume a model-theoretic semantics and Unicode for now.

Since, to the best of our knowledge, there is no ontology language that meets all these requirements, one would have to proceed to *Step 4*: specifying the syntax and semantics, a glossary, and, optionally, a metamodel. We will not pursue this here, since that easily can take up the space of a paper and the focus is on the process of language design and the ontological analysis for certain choices.

The process finalises with the evaluation of the language defined against the set of requirements (Fig. 2) and devising test cases. The test cases should match with the use case scenarios specified in *Step 2b*. Case1 could be tested by examining whether one can represent exactly that: with parthood as $\preceq$, then, say, $\mathsf{Human} \preceq_{=2} \mathsf{Leg}$ in some made-up DL-like syntax, which the language would allow if it meets requirements E-1 (classes), E-2 (for domain and range declaration), and O-6 (parthood). Case2 tests are more elaborate to realise, testing whether requirements E-1 (classes and relationships), E-2 (to declare properties needed for classification), UC-8 (classification) and O-6 (subsumption) have been implemented.

What this use case evaluation has shown is that it is feasible to step through the process of language design, including ontological analysis. It also illustrates that more research and method development could assist this process further, such as possible dependencies between requirements, and how to systematically match use cases to requirements and to test cases.

5. Conclusions

We have outlined an engineering approach to the design of an ontology language, with a particular emphasis on the ontological commitments that hitherto were typically brushed over in the language design stage. It has put more and less common topics and debates in ontology in a different context for a new way of using them: before the actual investigation and representation of a universe of discourse. These results presented may contribute to reflecting on the language as enabler or inhibitor to formally characterising an ontology or an ontological investigation, and spur research into the effects of the representation language on what is eventually represented in the ontology.

References

[1] Kendall EF, McGuinness DL. Ontology Engineering. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool Publishers; 2019.
[2] Gal A. Ontology Engineering. In: Liu L, Özsu MT, editors. Encyclopedia of Database Systems, Second Edition. Springer; 2018. .
[3] Keet CM. An Introduction to Ontolgy Engineering. vol. 20. College Publications; 2018.
[4] Tudorache T. Ontology engineering: Current state, challenges, and future directions. Semantic Web. 2020;11(1):125–138.
[5] Shoval P, Shiran S. Entity-relationship and object-oriented data modeling—an experimental comparison of design quality. Data and Knowledge Engineering. 1997;21:297–315.
[6] Keet CM, Fillottrani PR. An analysis and characterisation of publicly available conceptual models. In: Proc. of ER'15. vol. 9381 of LNCS. Springer; 2015. p. 585–593. 19-22 Oct, Stockholm, Sweden.
[7] Brachman R, Schmolze J. An overview of the KL-ONE Knowledge Representation System. Cognitive Science. 1985;9:171–216.
[8] Horrocks I, Patel-Schneider PF, van Harmelen F. From SHIQ and RDF to OWL: The making of a web ontology language. Journal of Web Semantics. 2003;1(1):7.
[9] Distributed Ontology, Model, and Specification Language; 2018. Available from: `http://www.omg.org/spec/DOL/`.
[10] Frank U. Domain-Specific Modeling Languages - Requirements Analysis and Design Guidelines. In: Reinhartz-Berger I, et al., editors. Domain Engineering: Product Lines, Conceptual Models, and Languages. Springer; 2013. p. 133–157.
[11] de Kinderen S, Ma Q. Requirements engineering for the design of conceptual modeling languages. Applied Ontology. 2015;10(1):7–24.
[12] Karsai G, Krahn H, Pinkernell C, Rumpe B, Schindler M, Völkel S. Design Guidelines for Domain Specific Languages. In: Proceedings of the 9th OOPSLA Workshop on Domain-Specific Modeling (DSM'09); 2009. Orlando, Florida, USA, October 2009.
[13] Guarino N. Formal Ontology and Information Systems. In: Guarino N, editor. Proceedings of Formal Ontology in Information Systems (FOIS'98). Amsterdam: IOS Press; 1998. p. 3–15.
[14] Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF, editors. The Description Logics Handbook – Theory and Applications. 2nd ed. Cambridge University Press; 2008.
[15] Common Logic (CL): a framework for a family of logic-based languages [Standard]; 2007. Available from: `https://www.iso.org/standard/39175.html`.
[16] Motik B, Grau BC, Horrocks I, Wu Z, Fokoue A, Lutz C. OWL 2 Web Ontology Language Profiles. W3C; 2009. Available from: `http://www.w3.org/TR/owl2-profiles/`.
[17] Fillottrani PR, Keet CM. Evidence-based Languages for Conceptual Data Modelling Profiles. In: Proc. of ADBIS'15. vol. 9282 of LNCS. Springer; 2015. p. 215–229. 8-11 Sept, 2015, Poitiers, France.
[18] Keet CM, Kutz O. Orchestrating a Network of Mereo(Topo)Logical Theories. In: Proceedings of the Knowledge Capture Conference. K-CAP 2017. New York, NY, USA: ACM; 2017. p. 11:1–11:8.
[19] Horrocks I, Kutz O, Sattler U. The Even More Irresistible $\mathcal{SROIQ}$. Proc of KR'06. 2006;p. 452–457.
[20] Motik B, Patel-Schneider PF, Parsia B. OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax. W3C; 2009. Http://www.w3.org/TR/owl2-syntax/.

[21] Mungall C, Ruttenberg A, Horrocks I, Osumi-Sutherland D. OBO Flat File Format 1.4 Syntax and Semantics; 2012. Available from: `http://purl.obolibrary.org/obo/oboformat/spec.html`.

[22] Gene Ontology Consortium. Gene Ontology: tool for the unification of biology. Nature Genetics. 2000;25:25–29.

[23] Artale A, Franconi E, Guarino N. Open Problems for Part-Whole Relations. In: Proc. DL'96. Cambridge, MA: AAAI Press; 1996. p. 70–73.

[24] Calvanese D, De Giacomo G, Lenzerini M. On the decidability of query containment under constraints. In: Proc. of PODS'98; 1998. p. 149–158.

[25] Toman D, Weddell GE. On Adding Inverse Features to the Description Logic $CFD^{\forall}{}_{nc}$. In: 13th Pacific Rim International Conference on Artificial Intelligence (PRICAI'14); 2014. p. 587–599.

[26] Fine K. Neutral Relations. The Philosophical Review. 2000;109(1):1–33.

[27] Keet CM. Positionalism of relations and its consequences for fact-oriented modelling. In: Proc. of ORM'09. vol. 5872 of LNCS. Springer; 2009. p. 735–744. Vilamoura, Portugal, Nov 4-6, 2009.

[28] Keet CM, Chirema T. A model for verbalising relations with roles in multiple languages. In: Blomqvist E, et al., editors. Proc. of EKAW'16. vol. 10024 of LNAI. Springer; 2016. p. 384–399. 19-23 Nov 2016, Bologna, Italy.

[29] Leo J. Modeling relations. Journal of Philosophical Logic. 2008;37:353–385.

[30] Leo J. Coordinate-free logic. The Review of Symbolic Logic. 2016;9(3):522–555.

[31] Donnelly M, Bittner T. Summation relations and portions of stuff. Philosophical Studies. 2009;143:167–185.

[32] Guizzardi G. On the representation of quantities and their parts in conceptual modeling. In: Proc. of FOIS'10. IOS Press; 2010. 11th-14th May 2010, Toronto, Canada.

[33] Keet CM. Relating some stuff to other stuff. In: Blomqvist E, et al., editors. Proc. of EKAW'16. vol. 10024 of LNAI. Springer; 2016. p. 368–383. 19-23 Nov 2016, Bologna, Italy.

[34] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. Ontology Library; 2003. Http://wonderweb.semanticweb.org. WonderWeb Deliverable D18 (ver. 1.0, 31-12-2003).

[35] Artale A, Franconi E, Wolter F, Zakharyaschev M. A temporal description logic for reasoning about conceptual schemas and queries. In: Proc. of JELIA'02. vol. 2424 of LNAI. Springer; 2002. p. 98–110.

[36] Batsakis S, Petrakis E, Tachmazidis I, Antoniou G. Temporal Representation and Reasoning in OWL 2. Semantic Web Journal. 2017;8(6):981–1000.

[37] West M, Partridge C, Lycett M. Enterprise Data Modelling: Developing an Ontology-Based Framework for the Shell Downstream Business. In: Proc. of FOMI'10; 2010. p. 71–84. 14-15 Dec., 2010, Trento, Italy.

[38] Guarino N, Welty C. An Overview of OntoClean. In: Staab S, Studer R, editors. Handbook on Ontologies. Springer Verlag; 2009. p. 201–220.

[39] Buitelaar P, Cimiano P, editors. Towards the Multilingual Semantic Web: Principles, Methods and Applications. Springer; 2014.

[40] Cimiano P, McCrae JP, Buitelaar P. Lexicon Model for Ontologies: Community Report. W3C; 2016. Available from: `https://www.w3.org/2016/05/ontolex/`.

[41] Rosch E. Principles of Categorization. In: Margolis E, Lawrence S, editors. Concepts: Core readings. MIT Press; 1978/1999. p. 186–206.

[42] Pawlak Z, Skowron A. Rudiments of rough sets. Information Sciences. 2007;177(1):3–27.

[43] Prinz J. Vagueness, Language, and Ontology. Electronic Journal of Analytical Philosophy. 1998 Spring;6. Available from: `http://ejap.louisiana.edu/EJAP/1998/prinz98.html`.

[44] Sorensen R. Vagueness. In: Zalta EN, editor. The Stanford Encyclopedia of Philosophy; 2003. Available from: `http://plato.stanford.edu/archives/fall2003/entries/vagueness/`.

[45] Lukasiewicz T, Straccia U. Managing Uncertainty and Vagueness in Description Logics for the Semantic Web. Journal of Web Semantics. 2008;6(4):291–308.

[46] Guarino N, Oberle D, Staab S. What Is An Ontology? In: Staab S, Studer R, editors. Handbook on Ontologies. Springer; 2009. p. 1–17.

[47] Miles A, Bechhofer S. SKOS Simple Knowledge Organization System Reference. World Wide Web Consortium (W3C); 2009. Available from: `http://www.w3.org/TR/skos-reference/`.

[48] Calvanese D, De Giacomo G, Lenzerini M. Identification Constraints and Functional Dependencies in Description Logics. In: Proc. of IJCAI'01. Morgan Kaufmann; 2001. p. 155–160. Seattle, USA, Aug. 4-10, 2001.

II. Social Entities

This page intentionally left blank

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200661

Legal Theories and Judicial Decision-Making: An Ontological Analysis

Cristine GRIFFO [a,1], João Paulo A. ALMEIDA [b] and Giancarlo GUIZZARDI [a]
[a] *Free University of Bozen-Bolzano, Italy*
[b] *Federal University of Espírito Santo, Ufes, Brazil*

Abstract. In this paper, we expose the legal theories underlying two important classes of Legal Core Ontologies and show how these ontologies inherit both limitations and benefits (such as explanatory power) of their underlying theories. We do that with the help of a real case study in which we have normative omission and collision of principles. We use this case study to conduct an ontological analysis of the support for judicial decision-making in LKIF-Core (representing Kelsen's Pure Theory of the Law) and UFO-L (representing Robert Alexy's Theory of Constitutional Rights). We show that UFO-L is able to articulate the semantics of the content of judicial decisions by making explicit the individual's legal positions that are raised in argumentation along a legal process. The same cannot be said of LKIF-Core that is based on the Kelsenian stance and focuses on the representation of general norms (norm types) and subsumption of facts to these norms.

Keywords. Legal Theory, Judicial Decision-Making, Legal Core Ontologies, LKIF-Core, UFO-L

Introduction

Judicial Decision-Making refers to the decision-making process through which judges make legal decisions. These are critical processes given that their outcome can substantially affect the lives of legal agents under a jurisdiction (e.g., people, organizations, countries, and collections thereof).

As shown in [1], there is evidence that the outcome of a judicial decision depends among other factors on the philosophical stance underlying the legal theories informing the decision maker. One of the most common of these stances is *Hans Kelsen's philosophy of Legal Positivism* (also known as *Kelsenism* or *Pure Legal Theory*). Under this view, all we have are the legal norms and decision making is, therefore, reduced to a process of subsuming legal facts to these legal norms that constitute a normative system. However, in practice, there are many practical cases in which we have normative omissions (e.g., there are no rules under which a fact can be subsumed). In these cases, legal positivism is unable to offer informative (i.e., non-trivial) insights.

[1] Corresponding Author: Free University of Bolzano, Piazza Domenicani 3, Bolzano, BZ, 39100, Italy; E-mail: cristine.griffo@unibz.it.

In philosophy of law, there are alternative stances that provide a theoretical support for dealing with such cases. A prominent one is *Robert Alexy's Theory of Constitutional Rights*. Alexy's theory is a relational theory, which by making explicit the legal roles and positions constituting legal relations provide mechanisms for reasoning about the different perspectives of different legal relata involved in specific cases, thus, providing informative guidance for legal operators. For a number of years, the so-called *core ontologies* have been used to support different tasks in Legal Informatics, including, representing the results of judicial decision-making. However, as discussed in [2], most legal core ontologies follow a Kelsenian stance. A representative example of this Kelsenian class is LKIF-Core [3]. An exception to this trend is the UFO-L core ontology [4].

In this paper, we demonstrate how these different legal theories underlying existing core ontologies influence their ability to articulate the results of Judicial Decision-Making. We do that with the help of a real judicial case exemplifying a case of normative omission. In this case, the legal norm-rule, that regulates the granting of leave of absence for dealing with private affairs (LDPA) for public servants, foresees specifically two hypotheses: if one is a stable servant, then they are entitled to the leave; if they are not a stable servant, then they are not entitled to the leave. However, the appeal and subsequent decisions in the court of appeals do not simply apply subsumption based on these two hypotheses. Instead, they invoke a collision of norm-principles to establish additional hypotheses not considered in the posited norm-rule. This is a case that received opposite decisions by different decision-makers of the case.

We illustrate how a representative Kelsenian Ontology (LKIF-Core) can be used to represent the decision-making process of the first judge of the case, and how it fails to provide informative insights about the specifics of the case. Then, we employ the UFO-L to fully represent the decision-making process of the second judge, which explicitly follows the principles of Alexy's theory. This analysis demonstrates the ability of UFO-L to: (i) support and explicitly represent the steps involved in an Alexyan decision-making process; (ii) provide informative insights about these cases falling outside a more traditional Kelsenian view. In the ontological analysis made, we verified that, because of the adopted stance (legal positivism), LKIF focuses instead on the content of general and abstract norms (and is not concerned with the legal positions of agents involved in adjudication). Thus, there is in principle a difference on the phenomena that LKIF and UFO-L address. In addition to that, there is no support for principles in LKIF, which are important to many current cases of judicial reasoning (in particular those involved in the so-called "hard cases"). LKIF only supports rules that classify situations into either *mandated* and *prohibited* (through subsumption). We demonstrate in this paper that UFO-L is able to articulate the semantics of the content of judicial decisions by making explicit the various legal perspectives that are raised in argumentation along a legal process.

The remainder of this paper is organized as follows. In Section 1, we briefly present the Logical Positivism of Hans Kelsen, the core ontology LKIF-Core, and show how the former has influenced the design of latter. *Mutatis Mutandis*, we do the same for Robert Alexy's Theory of Constitutional Rights and UFO-L, thus, showing how the latter is an ontological representation of the former. In Section 2, we describe our case study. In Sections 3 and 4, we then employ each of these two ontologies to represent and explicate the decision-making processes of the judges of the case. In particular, in Section 4, we show the notable difference in insight and details supported by UFO-L in contrast to LKIF. In Section 5, we present our final considerations.

1. Background

For many years, ontologies at different levels of generality have been employed to support conceptual modeling and knowledge representation in the legal domain. In a previous paper [2], we present a systematic mapping of the literature on legal ontologies. The term *Legal Core Ontology*[2] was introduced in the mid-1990s by [7] to define the class of ontologies that establish relevant categories used in law and reflect the main reasoning structure in this field. These ontologies propose the representation of general concepts of law (e.g. legal relation, legal norm) that can be used in many sub domains (e.g. Criminal Law, Civil Law, Constitutional Law). Of the several legal core ontologies reviewed in [2], we highlight the following: Frame-Based Ontology (FBO) [8], Functional Ontology of Law (FOL) [9], Legal Top Ontology [10], Core Legal Ontology (CLO) [11], LKIF-core [12], and the UFO-based ontology of legal relations named UFO-L [4]. Most of the core ontologies are based on Kelsen's theory. A prominent example is LKIF-Core [13]. An exception to this trend is UFO-L, which is based on Alexy's Theory of Constitutional Rights. In the next two subsections, respectively, we present Kelsen's theory and its reflection in LKIF-Core (which we select here as representative of the positivist stance), and Alexy's theory as represented in UFO-L.

1.1. Kelsenism and LKIF-Core

In his work Pure Theory of Law [14], published at the beginning of the 20th century, Kelsen proposes to conceive the law as "a matter of what has been posited (ordered, decided, practiced, tolerated, etc.)" [15] downplaying considerations of political and moral merit. A law in this setting has the form of a conditional order to apply sanctions if a certain prohibited behavior is performed. A legal norm for Kelsen is then formulated as a hypothetical proposition, following the formula: if A, then must be S, where A is hypothetical conduct and S is the sanction that follows the occurrence of hypothesis.

An ontology based on the Kelsenian view has as its ontological commitment to define law as a system of relations between general norms (norm types) given *a priori*. Another important point is the subsumption operation that legally typifies the facts. For example, "Joseph stole John's vehicle" becomes legally relevant when an abstract norm type is found[3] under which this fact can be classified. By subsuming the fact to the norm, the consequence is asserted, that is, a type of sanction for that typified fact is entailed.

The LKIF-Core is a legal core ontology that is part of the Legal Knowledge Interchange Format (LKIF) initiative. LKIF is meant to: enable the translation between legal knowledge bases written in different representations; be a formalism to automate legal reasoning; represent a fragment of LKIF-Core.

Being a typical example of Kelsenian ontology, LKIF-Core [3] has as central notions legal norms and legal facts (represented by the notion of *qualified situations* or *cases*,

[2] In this paper, "legal ontology" is the kind of which "legal core ontology" is a subkind. However, in literature, the term "legal ontologies" is applied to any legal ontology class. In fact, several studies proposing domain ontologies, application ontologies and core ontologies use the generic term "legal ontology" rather than a more specific term. For example, both the Legal Taxonomy Syllabus tool [5], proposed for the building of legal ontologies, and the OPJK ontology proposed by [6] for legal knowledge representation are presented in the literature as "legal ontologies".

[3] as is the case for Article 155 of the Brazilian Penal Code: "To subtract, for their own benefit or of other's, someone's movable assets. Sanction (...)" [16]

see Figure 1). These are related by the aforementioned subsumption mechanism depicted schematically in Figure 2). As explained in [13]: a norm applies to (or *qualifies*) a certain generic situation (the *qualified situation*). It *allows* certain cases and *disallows* others. The obliged and prohibited cases are both subsumed by the situation to which the norm applies. These, by definition, form a complete partition of the case to which the norm applies. In LKIF-core, legal norms are classified as: Permission, Prohibition, Obligation and Rights and the legal facts are subsumed to them.

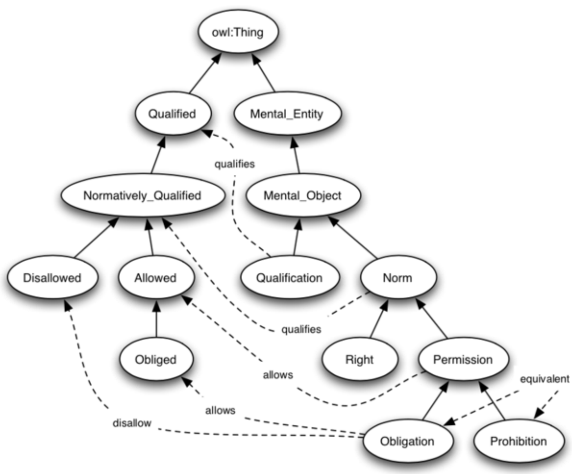


Figure 1. A fragment of LKIF-core representing Qualifications and Norms (from [13])

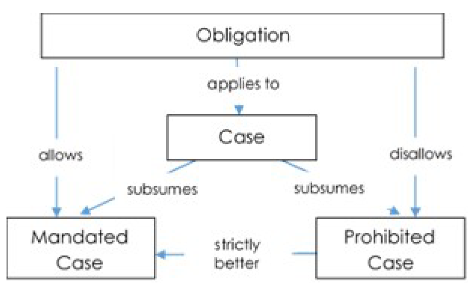


Figure 2. Subsumption structure in LKIF-core ontology (from [17])

1.2. Alexy's Theory and UFO-L

The Theory of Constitutional Rights (henceforth TCR) proposed by Robert Alexy [18] is a subjective theory of the law, in the sense that instead of focus on general relations between norms as *universals*, it contemplates legal relations that are manifested as relationships among *individuals* (subjects) in concrete specific situations. The context of TCR is the so-called Substantive Law, which is a branch of the Law that creates and regulates existing legal dispositions (e.g. rights, duties, liberties, permissions, powers) between individuals.

In a series of papers [4, 19, 20], we have proposed UFO-L, a core ontology of legal relations grounded in the Unified Foundational Ontology (UFO) [21]. UFO is a formal ontology composed of three parts: UFO-A, which is an ontology of endurants [22]; UFO-B, which is an ontology of perdurants [23]; and UFO-C, which is an ontology of social and intentional entities. UFO-L explicitly represents and articulates TCR in terms of the ontological theory of relations proposed by Guarino and Guizzardi [24] (which is a constituent micro-theory of UFO-A), and by reusing from UFO-B and UFO-C.

UFO is a Four-Category ontology that is, thus, organized around the so-called *Aristotelian Square*. In other words, it is an ontology that contemplates individuals and universals, having both (independent) substantial individuals (and substantial universals), as well as (existentially dependent) moment individuals (and moment universals). Moments can be intrinsically dependent (i.e., qualities, modes) or dependent on multiple individuals (i.e. relators) thus, binding them. Universals are further specialized into *kinds* of things, *roles* played by things of a given kind, *role mixins* representing role-like dispersive properties played by things of multiple kinds, among others.

UFO-L (whose fragment is depicted in Figure 3) extends the basic categories of UFO, prescribing the general notion of *Legal Thing*, i.e., an individual or universal that is defined in a *Legal Normative Description* (itself a type of legal individual). Legal individuals are specialized in *Legal Events*, *Legal Substantials* (*Legal Objects*, including *Legal Norms* and *Legal Normative Descriptions*, and *Legal Agents*), and *Legal Moments*. The latter is further specialized into *Legal Relators*, which are constituted by *Legal Positions* such as individualized rights, powers, liberties, subjections, etc [20]. Legal positions are *modes* and, hence, are existentially dependent on specific individuals. Based on Alexy's theory, *Right* is a legal position type having as subtypes: *Right to an Action*; *Right to an Omission*. In turn, *Liberty* (another legal position type) has as subtypes: *Unprotected Liberty*; and *Protected Liberty*. In UFO-L, a legal position has a relational nature (i.e., it is an externally dependent mode) and, thus, it exists in the scope of a reified legal relation (*Legal Relator*). Legal relations are bonds between *Legal Agents*, who then play *Legal Roles* in their scope. Legal Roles (which specialize the general notion of anti-rigid relational sortals in UFO) are prescribed by *Legal Norms*. *Legal Norm* is further specialized in *Rule* and *Principle*.

As discussed in depth in [25], UFO-L is organized in terms of a number of *Ontology Design Patterns* that extend the basic relator pattern proposed in [21]. In the sequel, we present one of these patterns, which will be instrumental to our analysis in Section 4. This pattern, termed the *Right-Duty to an Action Legal Relator (P1-RDA-LR)* [26] is depicted in Figure 4. As shown in this model, a Right-Duty to an Action is established between a *Right Holder* and a *Duty Holder*. The Legal Relator is composed of a pair of counterpart legal positions: a *Right to an Action* inhering in a *Right Holder* and externally dependent of a *Duty Holder*; and a *Duty to Act*, which inheres in a *Duty Holder* and is externally dependent on a *Right Holder*. As any legal relation, a Right-Duty to an Action is created, modified or extinguished by an event (natural or social) relevant in the scope of that normative system.

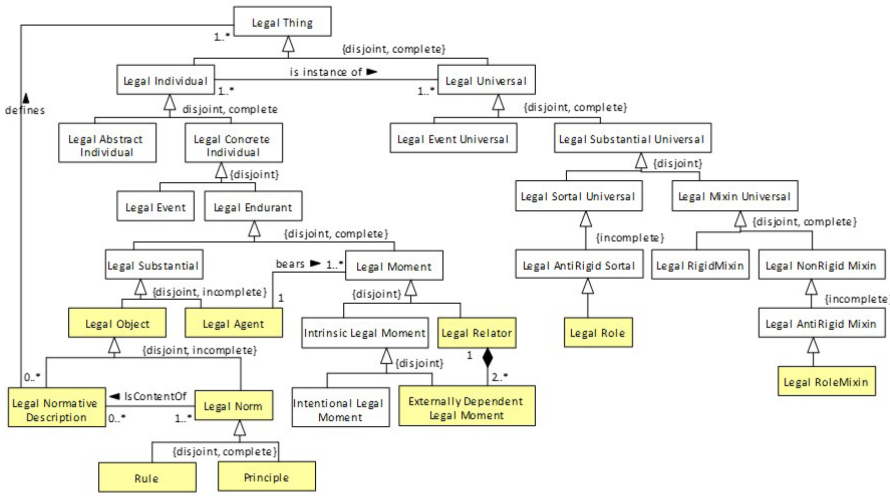


Figure 3. A Fragment of UFO-L

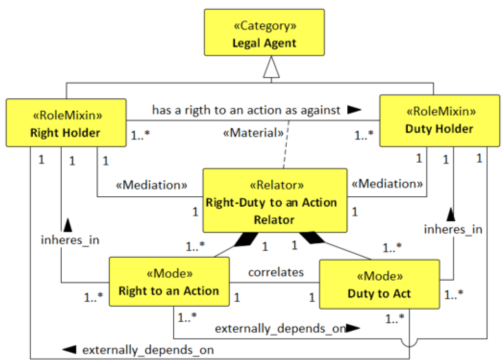


Figure 4. UFO-L: Right-Duty to an Action Legal Relator Pattern

2. Case Study: The Dedier Case

2.1. Case Description

This case (hereinafter called Dedier case) was selected from the database of the Court of Appeals of the State of Espírito Santo, in Brazil. It consists of an appeal against a decision given by the court in the first instance to a *writ of mandamus*[4]. In the sequel, we present a summary of this case:

Dedier, a civil police officer, public servant in probationary period (PE), required a leave of absence for dealing with private affairs (LDPA), more specifically, a leave from

[4]Writ of mandamus, "writ of security" or mandamus in Brazilian judiciary system is a type of action used to protect either individual or collective rights against abuse of power or illegality of a public authority or the representative of a legal entity in charge of public attributions when there is a threat to a clear legal right. It is a very similar instrument to the "writ of mandamus" in the United States of America's legal system.

his work so that he could attend a clerk training course at the National Academy of the Federal Police. This position as a trainee at the National Academy is considered a Public Position in itself. However, the Civil Police Chief of the State of Espírito Santo (PC-ES) denied his leave request based on the Complementary Law LC n.46/94, article 41 (hereafter LC 46/94) [27] that does not allow the granting of LDPA for public servants in probationary period. Dissatisfied with this decision, Dedier filed a writ of mandamus with a summary judgement injunction invoking the Brazilian constitutional principle of access to public positions prescribed in Article 5º of Brazilian Constitution (hereafter CRFB/1988) [28] and the right to LDPA. The judge of the first instance denied summary judgement because he understood that, prima facie, the right to leave would not apply for servants on probationary period. Once more, discontented with the judge's decision, Dedier filed an appeal before the Court of Appeals of the State of Espírito Santo (TJES) (process number 24079009809). Justice 'ad quem' partially overturned the first instance judge's decision, in view of the fact that, in applying Alexy's Proportionality Postulate, he found that the most appropriate rule-principle was that which least violated the principles involved (principle of probationary period versus principle of access to a public positions and principle of due process of law).

2.2. Brief Remarks

As social relations become more complex, we find more and more cases that fall outside what is prescribed by our normative system, i.e., cases for which there are no legal rules to which the facts can be subsumed. In analysing these cases of omissions, we often observe that in their core, they also exhibit situations of *collisions of constitutional rights* (or *collisions of principles* in Alexy's theory). For instance, the principle of information can collide with the principle of privacy; the principle of public health with the principle of freedom of come and go. The analysis of merit in these cases is complex and encourages the use of legal theories that propose a model of weighting and balancing solutions as opposed to the model of subsumption of facts to general and abstract norms in a closed normative system. To support this analysis, we need a model that allows for reasoning about the law in case-by-case basis, with a different analysis of the effects of norms on positions and legal relations.

3. An Ontological Analysis in LKIF-Core following Kelsen's Approach

This section presents the decision in the Dedier case in the perspective of a decision-maker who decides only based on the norms and rules at hand, as the first judge of this case. In the case under analysis, consider the fact F_D*: Dedier, a public official on probationary period at the Civil Police of the State of Espírito Santo, required a leave of absence for attendance of private interests*. Thus, if the fact F exists and a rule R exists in the legal system, applicable to fact F, it is said that fact F subsumes under the rule R. In the case of Dedier, fact F_D subsumes under the rule LC46/94 as there is no exception in the legal rule prescribed. Therefore, Dedier does not have right to the required leave.

By modeling Dedier case in LKIF-core, the module NORM proposed in [13], allows (ALLOWS) situations that match the following description:

$$\text{PUBLIC\_SERVANT\_ON\_LEAVE} \sqcap \text{TENURED\_PUBLIC\_SERVANT}$$

thus, disallowing (DISALLOWS) descriptions that match the following situation (where being a tenured public servant is the complement—logical negation of—being a public servant on probationary period):

$$\text{PUBLIC\_SERVANT\_ON\_LEAVE} \sqcap \neg\text{TENURED\_PUBLIC\_SERVANT}$$

In other words, it is necessary that every PUBLIC_SERVANT_ON_LEAVE is a TENURED_PUBLIC_SERVANT:

$$LDPA \sqsubseteq allows\ \textbf{only}\ (\text{PUBLIC\_SERVANT\_ON\_LEAVE} \sqcap \text{TENURED\_PUBLIC\_SERVANT})$$

The specific aspects of the the Dedier case are not foreseen by the Brazilian normative system. That is, from the perspective of a closed normative system, there is no norm type in this system under which this case would naturally be subsumed with all its conflicting aspects. Thus, a possibility to a favorable decision to Dedier using this perspective would be the alter of LC 46/94 by the Legislative process (i.e., the approval of a rule exception by the Parliament).

Similar to LKIF-core in terms of deontic operators, the CLO ontology implements the subsumption operation as a task of conformity checking, which, when applied to a case, it classifies the case as *conforming to* or *not conforming to* the norm. In these ontologies, the concept of *conformity* or *nonconformity* of the case to the norm is used together with the deontic operators (prohibition, permission and obligation) in monadic formulae. Conformity analysis is performed only in relation to the rule-principle to which the fact is subsumed and not to the whole set of rules-principles and principles that exist in the normative system.

In summary, in these ontologies and under the Kelsenian view, it is not possible to properly model the decision pronounced by the second judge of the case (the judge of the Appeal Court), since this view is only based on 1) legal rules of a closed normative system; and 2) a subsumption operation of fact to a given legal type.

4. An Ontological Analysis in UFO-L following Alexy's Approach

As previously discussed, following an Alexyan approach allows us to reason about the specificity of particular cases. In this particular case, we have a case of collision of principles. The collision is not observed as a collision of the rule-norm in their abstract form, but for the case of a particular individual, who playing different roles is entitled to different rights. To put it in a different way, the situation described by the combination of roles instantiated by the individual is not prescribed by that normative system (as it is often the case of complex systems in which individuals can play multiple independent roles). That is why this can be considered also as a case of normative omission.

For the Dedier case, an ontological analysis based in UFO-L allows us to identify a number of elements that are used in the decision-making process by the different legal agents involved in this case. These elements are identified in Tables 1, 2 and 3, and are explicitly represented and articulated in the remainder of this section. In particular, we model the several perspectives of the case by instantiating the Ontology Design Pattern of UFO-L introduced in Section 1 for the case of each legal relation and legal role identified.

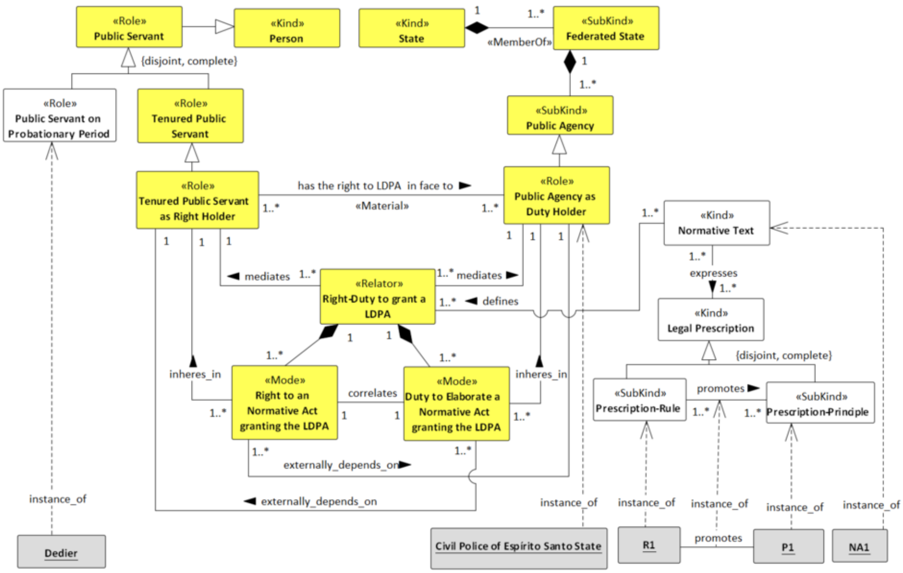


Figure 5. Perspective$_1$ modeled in UFO-L

Perspective$_1$ (Dedier does not not have the right to a LDPA): This perspective represents thesis$_1$ (Table 1), i.e., the positions of the Chief of the Civil Police and of the first judge (judge$_1$) of the case. As shown in Figure 5, the public servant Dedier is not an instance of *Tenured Public Servant Right Holder* and, as consequence, he is not entitled to the right to a leave of absence for dealing with private affairs (LDPA).

Table 1. Elements of the Decision-Making Process - Perspective$_1$

Fact *F*	Dedier, a public servant on probationary period at the Civil Police of the State of Espírito Santo, applied for a leave of absence for dealing with private affairs (LDPA).
Thesis$_1$	Dedier is a public servant on probationary period and, thus, he does not have the right to a LDPA.
Principle P$_1$	Constitutional Principle of probationary period seeks to allow for the evaluation of the aptitude of a public servant to occupy a given public position.
Normative Act NA$_1$	*LC 46/94, art. 41: The types of leaves prescribed in art. 122, V and VIII will not be granted to public servants on probationary period.*
Rule R$_1$	*Ought not* (grant the types of leaves prescribed in art. 122, V and VIII to public servants on probationary stage). LC 46/94, art. 122, VIII and LC 46/94, art. 41 are adequate to promote the Constitutional Principle P$_1$.
Legal Relation in P$_1$	There is no legal relationship between Dedier and the Civil Police Department of Espírito Santo State in terms of granting a LDPA because Dedier is in a probationary stage.

Perspective$_2$ (Dedier has the right to a LDPA): This perspective represents Dedier's thesis (Table 2) and it is the basis of his appeal petition before the Appeal Court. In this scenario (Figure 6), we have the representation of other rights to which Dedier is entitled, namely, the *Right to Non-Hindrance in Access to Public Positions* as well as the *Right to a Normative Action to have Access to Public Positions*. The articulation of legal

Table 2. Elements of the Decision-Making Process - Dedier's thesis

Thesis$_2$	As a Brazilian citizen and having fulfilled the requirements established by law, the norm that ensures access to public employment must stand out against an infra constitutional norm colliding with this principle.
Principle P$_2$	Constitutional Principle of accessibility to public employment. Access to public employment is guaranteed to all Brazilians who meet the requirements established by law and to foreigners, in accordance with the law.
Normative Act NA$_2$	Brazilian Constitution, art. 37, I.
Rule R$_2$	*Ought* to guarantee the access to public position and *Ought not* hinder access to the public position. Brazilian Constitution, art. 37, I is adequate to promote Principle P$_2$.
Legal Relation in P$_2$	Every Brazilian has the right that the State guarantees access to public employment, once the candidates fulfill the requirements established by law. Thus, there is a relationship between Dedier and PC-ES Dept. in terms of rights to a LDPA.

scenarios reflects the judge's consideration of a possible "tension" between the principles involved. In this case, we have: principle P$_1$ that articulates that a public servant on probationary period should not be entitled to leaves of absence (5); but we also have principle P$_2$ (defined by constitutional rule R$_2$) guaranteeing access to public positions. It is important to highlight that there is no conflict in the norm-rules themselves but in norm-principles in this case. The collision exists because, in this concrete case, we have the same individual playing different roles and, hence, being entitled to different rights based on different principles. In other words, the conflict arises because a decision that conforms to one of the roles might hurt rights entailed by other roles (in other relations).

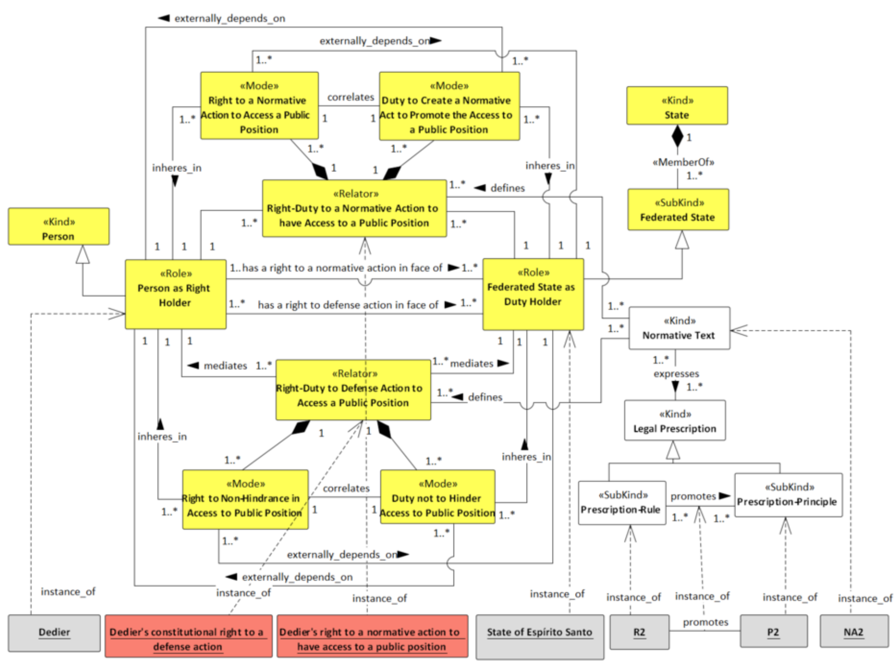


Figure 6. Perspective$_2$ modeled in UFO-L

Perspective*$_3$ *(Reasoning of Appeal Court's Justice): Figure 7 depicts the representation of a reasoning fragment of the Appeal Court's justice (judge$_2$) described in Table 3. It emerges from the existence of the *principle of due process of law* prescribed in Brazilian Constitution, art. 5, LIV [28] inherent to all procedural legal relations, as well as from the *principle of substantive due process of law*, stemming from the former. By the principle of formal due process of law, no one shall be deprived of liberty or property without a due process, and also, every act emanating from a judging authority shall observe the legal procedure to be valid and effective. The consequence of this principle is the principle of the substantive due process, that guarantees not only the formality of the process (external character), but also requires proportionality and the reasonableness of the decisions. Therefore, the principle P_3 promoted by rule R_3 is a principle that regulates every legal (and extra-judicial) procedural relation and can be raised both by the parties and judges, as occurred in the Dedier Case.

Table 3. Elements of the Decision-Making Process - Judge's reasoning

Situation$_1$	Collision of Constitutional Principles (P_1 versus P_2)
Principle P_3	Constitutional principle of substantive due process of law, express in Brazilian Constitution, art. 5º, LIV (Rule R_3).
Legal Relation in P_3	Every person has the right to the guarantee to substantive due process of law before Brazilian State.
Rule RC	Concrete legal norm introduced by the second judge of the case: the probationary stage is interrupted during the intended leave, but the full realization of the principle P_1 is possible after the end of the leave without violation of the principles P_2 and P_3.
Reasoning of Judge$_2$	Rule R_1 blocks the fulfillment of principle P_2. Rule R_1 is unconstitutional in this case, because it violates the constitutional *principle of substantive due process of law* (P_3).
Judge$_2$'s Ruling	Once rule RC is prescribed, fact F must subsumed under rule RC and not rule R_1.
Legal Relation in RC	There is a new legal relationship between Dedier and the PC-ES Dept. once the public servant has the right to LDPA before the PC-ES Dept. according to rule RC.

Judge*$_2$*'s Ruling - the analysis of perspectives 1, 2 and 3 reveals that the rule in LC 46/94 ensures the constitutional rule that promotes the principle of probationary stage (P_1), but conflicts with the constitutional rules that promote the principles of access to public positions (P_2) and substantive due process of law (P_3). Therefore, judge$_2$ deciding on the appeal of this case understood that it is possible to consider a new ruling RC (Rule RC) that ensures a minor violation of the principles involved (Figure 8).

What we have modeled in Figure 5 can be seen as a legal domain ontology module focused on the rights involved in a LDPA. What we have in Figure 8, in contrast, can be seen as an extension to this module with a specialization of the role *Public servant on Probationary Stage*. In this manner, UFO-L allows the representation of several existing perspectives in a case, according to the existence of legal relations and their relation to rules and principles. Furthermore, as we have seen in this case, it is possible to model in UFO-L both conflict of rules and collision of principles since it permits the judge to insert prescriptions in the normative system by means of judicial decisions in concrete cases.

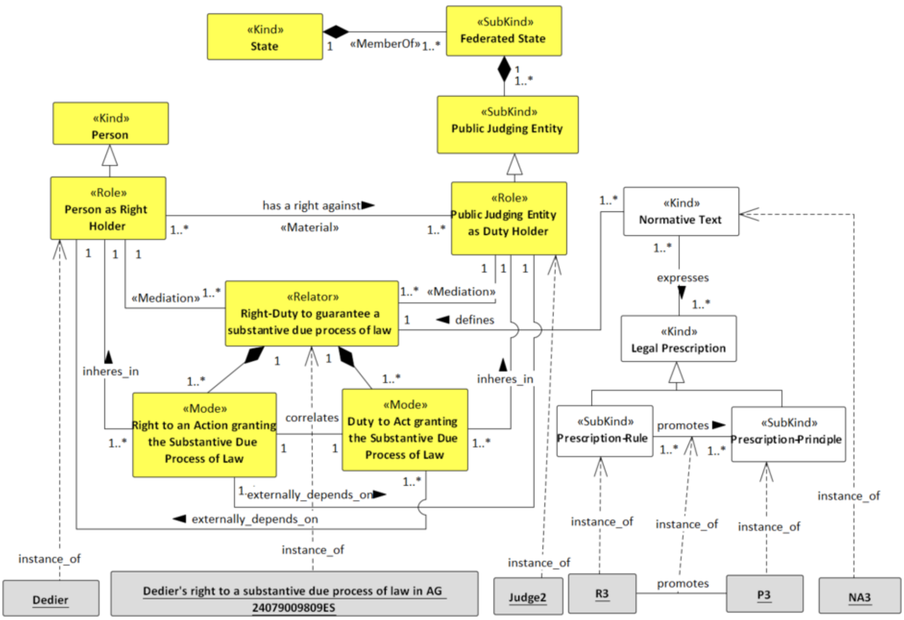


Figure 7. Perspective$_3$ modeled in UFO-L: judge$_2$'s reasoning fragment

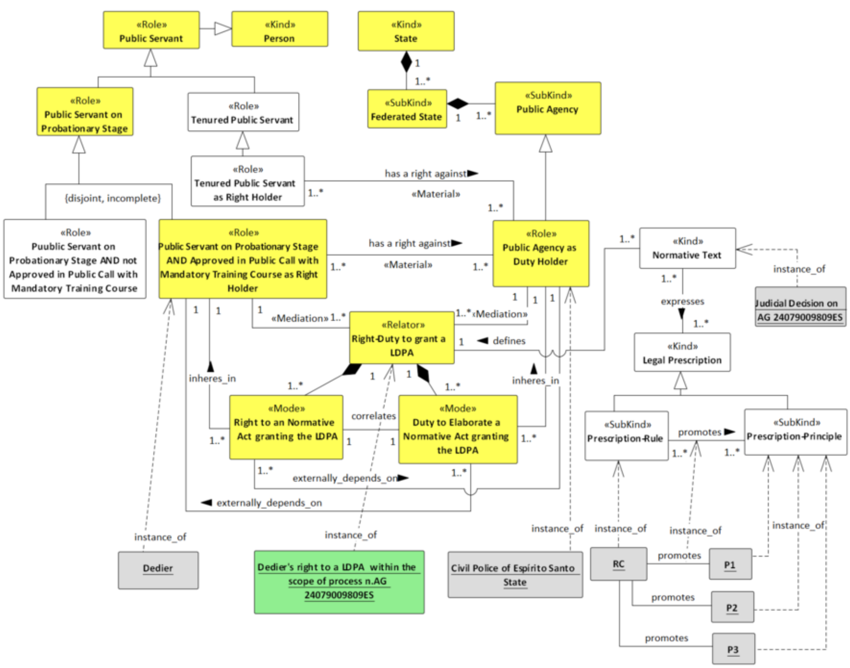


Figure 8. Perspective$_3$ modeled in UFO-L: judge$_2$'s ruling

5. Final Considerations

In this paper, we conducted an ontological analyses with the support offered by two Legal Core Ontologies (LKIF-Core and UFO-L) for Judicial Decision-Making. We chose these ontologies because they represent two classes of Legal Theories, namely, Kelsen's Pure Theory of the Law (LKIF-Core) and Alexy's Theory of Constitutional Rights (UFO-L). Our goal is to demonstrate how these theories embedded in the design of these ontologies influence their ability to offer informative insights in complex cases of decision-making in which we have normative omissions and collisions of principles. We do this by employing a selected real case.

Normative omissions and collision of principles are not logically connected but frequently appear together in real cases. The underlying phenomenon here is the following. In legal positivism (Kelsen), the law is described as a system of relations between *general* (universal) norms. However, in complex legal settings, we often have cases in which, despite the absence of conflict between general norms, we have conflicts of legal positions (e.g., rights) inhering in specific *individuals*. In other words, the collision emerges because the same individual can play different roles (and each of which can entail conflicting positions). The collision of principles in these cases can be considered a case of omission because the rules that would otherwise govern that particular individual case (i.e., under which that case could be subsumed) will always be missing in practice in these cases. Since no sufficiently complex normative system (defined at the level of general norms) can be guaranteed to be complete (i.e., accounting for all particular situations), judicial decision-making should be supported by a framework that allows for reasoning in terms of individuals, the roles they play, the legal positions they bear and the relations they establish with other legal agents. As our analysis shows, Alexy's theory and the UFO-L ontology allows for a much more suitable, detailed and informative analysis of such cases, which is required to capture the semantics of legal decisions such as those discussed in the Dedier case.

A future work will be to perform an additional empirical study with a relevant sample of cases with collisions of principles to verify the degree to which UFO-L helps in bringing more clarity and understanding of the theses in a judicial case.

Acknowledgement. João Paulo A. Almeida is funded by the Brazilian Research Funding Agencies CNPq (grant number 312123/2017-5).

References

[1] Capurso TJ. How judges judge: Theories on judicial decision making. University of Baltimore Law Forum. 1999;29:5.

[2] Griffo C, Almeida JPA, Guizzardi G. A Systematic Mapping of the Literature on Legal Core Ontologies. In: Proc. Brazilian Seminar on Ontologies (ONTOBRAS 2015). vol. 1442 of CEUR Workshop Proceedings; 2015. p. 1–12.

[3] Rinke H, Joost B, Marcello DB, Alexander B. LKIF Core: Principled Ontology Development for the Legal Domain. Frontiers in Artificial Intelligence and Applications. 2009;188: Law, Ontologies and the Semantic Web:21–52.

[4] Griffo C, Almeida JPA, Guizzardi G. Towards a Legal Core Ontology based on Alexy's Theory of Fundamental Rights. MWAIL, ICAIL. 2015;.

[5] Ajani G, Boella G, Lesmo L, et al. Multilevel Legal Ontologies. In: LNCS. vol. 6036. Springer Berlin Heidelberg; 2010. p. 136–154.
[6] Caralt NC. Modelling Legal Knowledge through Ontologies. OPJK: the Ontology of Professional Judicial Knowledge [PhD thesis]. UAB, Spain. Barcelona; 2008.
[7] Valente A, Breuker J. Towards Principled Core Ontologies. Proc Tenth Work Knowl Acquis Knowledge-Based Syst. 1996;.
[8] Kralingen RV, Oskamp E, Reurings E. Norm Frames in the Representation of Laws. In: Leg. Knowl. Based Syst. JURIX 93.; 1993. p. 11–22.
[9] Valente A. Legal Knowledge Engineering; A Modelling Approach. Amsterdam: IOS Press; 1995.
[10] Shaheed J, Yip A, Cunningham J. A Top-Level Language-Biased Legal Ontology. In: ICAIL Work. Leg. Ontol. Artif. Intell. Tech.; 2005. .
[11] Gangemi A, Sagri MT, Tiscornia D. A Constructive Framework for Legal Ontologies. In: Law Semant. Web. Berlin: Springer Berlin Heidelberg; 2004. p. 97–124.
[12] Breuker J, Hoekstra R, Boer A, van den Berg K, Sartor G, Rubino R, et al. OWL Ontology of Basic Legal Concepts (LKIF-Core). University of Amsterdam; 2007.
[13] Hoekstra R, Breuker J, Di Bello M, Boer A. The LKIF Core Ontology of Basic Legal Concepts. In: CEUR Work. Proc.. vol. 321; 2007. p. 43–63.
[14] Kelsen H. Pure Theory of Law. Berkeley, CA: University of California Press; 1934.
[15] Green L, Adams T. Legal Positivism. In: The Stanford Encyclopedia of Philosophy. Winter 2019 ed. Metaphysics Research Lab, Stanford University; 2019. .
[16] Brasil. Código Penal. 45th ed. São Paulo: Saraiva; 2007.
[17] Boer A, Winkels R, Vitali F. Metalex XML and the legal knowledge interchange format. In: Computable models of the law. Springer; 2008. p. 21–41.
[18] Alexy R. A Theory of Constitutional Rights. Oxford: Oxford Univ. Press; 2009.
[19] Griffo C, Almeida JPA, Guizzardi G, Nardi JC. From an Ontology of Service Contracts to Contract Modeling in Enterprise Architecture. In: 2017 IEEE 21st International EDOC Conference; 2017. p. 40–49.
[20] Griffo C, Almeida JPA, Guizzardi G. Conceptual Modeling of Legal Relations. In: Int. Conf. Concept. Model. Xi'an, China: Springer; 2018. p. 169–183.
[21] Guizzardi G. Agent Roles, Qua Individuals and The Counting Problem. In: Int. Work. Softw. Eng. Large-Scale Multi-agent Syst.; 2005. p. 143–160.
[22] Guizzardi R, Perini A, Susi A. Aligning goal and decision modeling. In: CAiSE Forum 2018, LNBIP. vol. 317. Springer International Publishing; 2018. p. 105–113.
[23] Benevides AB, Bourguet JR, Guizzardi G, Penãloza R, Almeida JPA. Representing a reference foundational ontology of events in SROIQ. Applied Ontology. 2019 Aug;14(3):293–334. Available from: http://doi.org/10.3233/AO-190214.
[24] Guarino N, Guizzardi G. "We need to discuss the Relationship": Revisiting Relationships as Modeling Constructs. In: 27th Int. Conf. CAiSE 2015 Proc. Sweden: Springer International Publishing; 2015. p. 279–294.
[25] Griffo C, Almeida JPA, Guizzardi G, Nardi JC. Service contract modeling in enterprise architecture: An ontology-based approach. Inf Syst. 2019;p. 101454.
[26] Griffo C. UFO-L: Uma Ontologia Núcleo de Aspectos Jurídicos Construída sob a Perspectiva das Relações Jurídicas [PhD Thesis]. Ufes; 2018.
[27] Espírito Santo (Estado). Lei Complementar n. 46/94. Regime Jurídico dos Servidores Públicos do Estado do Espírito Santo. Vitória, ES: DOES; 1994.
[28] Brasil. Constituição da República Federativa do Brasil. Brasília: Senado Federal; 1988.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200662

A Commonsense Theory of Secrets

Haythem O. ISMAIL [a] and Merna SHAFIE [b]
[a] *Cairo University and the German University in Cairo, Egypt, haythem.ismail@guc.edu.eg*
[b] *The German University in Cairo, Egypt, merna.shafiee@guc.edu.eg*

Abstract. With the advent of social robots, precise accounts of an increasing number of social phenomena are called for. Although the phenomenon of secrets is an important part of everyday social situations, logical accounts of it can only be found, in a rather strict sense, within logical investigations of systems security. This paper is an attempt to formalize the logic of a commonsense notion of secrets as a contribution to ontologies of social and epistemological phenomena. We take a secret to be a five-way relation between a proposition, a group of secret-keepers, a group of nescients, a condition of secrecy, and a time point. A bare-bones notion of secrets is defined by providing necessary and sufficient conditions for said relation to hold. Special classes of secrets are then identified by considering an assortment of extra conditions. The logical language employed formalizes a classical account of belief and intention, a theory of groups, and a novel notion of revealing. In such a rich theory, interesting properties of secrets are proved.

Keywords. Secrets, Commonsense reasoning, Belief, Intention, Revelation

1. Introduction

Secrets come in all shapes and sizes: They can be classified military maps, family-devastating incidences of spouse infidelity, critical credit card pin numbers, questions in an exam, names of academy award winners, locations of treasures, sorcerous procedures for invisibility, or embarrassing childhood mischiefs. Secrets are often hard to keep, yet they are sometimes gratifying to be part of. They are catalysts for suspicion, but they are also gauges of trust. Some secrets are a social necessity, and most are psychological burdens [1,2,3].

The list of intuitions about and curiosities of secrets can go on and on. But we are not concerned here with enumerating them, nor are we willing to analyze most of them. Our objective is more modest and more fundamental. For, while secrets are social/psychological phenomena par excellence, they have an obscure ontological/epistemological flip side. Guided by some foundational intuitions about secrets, we seek to arrive at a commonsense theory of secrets which is precise enough to be amenable to logical analysis. Such *logic of secrets* should be a necessary component of a logic-based artificial intelligence system which is expected to competently engage in social interaction with people. In the near future, social robots may be everywhere around us, assisting us at work, at hospitals, with house chores, and granted the status of trusted life partners [4]. These robotic companions should be capable of understanding what secrets are and of keeping our secrets.

As far as we know, studies of secrecy, within the logical tradition, are confined to issues related to system security [5,6,7,8,9,10,11,12,13,14]. In such studies, a secret is presented as a *true* piece of information about an agent/a system which is not known by a certain adversary group; the focus is mostly on identifying effective, and often subtle, methods for keeping the secret. Several aspects of secrecy are not considered by these studies. For example, there is no account of a secret keeper's intention to keep the secret (which is *the* defining characteristic of secrecy [3]), no investigation of what it means for a secret to be revealed to someone (which does not, in general, effect knowledge or belief), and no discussion of the possible relations among keepers of the same secret.

The paper is structured as follows. Section 2 discusses ten commonsense intuitions about secrets and Section 3 motivates the novel notion of revelation. Section 4 presents a logical language for reasoning about secrets. Section 5 includes a number of theorems proved using the introduced logic. A typology of secrets is presented in Section 6. Finally, Section 7 reviews related work.

2. Ten Intuitions About Secrets

We would like to start our investigation by drawing a distinction between secrets on one hand and the *objects of* secrets on the other. That a particular military map is classified is not, in general, an intrinsic property of the map itself [15]; rather, it is an extrinsic property that the map acquires (possibly temporarily) by virtue of standing in a complex relation to other entities, notably a secret keeper or a group thereof. Further, it is possible that the same map is kept secret by General *A* from their spouse and, simultaneously and independently, by General *B* from their own spouse. We would like to say that there are *two* secrets here, both having the same map as their object, and that, for example, one secret may be divulged and the other not. While English locutions such as "This map is a secret" are common, a careful analysis of secrecy should not conflate secrets and their objects. In particular, the existence of objects of secrecy is a necessary but an insufficient condition for the existence of secrets. Henceforth, we shall refer to objects of secrecy as "secreta" (plural of "secretum") and shall take secrets to be relations between secreta and other entities—this is our first intuition.[1]

I1. Secrets are distinct from secreta; they are relations between secreta and other entities.

This, then, leads to the following question: What kind of entity can a secretum be? It would seem that all sorts of beasts can be secreta: pin numbers, names of academy award winners, military maps, recipes for invisibility, etc. We contend, however, that this display of diversity is an artifact of the elliptical language we use to talk about secrets. It is not, for example, the credit card pin number itself that is the secretum; the pin number may happen to be the date of birth of the card owner who can write it down on a sheet of paper, show it to everybody, and say that it is their birth date without thus revealing their secret pin number. Rather, the secretum is *the proposition that this number is the credit card pin number*. Similarly, prior to announcing the names of the academy award winners in 2019, a name such as "Rami Malek" was by no means a secretum, the secretum was *the proposition that "Rami Malek" is the name of the winner of the Best Actor award*. Henceforth, we uphold the following intuition.[2]

[1] This is similar to distinguishing the object of an intention–an action, for instance–from the intention itself which is a complex attitude an agent holds towards that object; likewise for belief and the object of belief.

[2] Some readers may be suspicious about **I2** due to examples such as the classified military map, where it makes sense for the responsible General *A* to physically hide the map itself from their spouse. While we agree that hiding the map is indeed the right thing to do, we do not agree that this makes the map itself a secretum.

I2. Secreta are propositions.

We now turn to the "entities", referred to in **I1**, whose standing in some relation to a secretum constitutes a secret. First, for a secret to exist, there must be some *secret keepers*. It makes little sense to claim that there is a secret which no one is keeping. Mere unawareness of a proposition or concealment thereof does not make it a secretum. For example, before discovering that the earth is spheroid, no one was aware of this fact. Nevertheless, it is hardly acceptable to claim that the earth's roundness was a secretum at that time, primarily because no one could *keep* it a secret since no one was aware of it in the first place. Similarly, that raw gold is *hidden* within some mountain is no secret until somebody discovers it and decides to keep it to themselves.

I3. For every secret, there is a group of secret keepers.

Not only do secrets require someone to keep them but they also require someone to be kept *from*. A person who is cast away on a deserted island with no hope of getting rescued cannot be said to be keeping any secrets simply because there is no one to hide it from. This is so even though whatever happens to them on the island is completely concealed from everyone. Hence, whenever there is a secret, there is a group from whom the secret is kept; we refer to its members as *nescients*.

I4. For every secret, there is a group of nescients.

Now, let us clarify what we mean by "group" in **I3** and **I4**. Each group is identified by a *group condition*. At any point in time, the set of group members is the set of agents satisfying the group condition at this time. An *extensional* group is one for which the group condition is membership in a certain *set*. Since members of a set are fixed over time, the members of an extensional group never change. An *intensional* group is a group which is not extensional. (Extensional groups are similar to the "plural individuals" of [16,17] or the "E-collectives" of [18]; intensional groups are the "groups" of [16,17] or the "I-collectives" of [18]). Secret keepers and nescients could be of either group type. For example, a crush on a high-school colleague is possibly a personal secret with an extensional (singleton) group of secret keepers. On the other hand, an esoteric sorcerous procedure for invisibility may be kept secret by an intensional group of sorcerers who keep on handing it down for centuries across generations. In both examples, nescients form the (extensional and, respectively, intensional) group which includes all those who are not secret keepers. Extensional and intensional groups may be empty. An extensional group is empty only if the corresponding set is empty. Such a (unique) group is *necessarily forever empty* (NFE). An intensional group may become temporarily empty for a certain period during which no one satisfies the group condition. If the group condition is such that, starting at some time, it is necessarily the case that no one would ever satisfy its condition, then this intensional group is also NFE. It is necessary for the existence of secrets that the group of keepers is not NFE, but it may (if intensional) be temporarily empty, and that the group of nescients is not believed to be NFE by any of the keepers.

Rather, the secretum is the proposition that the map is a map of some critical military site. The spouse's finding the map causes the revelation of *the proposition that there is this strange map* which, given the nature of the Genaral's work, may lead to the conclusion that it is a map of some military site.

I5. Secret keepers and nescients form extensional or intensional groups. The group of keepers is not necessarily forever empty and none of its members believe that the group of nescients is necessarily forever empty.

Secrets are, in general, not eternal [19]. Most secrets are only kept so long as some condition of secrecy holds. For example, as per the "Automatic Declassification Program" in the United States of America,

> Information appraised as having permanent historical value is automatically declassified once it reaches 25 years of age unless an agency head has determined that it falls within a narrow exemption that permits continued classification and it has been appropriately approved. [20]

I6. A proposition is only a secretum as long as some condition of secrecy holds.

The condition of secrecy is a condition on the persistence of the keepers' intention to keep the secret. It often happens, however, that a secret is (accidentally or ill-intentionally) exposed to a nescient when the condition of secrecy still holds. These are cases in which the secret keepers fail to keep the secret. Compare, for example, between the natural expiration of a secret exam, which happens when students sit for the exam, and its premature exposure as a result of a malicious student's gaining access to the professor's computer. Hence, secrets are temporary in the stronger sense that, regardless of the secrecy condition, they may fail to be kept as long as intended.[3]

I7. Secrets are temporary.

Hence, we take secrets to be four-way relations which temporarily hold between a secretum ϕ, a group K of secret keepers, a group N of nescients, and a condition C of secrecy. Formally, we write $Secret(\phi, K, N, C, t)$ to state that, at time t, proposition ϕ is kept a secret by the group K from the group N while proposition C holds. Exactly what conditions are necessary and sufficient for said relation to hold is what we now turn to.

First, in a genuine secrecy situation, all members of K believe ϕ [21] and C—otherwise they will have no motivation for keeping ϕ a secret.

I8. Secret keepers believe both the secretum and the secrecy condition.

A possible objection to **I8** is that people often confide their secrets to others [22]. For example, x may inform y about their secretum ϕ and ask them to never reveal it to anybody. While y may fail to believe ϕ, they, nevertheless, form the intention of never mentioning it to anyone. Can we then say that the (extensional) group formed of x and y is keeping ϕ a secret from everyone else? We do not think so. Note that y's intention to never mention ϕ to anyone may be based solely on their commitment to honesty; since y does not believe ϕ, it would be deceptive to state it [23]. Thus, the situation here is indistinguishable from one in which y is simply being honest; it would be awkward to claim that every time we decide to not tell a lie we are keeping a secret. Even if y is generally dishonest, and are more than willing to lie about ϕ, but they do not do it out of respect for x's wishes, it is

[3] We follow [19] in taking "temporary" to qualify a phenomenon as being not necessarily permanent. Hence, while secrets are, in general, temporary, some secrets may *happen* to be kept permenantly.

still not plausible to claim that they are keeping ϕ a secret. We have all sorts of reasons for not saying certain things (especially if we do not believe them); among other things, we do it to be respectful, polite, suspenseful, and even spiteful. It is hardly acceptable to claim that in all these situations we are keeping a secret.[4] Notwithstanding the above argument, we are not saying that y is not keeping any secrets here; y is indeed keeping a secret, the secretum is not ϕ, but that ϕ *is a secretum of* x, which y indeed believes.

The second ingredient of the secrecy relation is that none of the secret keepers believes that the secretum has been revealed to a nescient. If they do, they will have no reason to continue keeping the secret.

I9. No secret keeper believes that the secretum has been revealed to a nescient.

There are at least two things to say about **I9**. First, sometimes a single nescient n gets to know about ϕ. This often does not result in the secret keepers' publicly disclosing the secret; they may choose to continue keeping it from the rest of the nescients. This is, however, not a counterexample to **I9**, for the secret has undergone a major change following the revelation to n. In particular, thenceforth, the group of nescients has changed into a group which does not contain n—resulting in a new secrecy relation. Second, **I9** is necessary to rule out certain situations which would otherwise be, counter-intuitively, counted as secrets. For example, n's friends, aware of how much weight-conscious they are, may decide, out of sheer courtesy, to never point out n's recent, visible weight gain. This is not a case of secrecy exactly because everyone knows that n is aware of the gain in their weight. The final ingredient of secrecy, and the most fundamental [3], is the keepers' *intention* to indeed keep the secret.

I10. Every secret keeper intends that the secretum is not revealed to a nescient as long as the secrecy condition holds.

Independently-motivated properties of intention yield intuitive properties of secrets. For example, according to [24], one cannot intend a proposition if they believe it to be false. Thus, in normal circumstances, it would be futile to keep the name of the capital of Georgia a secret since anybody can easily gain access to this public piece of information.[5]

In Section 4, we present a logical language in which we formalize the definition of a secret with regards to the previously mentioned intuitions. First, however, we need to elucidate the central notion of revelation.

3. Revelation

What does it mean for ϕ to be revealed to n? A prototypical revelation scenario is one in which an agent A truthfully states the true proposition ϕ to n, who does not know ϕ, thereby resulting in n's coming to believe ϕ. Not all instances of revelation, however, share the features of this idealized situation. First, whereas typical uses of the English "reveal" seem to indeed presuppose the truth of the revealed proposition [25], in our analysis of secrets (particularly, **I9** and **I10**), we do not want to assume that ϕ is true. Hence, the

[4] A parent, for example, may not want their young, gullible children to get exposed to some racist doctrines, which the parent does not believe in, lest they may subconsciously adopt them. We would not say that such doctrines are secrets *of* the parent but that they rather be kept unrevealed to their children.

[5] In *abnormal* circumstances, where the secret keeper can lock up the nescient and isolate them from the rest of the world, such a secret would be possible.

notion of revelation we need here does not carry this particular presupposition of the English verb. Second, we would like to capture a notion of revelation which does not assume that the revealer believes ϕ. Someone who is keeping ϕ a secret from n would take their secret to have been divulged following A's revelation, even if A does not believe ϕ and is attempting to mislead n. Third, ϕ's being revealed to n need not necessarily imply n's believing ϕ. For example, consider a professor who is keeping the questions of an exam secret from their students, but not from their assistant. Now suppose that the assistant discloses the contents of the exam to a student. The student, however, does not believe the assistant, thinking that they must be misleading them. In this situation, the professor would still consider their secret to have been divulged and would, typically, change the exam. Finally, in many cases, there is no agent A who reveals ϕ to n; mere perception of a state of affairs by n may be sufficient for the revelation of ϕ.

We are, thus, left with a very weak notion of revelation: "ϕ is revealed to n" means that n was somehow (possibly via perception) informed about ϕ. Revelation is not vacuous, though; it is strictly stronger than mere *awareness* [26]. For example, prior to announcing names of the academy award winners in 2019, everybody (who was interested) entertained, and was thus *aware*, of the proposition that Rami Malek is the winner of the best actor award; but this proposition was only *revealed* during the ceremonies. Thus, revelation is strictly stronger than awareness but strictly weaker than belief. We propose to intuitively construe ϕ's revelation to n as n's having (positive) evidence for ϕ. This being said, revelation is, thus, a special kind of modality. In particular, one can have evidence for both ϕ and $\neg\phi$; hence, both propositions may be revealed. We take this intuition up more seriously below by modeling revelation along the lines of the logic of evidence of [27].

4. Formalizing Secrets

To formalize secrets, we use a language $\mathscr{L}_S$ based on (a fragment of) the language VEL of [28], equipped with a special sort for groups, two normal modal operators for belief and intention, and a non-normal modal operator, akin to the evidence operator of [27], for revelation. Limitations of space allow us to only provide a sketch of the syntax and semantics of $\mathscr{L}_S$.

$\mathscr{L}_S$ is a sorted, first-order language with equality. In particular, there is a sort σ_A for agent-denoting terms, a sort σ_G for group-denoting terms, and a sort σ_T for time-denoting terms. A set of $\mathscr{L}_S$-atoms is generated in the usual way from countable sets of predicate symbols, function symbols, and variables. A special function symbol $[\cdot]$ combines with a term of sort σ_A to form a term of sort σ_G. Function symbols $\sqcup$ and $\sqcap$ form terms of sort σ_G from pairs of σ_G terms. Intuitively, $[A]$ denotes the extensional group comprised of the single member denoted by A, $G_1 \sqcup G_2$ and $G_1 \sqcap G_2$ denote the groups whose sets of members at any time are, respectively, the union and intersection of the sets of members of G_1 and G_2. A special binary predicate symbol Mem forms an atom by combining with terms of sorts σ_A and σ_G, respectively; intuitively, $Mem(A,G)$ means that agent A is a member of group G. Moreover, we have atoms of the form $\alpha = \beta$ (with the obvious semantics), where α and β are of the same sort, atoms of the form $t_1 \leq t_2$, where t_1 and t_2 are of sort σ_T, which mean that time point t_1 is no later than time point t_2, and atoms of the form $AT(t)$ which mean that the time (of evaluation) is t. $\mathscr{L}_S$ is the smallest set of formulas generated by the following grammar (and respecting the signatures of the predicate and function symbols).

$$\phi := P \mid \neg\phi \mid \phi \wedge \phi \mid \forall x[\phi] \mid \circledast\phi \mid H(\phi,t) \mid B(A,\phi) \mid I(A,\phi) \mid R(A,\phi)$$

where P is an atom, A is of sort σ_A, and t is of sort σ_T. Other logical connectives and the existential quantifier are defined in the standard way.

Expressions of $\mathscr{L}_S$ are interpreted over a branching tree structure. Each node in the tree is referred to as a *state*, and every state has a unique past and several possible futures [28]. A complete branch through the tree is a *history*, which is a bijection from a linearly-ordered set of time points to the set of states. Thus, one can view a history-time pair (h,τ) as a state. All expressions of the language are interpreted at such a pair (h,τ). In particular, where $\mathscr{V}$ is a valuation of the terms and the atoms, $H(\phi,t)$ means that "ϕ holds at t" and $[\![H(\phi,t)]\!]^{\mathscr{V}}_{h,\tau}$ is true if and only if $[\![\phi]\!]^{\mathscr{V}}_{h,[\![t]\!]^{\mathscr{V}}_{h,\tau}}$ is true. The expression $[\![\circledast\phi]\!]^{\mathscr{V}}_{h,\tau}$ is true if $[\![\phi]\!]^{\mathscr{V}}_{h',\tau}$ is true at all histories h' that coincide with h up to τ.

Formulas of the form $B(A,\phi)$ and $I(A,\phi)$, respectively, mean that "agent A believes ϕ" and "agent A intends ϕ", and are interpreted in the standard way using accessibility relations, one for each agent, on the set of history-time pairs (h,τ). A formula $R(A,\phi)$ intuitively means that ϕ is revealed to A. Following [27], we interpret revelation formulas using a function $\mathscr{R}$ which maps every agent and history-time pair (h,τ) to a family of sets of history-time pairs (h,τ) (each set, intuitively, corresponding to a proposition which is revealed to the agent in the history-time pair (h,τ)). Two important constraints on these families is that none of them is empty (tautologies are all revealed) or contains the empty set (contradictions are never revealed). Crucially, the families of sets are not closed under intersection, allowing agents to have contradictory propositions revealed to them without commitment to the revelation of falsehood.Thus, R is not a normal modal operator [29]. Hence, following [27], $[\![R(A,\phi)]\!]^{\mathscr{V}}_{h,\tau}$ is true if and only if there is some $X \in \mathscr{R}([\![A]\!]^{\mathscr{V}}_{h,\tau}, h, \tau)$ such that $[\![\phi]\!]^{\mathscr{V}}_{h',\tau'}$ is true for every $(h',\tau') \in X$.

Note that the notion of revelation proposed here is a *passive* one; our modal operator R informally stands for what it means for a proposition to be revealed (in the sense of its being exposed or not covered) to an agent. We are not accounting for *acts* of revelation. As such, R is akin to B, and our account does not explain how revelation is caused just as no common account of belief investigates events that result in belief.

An axiomatic system, referred to as Σ, that captures the basic intuitions we have about the meaning of the various operators is displayed in Figure 1. (Variables are universally-quantified with widest scope unless otherwise indicated.)[6] Though not crucial for proving our theorems, we include (in the right column) axioms for the VEL [28] fragment we employ for completeness. As is common, B is a $KD45$ and I is a KD modal operator. **IB1** and **IB2** indicate that agents are never wrong about having or lacking intentions. **IB3** is motivated by [24]. It captures the intuition that intentions should be dropped once it is realized that they are impossible to achieve.

R1 indicates that tautologies are revealed and contradictions are not, **R2** requires revelation to be closed under logical implication, and **R3** states that a revelation of a revelation amounts to a revelation. (Imagine someone telling A that B was told that C's credit card pin number is C's birth date.) **BR1** and **BR2** demonstrate the intimate relation between belief and revelation. **BR1** is a weakened variant of a K axiom for R; the requirement that $\neg\phi$ is not believed is necessary to avoid cases where $\phi \to \psi$ is only

[6]Constraints on the semantic structure that ensure the validity of these axioms were identified but are not discussed here for limitations of space. Most of these are standard, however, except possibly for those pertaining to the revelation axioms.

*KD*45 axioms for *B*.
KD axioms for *I*.

***IB* bridge axioms.**
IB1. $\neg I(x,\phi) \leftrightarrow B(x,\neg I(x,\phi))$
IB2. $I(x,\phi) \leftrightarrow B(x,I(x,\phi))$
IB3. $I(x,\phi) \to \neg B(x,\neg\phi)$

***R* axioms.**
R1. $R(x,\phi) \wedge \neg R(x,\neg\phi)$, if $\vdash \phi$
R2. $R(x,\phi) \to R(x,\psi)$, if $\vdash \phi \to \psi$
R3. $R(x,R(y,\phi)) \to R(x,\phi)$

***BR* bridge axioms.**
BR1. $[B(x,\phi \to \psi) \wedge \neg B(x,\neg\phi)] \to [R(x,\phi) \to R(x,\psi)]$
BR2. $R(x,\phi) \to B(x,R(x,\phi))$

Group axioms.
G1. $Mem(x,[y]) \leftrightarrow x = y$
G2. $Mem(x,G1 \sqcup G2) \leftrightarrow Mem(x,G1) \vee Mem(x,G2)$
G3. $Mem(x,G1 \sqcap G2) \leftrightarrow Mem(x,G1) \wedge Mem(x,G2)]$

VEL Axioms [28].
TP1. $H(\phi,t)$, if $\vdash \phi$
TP2. $(t \leq t' \wedge t' \leq t'') \to t \leq t'$
TP3. $t \leq t' \vee t' \leq t$
TP4. $(t \leq t' \wedge t' \leq t) \leftrightarrow t = t'$
TP5. $(H(\phi,t) \wedge H(\phi \to \psi,t)) \to H(\psi,t)$
TP6. $\neg H(\phi \wedge \neg\phi,t)$
TP7. $H(\phi,t) \vee H(\neg\phi,t)$
TP8. $H(\phi,t) \leftrightarrow H(H(\phi,t),t')$
TP9. $t \leq t' \leftrightarrow H(t \leq t',t'')$
TP10. $\forall t[H(\phi,t')] \to H(\forall t[\phi],t')$
TP11. $AT(t) \wedge AT(t') \to t = t'$
TP12. $H(AT(t),t)$
TP13. $\phi \to \exists t[H(\phi,t)]$
BA1. $\boxast\phi$, if $\vdash \phi$
BA2. $(\boxast\phi \wedge \boxast(\phi \to \psi)) \to \boxast\psi$
BA3. $\boxast\phi \to \phi$
BA4. $AT(t) \to \boxast AT(t)$
BA5. $t \leq t' \to \boxast(t \leq t')$
BA6. $H(\boxast\phi,t) \wedge t \leq t' \to H(\boxast H(\phi,t),t')$

Figure 1. System Σ of $\mathscr{L}_S$ axioms

- **T1.** $R(x,\phi \wedge \psi) \to R(x,\phi) \wedge R(x,\psi)$
- **T2.** $B(x,\phi) \to R(x,\phi)$
- **T3.** $R(x,\phi) \leftrightarrow R(x,R(x,\phi))$
- **T4.** $B(x,\phi) \to B(x,R(x,\phi))$
- **T5.** $B(x,R(x,\phi)) \to R(x,\phi)$
- **T6.** $B(x,\neg R(x,\phi)) \to \neg R(x,\phi)$
- **T7.** $R(x,B(x,\phi)) \to B(x,R(x,\phi))$

Figure 2. Some theorems of Σ

trivially believed. **BR2** means that agents have complete beliefs about their revelations. The revelation theorems in Figure 2 can be easily proved to follow from Σ.[7]

Henceforth, we make use of the following abbreviation: If O is B, I, R, or Mem; α and β are terms of the appropriate sorts; and t is of sort σ_T then we write $O(\alpha,\beta,t)$ as a shorthand for $H(O(\alpha,\beta),t)$. The following definition is a precise characterization of the simplest, bare-bones notion of secrecy based on the intuitions presented in Section 2:

$$Secret_0(\phi,K,N,\psi,t) =_{\text{def}} \neg NFE(K,t) \wedge \forall x[Mem(x,K,t) \to B(x,\phi \wedge \psi \wedge \neg NFE(N,t),t) \wedge \neg\mathscr{B}(\phi,x,N,t) \wedge \mathscr{I}(\phi,x,N,\psi,t)]$$

where

$$NFE(G) =_{\text{def}} \boxast\neg\mathbf{F}(\exists x[Mem(x,G)])$$
$$\mathbf{F}\phi =_{\text{def}} \exists t1,t2[AT(t1) \wedge t1 \leq t2 \wedge H(\phi,t2)]$$
$$\mathscr{B}(\phi,\alpha,N,t) =_{\text{def}} B(\alpha,\ \exists y[Mem(y,N,t) \wedge R(y,\phi,t)],\ t)$$

[7] All proofs are available here.

$$\mathscr{I}\ (\phi,\alpha,N,\psi,t) =_{\text{def}} \forall y,t'[I(\alpha,\, t \leq t' \ \wedge\ Mem(y,N,t') \ \wedge \forall t''[t < t'' \leq t' \rightarrow H(\psi,t'')] \rightarrow \neg R(y,\phi,t'),\, t)]$$

Thus, at time t, group K keeps the secretum ϕ a secret from group N, under the condition ψ if, at t, the group of secret keepers is not necessarily forever empty and each secret keeper

1. believes ϕ, ψ and that the group of nescients is not necessarily forever empty (**I5**, **I8**);
2. does not believe that there is a nescient to whom ϕ is revealed at t (**I9**); and
3. has the intention that at all future times t', such that ψ persists from t through t', ϕ is not revealed to any nescient (**I10**).

5. Seven Theorems on Secrets

In this section, we prove some results about secrets. Some of these are quite intuitive; others may seem counter-intuitive at first glance, but they are instructive in that they sharpen our intuitions about secrets. Henceforth, we write S to refer to the statement $Secret_0(\phi,K,N,\psi,t)$.

First, it should be uncontroversial that a revelation of the secrecy of ϕ is a revelation of ϕ (at a time when there is at least one keeper). Consequently, a keeper does not believe that there is a nescient to whom the secrecy of ϕ is revealed.

Theorem 1 *The following statements follow from* Σ.

1. $R(x,S \wedge \exists y Mem(y,K,t),t) \rightarrow R(x,\phi,t)$
2. $S \wedge Mem(x,K,t) \rightarrow \neg B(x,\exists y[Mem(y,N,t) \wedge R(y,S \wedge \exists z Mem(z,K,t),t)],t)$

Beliefs, intentions and revelations of a group g are inherited by every *subgroup* thereof. Where $g \sqsubseteq g' =_{\text{def}} \forall t,x[Mem(x,g,t) \rightarrow Mem(x,g',t)]$, the following follows.

Lemma 1 *The following statements are entailed by* Σ.

1. $(\forall x,t[Mem(x,g',t) \rightarrow B(x,\phi,t)] \wedge g \sqsubseteq g') \rightarrow \forall y,t[Mem(y,g,t) \rightarrow B(y,\phi,t)]$
2. $(\forall x,t[Mem(x,g',t) \rightarrow I(x,\phi,t)] \wedge g \sqsubseteq g') \rightarrow \forall y,t[Mem(y,g,t) \rightarrow I(y,\phi,t)]$
3. $(\forall x,t[Mem(x,g',t) \rightarrow R(x,\phi,t)] \wedge g \sqsubseteq g') \rightarrow \forall y,t[Mem(y,g,t) \rightarrow R(y,\phi,t)]$

Hence, secrets are also inherited by subgroups (assuming that the keepers' sub-group is not necessarily forever empty and every member of it believes that the nescients' sub-group is not necessarily forever empty). It follows that, given two secrets with the same secretum and secrecy condition, the intersection of the keepers is keeping the secret from the union of the nescients and the union of the keepers is keeping the secret from the intersection of the nescients.

Theorem 2 *The following are entailed by* Σ.

1. $S \wedge (K' \sqsubseteq K) \wedge (N' \sqsubseteq N) \wedge \neg NFE(K',t) \wedge \forall x[Mem(x,K',t) \rightarrow B(x,\neg NFE(N',t),t)] \rightarrow Secret_0(\phi,K',N',\psi,t)$

2. $Secret_0(\phi, K1, N1, \psi, t) \wedge Secret_0(\phi, K2, N2, \psi, t) \rightarrow$
$[\neg NFE(K1 \sqcap K2, t) \rightarrow Secret_0(\phi, K1 \sqcap K2, N1 \sqcup N2, \psi, t)] \wedge$
$[\forall x[Mem(x, K1 \sqcup K2, t) \rightarrow B(x, \neg NFE(N1 \sqcap N2, t), t)]$
$\rightarrow Secret_0(\phi, K1 \sqcup K2, N1 \sqcap N2, \psi, t)]$

Given that keepers are consistent believers, the secrecy condition ψ must, at any time t, be consistent with each keeper's beliefs and intentions, lest the group of keepers happens to be empty at t. Given that keepers believe the secretum and certain properties on the nescients and what is revealed to them the following theorem highlights some particularly important aspects of this constraint on the secrecy condition.

Theorem 3 *The following follows from* Σ.
$S \rightarrow \neg\exists x[Mem(x, K, t) \wedge B(x, \psi \rightarrow [\neg\phi \vee NFE(N, t) \vee \mathscr{B}(\phi, x, N, t) \vee \neg\mathscr{I}(\phi, x, N, \psi, t)], t)]$

The next theorem captures the intuition that secreta should not be bound to be revealed to the nescients while the secrecy condition holds. (This includes the trivial case where the secretum is a tautology.)

Theorem 4 *The following follows from* Σ.

$$S \rightarrow \neg\exists x[Mem(x, K, t) \wedge B(x, \exists y, t'[t < t' \wedge Mem(y, N, t') \wedge \forall t''[t < t'' \leq t' \rightarrow H(\psi, t'')] \wedge R(y, \phi, t')], t)]$$

The clauses of the following theorem indicate that, given $Secret_0(\phi, K, N, \psi, t)$, under certain conditions some propositions, other than ϕ, are also secreta or are believed to be secreta by members of K.

Theorem 5 *The following statements follow from* Σ.

1. $S \wedge Secret_0(\xi, K, N, \psi, t) \rightarrow Secret_0(\phi \wedge \xi, K, N, \psi, t)$
2. $S \rightarrow Secret_0(\exists x R(x, \phi, t), K, N, \psi, t)$
3. $B(x, S \wedge Mem(x, K, t), t) \rightarrow Secret_0(\phi, [x], N, \psi, t)$
4. $S \wedge Mem(x, K, t) \rightarrow B(x, Secret_0(\phi, [x], N, \psi, t), t)$
5. $S \wedge \forall x[Mem(x, K, t) \rightarrow B(x, S \wedge \exists y[Mem(y, K, t)], t)] \rightarrow$
$Secret_0(S \wedge \exists y[Mem(y, K, t)], K, N, \psi, t)$
6. $S \wedge Mem(x, K, t) \rightarrow B(x, Secret_0(Secret_0(\phi, [x], N, \psi, t), [x], N, \psi, t), t)$

The first two clauses should be obvious enough: the conjunction of two secreta is a secretum and so is the revelation of a secretum to some agent. According to the third clause, an agent who believes that there is a secret of some group, and that they are a member of that group, is actually holding the secret. This is so even though the agent may be mistaken about the group's holding the secret or about their membership in the group. Clause 4 indicates that each secret keeper believes that the secret is kept by the group to which only they belong. This is the closest we can get to an *introspection* result for secrets; in particular, this keeper may be keeping the secret but is not aware of the existence of the group or of its keeping the secret. However, as per the fifth clause, if every secret keeper is aware of the existence of the group and of its keeping the secret, then the secrecy of the secretum is itself a secretum of the same group of keepers from the same group of nescients under the same secrecy condition. Nevertheless, by Clause 6, even in this case where the keepers are aware of the group secret, they might still not believe in the secrecy of the secret for the group, simply because they may fail to believe that other keepers are aware of the secret. Hence, we can only prove a result akin to Clause 4.[8]

The following theorem presents *separation* results about K and N. First, no secret keeper believes that they are a nescient (Clause 1). Hence, no keeper is a nescient if the identity of nescients is known by each keeper (Clause 2). On the other hand, it may happen that an agent A who was once a secret keeper becomes a nescient. (Imagine players of team A keeping a secret from team B and at some later time an A player joins team B) This, however, does not mean that the secret is no longer kept; there are at least three reasons for this. First, the current secret keepers may not be aware of this conversion of their old co-keeper; second, they may not be aware that A *was* a co-keeper; and, third, it may be the case that the secretum, though once believed by A, is no longer revealed to them. While this last possibility is indeed moot, we do not want to commit to the permanence of revelation. However, assuming that a secret keeper is aware of the relevant facts and believes in the persistence of revelation, a contradiction is inevitable (Clause 3, where S' is just like S with t replaced by t'.)

Theorem 6 *The following follow from* Σ.

1. $S \wedge Mem(x,K,t) \rightarrow \neg B(x, Mem(x,N,t),t)$
2. $S \wedge Mem(x,K,t) \wedge \forall y[Mem(y,N,t) \rightarrow B(x, Mem(y,N,t),t)] \rightarrow \neg Mem(x,N,t)$
3. $[S \wedge Mem(x,K,t') \wedge B(x, t \leq t' \wedge [R(C,\phi,t) \rightarrow R(C,\phi,t')] \wedge Mem(x,K,t') \wedge S \wedge Mem(C,K,t) \wedge Mem(C,N,t'),t') \rightarrow B(x,\neg S',t')]$

Finally, if a secret keeper, A, believes that nescients believe that ξ implies the secretum ϕ, then A does not believe that ξ is revealed to any nescient and they do not intend to reveal it as long as the secrecy condition holds. Note, however, that ξ need not be a secretum since it is possible that A does not believe it.

Theorem 7 *The follows follows from* Σ.

$$S \wedge Mem(x,K,t) \wedge B(x, \forall y[Mem(y,N,t) \rightarrow \neg B(y,\neg\xi,t) \wedge B(y, \xi \rightarrow \phi,t)],t) \rightarrow \neg\mathscr{B}(\xi,x,N,t) \wedge \neg\mathscr{I}(x, \forall y,t'[t \leq t' \wedge Mem(y,N,t') \wedge \forall t''[t \leq t'' \leq t' \rightarrow H(\psi,t'')] \rightarrow R(y,\xi,t')],t)$$

6. A Typology of secrets

Secret$_0$ is only a bare-bones and, hence, weak notion of secrecy. We intuitively think of most secrets as involving stronger conditions. Five such stronger notions of secrecy are shown in Figure 3; all imply the bare-bones notion. *Secret*$_1$ is a secret in which the secretum is indeed not revealed to the nescients; *Secret*$_2$ is a secret of which keepers are aware; and *Secret*$_3$ holds when the keepers believe that the secretum ϕ is indeed not revealed to the nescients. In a *Secret*$_4$ situation, keepers are aware of the identity of all keepers, while, in a *Secret*$_5$ situation, they are aware of their membership in the group and believe that all keepers are aware of the secret.

[8] For most common cases of secrecy, stronger results can be proven since, in such cases, keepers are typically aware of the existence of K and of their membership thereof. In particular, we can prove that if the condition of membership in K is mere revelation of the secretum (which is typical of many secrets) each keeper believes that they are a member of K: $S \wedge Mem(x,K,t) \wedge B(x, \forall y[Mem(y,K,t) \leftrightarrow R(y,\phi,t)],t) \rightarrow B(x, Mem(x,K,t),t)$.

1. $Secret_1(\phi, K, N, \psi, t) =_{def} S \wedge \forall y[Mem(y, N, t) \rightarrow \neg R(y, \phi, t)]$
2. $Secret_2(\phi, K, N, \psi, t) =_{def} S \wedge \forall x[Mem(x, K, t) \rightarrow B(x, S, t)]$
3. $Secret_3(\phi, K, N, \psi, t) =_{def}$
   $S \wedge \forall x[Mem(x, K, t) \rightarrow B(x, \forall y[Mem(y, N, t) \rightarrow \neg R(y, \phi, t)], t)]$
4. $Secret_4(\phi, K, N, \psi, t) =_{def}$
   $S \wedge \forall x, y[Mem(x, K, t) \rightarrow [Mem(y, K, t) \leftrightarrow B(x, Mem(y, K, t), t)]]$
5. $Secret_5(\phi, K, N, \psi, t) =_{def}$
   $S \wedge \forall x[Mem(x, K, t) \rightarrow B(x, Mem(x, K, t) \wedge \forall y[Mem(y, K, t) \rightarrow B(y, S, t)], t)]$

Figure 3. Some common stronger notions of secrecy

Henceforth, we write S_n where is n is 1,2,3,4 or 5 referring to the corresponding secret type. Perhaps most common secrets are instances of all five types, satisfying $\bigwedge_{i=1}^{5} S_i$. These types are not totally independent though, as demonstrated by the following theorem. The first clause states that S_2 and S_3 hold if and only if there is a secret and every secret keeper believes the secret and that the secretum is not revealed to any nescient (S_1). By Clause 2, S_2 follows immediately from S_5. If all the keepers are unmistakably aware of one another (S_4) then S_5 holds if and only if all keepers believe both the secret and that all co-keepers believe the secret (S_2). Of particular interest is the fourth clause which indicates that S is equivalent to S_3 in case the secrecy condition implies (or *is*) that the secretum is not revealed to a nescient, which is a quite common condition of secrecy.

Theorem 8

1. $\Sigma \vdash S_2 \wedge S_3 \leftrightarrow S \wedge \forall x[Mem(x, K, t) \rightarrow B(x, S_1, t)]$
2. $\Sigma \vdash S_5 \rightarrow S_2$
3. $\Sigma \vdash S_4 \rightarrow [S_5 \leftrightarrow \forall x[Mem(x, K, t) \rightarrow B(x, S_2, t)]]$
4. If $\psi \vdash \forall y[Mem(y, N, t) \rightarrow \neg R(y, \phi, t)]$*, then* $\Sigma \vdash S \rightarrow S_3$

The clauses of Theorem 9 state that, depending on the secret type, secret keepers are bound to have certain properties (mostly beliefs). In an S_1 situation, we get complete separation of the groups of keepers and nescients (Clause 1); this separation is only a belief of each keeper in an S_2 situation (Clause 2). Similar results are indicated by Clauses 3 and 4 but with respect to the nescients' not believing the secret. The fifth clause states that, given S_2, every keeper holds the *de dicto* belief that members of K are individually keeping the secret. The sixth clause states that the same belief is held *de re* if both S_2 and S_4 hold.

Theorem 9 *The following follow from* Σ*.*

1. $S_1 \rightarrow [Mem(x, K, t) \rightarrow [\neg Mem(x, N, t)]]$
2. $S_2 \rightarrow [Mem(x, K, t) \rightarrow B(x, \neg\exists y[Mem(y, K, t) \wedge Mem(y, N, t)], t)]$
3. $S_1 \rightarrow [\forall y[Mem(y, N, t) \rightarrow \neg B(y, S_0 \wedge \exists z Mem(z, K, t), t)]]$
4. $S_3 \rightarrow [Mem(x, K, t) \rightarrow B(x, \forall y[Mem(y, N, t) \rightarrow \neg B(y, S_0 \wedge \exists z[Mem(z, K, t)], t)], t)]$
5. $S_2 \rightarrow [Mem(x, K, t) \rightarrow B(x, [Mem(y, K, t) \rightarrow Secret_0(\phi, [y], N, \psi, t)], t)]$
6. $S_4 \wedge S_2 \rightarrow [Mem(x, K, t) \rightarrow [Mem(y, K, t) \rightarrow B(x, Secret_0(\phi, [y], N, \psi, t), t)]]$

7. Related Work

Philosophical and psychological investigations of secrets are best represented by the work of Bok in philosophy [1] and Kelly's book [2] and the work of Slepian et al [3,22, for example] in psychology. These authors share our intuition that secrecy is mostly about the intention to conceal. Their interests in secrets are different from ours though; they are primarily interested in ethical issues related to secrets [1] and in the motivations for and the psychological effects of keeping secrets [2,3,22]

Logical accounts of secrecy abound in the literature on system security [5,6,7,8,9, 10,11,12,13,14, for instance]. Much of this literature is rooted in, or best represented by, the work of Halpern and O'Neill [9]. The authors consider multi-agent systems with a branching time structure where, at any time, each agent is in some *local state* comprising all the information accessible to the them. Agents are never mistaken about their local states; they never hold false beliefs. Using this machinery, Halpern and O'Neill define several notions of secrecy. The most fundamental of these, *total secrecy*, is defined as follows. The actual local state of agent j is totally secret from agent i if i cannot "rule out" any possible local state of j.

The usefulness of this account, and of most other accounts in the literature [11,13, for example], is based on a couple of assumptions:

1. **Local states are typically a collection of assignments of values to variables.** If said variables correspond to propositions, then, assuming a classical bivalent logic, there can only be two values: true and false. Hence, not being able to rule out a local state amounts to not being able to decide whether a proposition is true or false.
2. **Agents cannot hold false beliefs.** As pointed out above, this is built into the theory. Hence, given the first point, if we think of secrets as propositions, a proposition can only be a secret from agent i if i is in suspense about the proposition.
3. **Systems can be constructed.** The assumption here is that it is always possible to fully characterize the system as a branching tree of states. This is probably always possible if systems are programs or simple database transactions [11, for example].

Given our objective to characterize secrecy in an unconstrained, commonsense setting, we cannot uphold any of the above assumptions. First, since we consider objects of secrecy to be only propositions, assumption 1 reduces to the case of variables with binary domains. Second, we cannot in general make the unrealistic assumption that agents have no false beliefs; assumption 2 does not allow us to account for situations where j keeps P a secret from i who believes $\neg P$. Third, in a general theory of secrets allowing all forms of complex social interactions, the assumption of a system which is constructible as a branching tree of states is at least questionable.

The revelation modality we introduced does not seem to have a thoroughly investigated precedent. It is perhaps possible to use a variant of the notion of *announcement* from dynamic epistemic logic [30,31] to model *acts* of revealing. This would, however, require, possibly extensive, revision of the principles underlying the logic of announcement. In particular, a *truthful announcement* of P results in the addressee's believing P (at least if P is atomic) and a *possibly lying announcement* thereof causes the addressee to believe that the announcer believes P [30,31]. Even if we adopt the latter, more cautious attitude towards announcement as a model of revelation, we are restricted to revelations made only by cognitive agents which may have beliefs. Our passive notion of revelation does not require this and is consistent with a revelation resulting from a simple act of perception not involving an announcing agent.

8. Conclusion

We presented foundations for a logical, commonsense theory of secrets. A secret is construed as a situation in which a group of secret keepers believe a proposition, which they do not believe to have been revealed to members of another group of nescients. Crucially, the keepers intend that this concealment from the nesciencts persists so long as some condition of secrecy holds. To that end, a non-normal modal operator for revelation was identified together with axioms relating it to belief. Further, towards an ontology of secrets, various types of secrets were identified, all of which including the bare-bones definition in addition to some extra common conditions. Several properties which sharpen our intuitions about secrets were proven and more are to be investigated in future work.

9. Acknowledgment

We are thankful to our colleagues Nourhan Ehab, Nada Sharaf and Ammar Yasser for their comments on earlier versions of the paper. We would also like to express our deepest gratitude to the reviewers for their helpful suggestions.

References

[1] Bok S. Secrets: On the Ethics of Concealment and Revelation. Vintage; 1989.
[2] Kelly AE. The psychology of secrets. Springer Science & Business Media; 2002.
[3] Slepian ML, Chun JS, Mason MF. The experience of secrecy. Journal of Personality and Social Psychology. 2017;113(1):1.
[4] Vincent J, Taipale S, Sapio B, Lugano G, Fortunati L, editors. Social Robots from a Human Perspective. Switzerland: Springer; 2015.
[5] Bieber P. A logic of communication in hostile environments. In: Proceedings of the Computer Security Foundations Workshop III; 1990. p. 14–22.
[6] Glasgow J, MacEwen G, Panangaden P. A logic for reasoning about security. In: Proceedings of the Computer Security Foundations Workshop III; 1990. p. 2–13.
[7] Cuppens F. A logical formalization of secrecy. In: Proceedings of the Computer Security Foundations Workshop VI; 1993. p. 53–62.
[8] Roy A, Datta A, Derek A, Mitchell JC, Seifert JP. Secrecy analysis in protocol composition logic. In: Annual Asian Computing Science Conference. Springer; 2006. p. 197–213.
[9] Halpern JY, O'Neill KR. Secrecy in multiagent systems. ACM Transactions on Information and System Security (TISSEC). 2008;12(1):5:1–5:47.
[10] More SM, Naumov P. An Independence Relation for Sets of Secrets. Studia Logica. 2010;94(1):73–85.
[11] Biskup J, Tadros C. Preserving confidentiality while reacting on iterated queries and belief revisions. Annals of Mathematics and Artificial Intelligence. 2015;73:75–123.
[12] Tao J, Slutzki G, Honavar V. A Conceptual Framework for Secrecy-preserving Reasoning in Knowledge Bases. ACM Transactions on Computational Logic. 2014;16(1):3:1–3:32.
[13] Tsukada Y, Sakurada H, Mano K, Manabe Y. On compositional reasoning about anonymity and privacy in epistemic logic. Annals of Mathematics and Artificial Intelligence. 2016;78:101–129.
[14] Ramezanifarkhani T, Owe O, Tokas S. A secrecy-preserving language for distributed and object-oriented systems. Journal of Logical and Algebraic Methods in Programming. 2018;99:1–25.
[15] Marshall D, Weatherson B. Intrinsic vs. Extrinsic Properties. In: Zalta EN, editor. The Stanford Encyclopedia of Philosophy. spring 2018 ed. Metaphysics Research Lab, Stanford University; 2018. .
[16] Link G. Algebraic Semantics in Language and Philosophy. Stanford, CA: CSLI Publications; 1998.
[17] Landman F. Groups I. Linguistics and Philosophy. 1989;12(5):559–605.
[18] Galton A, Wood Z. Extensional and intensional collectives and the *de re/de dicto* distinction. Applied Ontology. 2016;11(3):205–226.
[19] Ismail HO. Stability in a Commonsense Ontology of States. In: Proceedings of the Eleventh International Symposium on Commonsense Reasoning (Commonsense-2013). Agya Napa, Cyprus; 2013. .

[20] The United States Department of Justice. Declassification; 2018. https://www.justice.gov/archives/open/declassification.
[21] Lane JD, Wegner DM. The cognitive consequences of secrecy. Journal of personality and social psychology. 1995;69(2):237.
[22] Slepian ML, Greenaway KH. The benefits and burdens of keeping others' secrets. Journal of Experimental Social Psychology. 2018;78:220–232.
[23] Mahon JE. The Definition of Lying and Deception. In: Zalta EN, editor. The Stanford Encyclopedia of Philosophy. winter 2016 ed. Metaphysics Research Lab, Stanford University; 2016. .
[24] Cohen PR, Levesque HJ. Intention is choice with commitment. Artificial intelligence. 1990;42(2-3):213–261.
[25] Beaver DI. Presupposition and Assertion in Dynamic Semantics. Stanford, CA: CSLI Publications; 2011.
[26] Fagin R, Halpern J. Belief, awareness, and limited reasoning. Artificial Intelligence. 1987;34(1):39–76.
[27] van Benthem J, Fernandez-Duque D, Pacuit E. Evidence and plausibility in neighborhood structures. Annals of Pure and Applied Logic. 2014;165(1):106–133.
[28] Bennett B, Galton AP. A unifying semantics for time and events. Artificial Intelligence. 2004;153(1-2):13–48.
[29] Pacuit E. Neighborhood semantics for modal logic. Springer; 2017.
[30] van Ditmarsch H, van der Hoek W, Kooi B. Dynamic Epistemic Logic. Dordrecht: Springer; 2008.
[31] Van Ditmarsch H. Dynamics of lying. Synthese. 2014;191(5):745–777.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200663

An Ontology for Formal Models of Kinship

Carmen CHUI [a], Michael GRÜNINGER [a], Janette WONG [b]

[a] *Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario, Canada*

[b] *Data & Analytics (DNA), Technology & Operations, Royal Bank of Canada, Toronto, Ontario, Canada*

Abstract. The near ubiquity of family relationship ontologies in the Semantic Web has brought on the question of whether any formal analysis has been done in this domain. This paper examines kinship relationships that are normally overlooked in formal analyses of domain-specific ontologies: how are such ontologies verified and validated? We draw inspiration from existing work done in anthropology, where attempts have been made to formally model kinship as atemporal algebraic models. Based on these algebraic models, we provide an ontology for kinship written in first-order logic and demonstrate how the ontology can be used to validate definitions found in Canadian legal laws and data collection documentation.

Keywords. algebraic model, anthropology, kinship, kin term map, relationships, domain ontology

1. Introduction

Despite the near ubiquity of family relationship ontologies within the Semantic Web community, there has been no *formal* analysis of any ontology for this domain that captures familial relationships in anthropology and in legal texts. Often used as an example to illustrate how to develop an ontology in the Web Ontology Language (OWL)[1], there is the *misconception* that family relationships are easy to model in an ontology. The ubiquity of using family relationships gives the impression that this particular domain is trivial to axiomatize. However, extensive work has been done within anthropology which can serve as the basis for the validation of any kinship ontology. In this paper[2], we propose a kinship ontology written in first-order logic to formalize anthropologist Dwight Read's algebraic models of kinship and the associated intended semantics, and to support reasoning problems and queries across demographic datasets found in anthropology.

[1] For example, see the OWL 2 language guide for the Family History Knowledge Base (FHKB) in [1].

[2] An extended version of this paper containing the proofs for the theorems and verification of the ontology can be found online: `http://stl.mie.utoronto.ca/publications/fois_kinship_extended.pdf`

2. Motivation

The popular Family History Knowledge Base (FHKB) presented in [1] was designed to test the limits of OWL reasoners and to maximize the use of inference since it is mostly taxonomic in structure and contains very few non-subclass axioms. There is no ontological basis in its design: no requirements were proposed, no verification nor validation was done, and there was no analysis of its ontological commitments. Further, there have been no formal axiomatizations of notions of kinship outside of FHKB[3]. Instead, we look at the terminology used to describe *kinship* in anthropology and terminology found within legal documentation in the Canadian context.

We need to use *ontologies* to represent these different systems and legal definitions because we want to do data quality and other kinds of queries with respect to that data. Taxonomies are insufficient to carry out such tasks due to their inabilities to *define* additional concepts and provide *explicit* axioms to describe relationships between concepts.

In anthropology, there are established kinship systems that contain various terminology used to describe relationships between people. Kinship patterns were identified by Lewis Henry Morgan [5] and further categorized by George P. Murdock [6], both anthropologists who studied family and kinship structures across different cultures. These various kinship systems[4] and their relationships are depicted and defined graphically in Figure 1: circles and triangles denote female and male, respectively, and colours denote the various relationships with a label describing the relationship underneath. Different societies describe kinship relationships differently. For a more detailed discussion of these kinship systems and terminologies, we refer the reader to [6], [5], and [7].

Additionally, kinship relationships can be defined using Anthony F.C. Wallace and John Atkins' anthropological definitions in [8], shown in Table 1. Abbreviations for familial relationships are as follows: father (Fa), mother (Mo), brother (Br), sister (Si), son (So), and daughter (Da). These terms are most familiar in most English-speaking parts of the world, and are treated as primitives in the English language. More complex kin relationships are treated as the relative product of two or more primitive terms. For example, the definition of grandfather in the first row can be defined as "the father of father" (FaFa) and "the father of mother" (MoFa), while grandmother is defined as "the mother of father" (FaMo) and "the mother of mother" (MoMo).

Further, we are also interested in the legal applications of kinship[5]. In Canada, several legal laws and acts outline the limitations of marriage and in official data collection agencies; these include concepts found in Statistics Canada ('StatsCan')[6], the Marriage (Prohibited Degrees) Act[7], and the Civil Marriage Act[8]. We are interested in defining these legal concepts using a formal kinship ontology.

Additionally, work done in anthropology shows there is interest in representing the structures of kinship algebraically and as formal models. Figures 2 and 3 illustrate me-

[3]There are various other discussions on kinship and reasoning in [2, 3], as well as an ontology pattern presented in [4], but these discussions all pertain to reasoning with OWL.

[4]We also acknowledge other cultural systems for kinship, primarily the Chinese kinship system, which are more descriptive and involves the notion of ages and social institutions. It would be of interest to further extend the ontology presented in this paper to cover this kinship system.

[5]Since the authors live in Canada, the Canadian legal context for these terms and definitions are of interest.

[6]`https://www.statcan.gc.ca/eng/concepts/index`

[7]`https://laws-lois.justice.gc.ca/eng/acts/M-2.1/page-1.html`

[8]`https://laws-lois.justice.gc.ca/eng/acts/c-31.5/page-1.html`

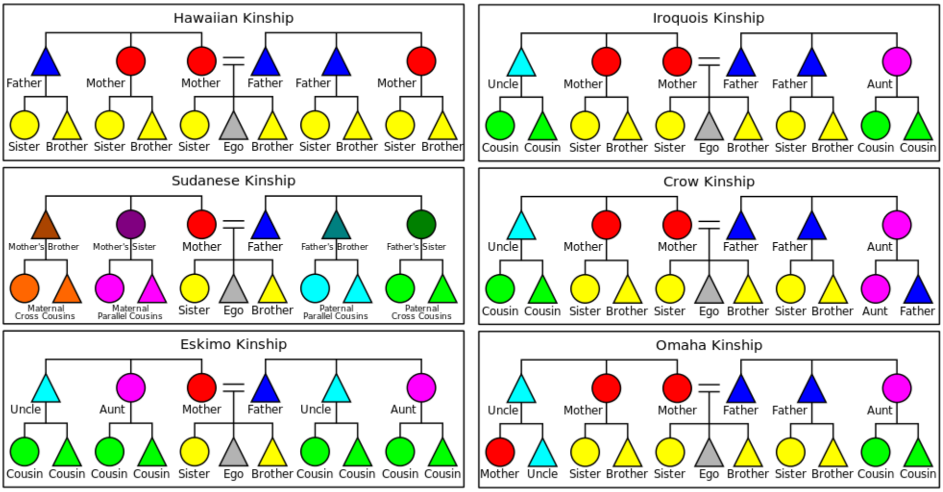


Figure 1. Basic kinship classification systems identified by Murdock in [6]. Triangles and circles denote sex (male/female). Colours denote the different types of relationships in each system systems, with a label for the relationship underneath.

Table 1. Example kinship definitions presented by Wallace and Atkins in [8].

Terminology		**Definition**
grandfather	:	FaFa, MoFa
grandmother	:	FaMo, MoMo
grandson	:	SoSo, DaSo
granddaughter	:	SoDa, DaDa
uncle	:	FaBr, MoBr, FaFaBr, MoFaBr, etc.
aunt	:	FaSi, MoSi, FaFaSi, MoFaSi, etc.
cousin	:	FaBrSo, FaBrDa, MoBrSo, MoBrDa, FaSiSo, FaSiDa, MoSiSo, MoSiDa, FaFaBrSo, FaMoBrSo, MoFaSiDa, etc.
nephew	:	BrSo, SiSo, BrSoSo, SiSoSo, etc.
niece	:	BrDa, SiDa, BrDaDa, SiDaDa, etc.

diation and algebraic structures (kin term map) found in [9–11]. In particular, Read et al. have developed a Kinship Algebraic Expert System (KAES) in [10] which takes kinship terminology and algebraically constructs kin term maps and genealogical diagrams (family trees) of the resulting kin term maps. As we will see later, the algebraic structures produced by Read et al. in [10] and [11] can be used to help formalize kinship relationships as definable relations. Figure 2 outlines *mediation structures* used to describe kinship in [11,12]: these mathematical structures are used to relate two, otherwise unrelated, conceptual categories together using a mediating category[9]. In Figure 2a, a structure for a family with one child is presented: the three categories are shown as the Mother, Father, and Child boxes, each with their own gender attributes. The *spouse* relation links the Mother and Father categories, and are linked to the Child category by the mother

[9]In the context of [12] and [11], the term 'category' refers to *conceptual categories*, which in our ontology are formalized as classes.

and father relations, respectively. Similarly, Figure 2b illustrates a structure for a family with two children and the inclusion of a sibling relation between offspring. As we will see, these mediation structures can be axiomatized with the kinship ontology presented in this paper.

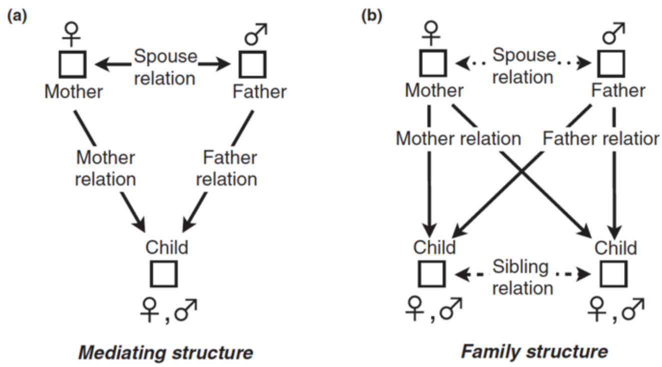


Figure 2. Read's mediation structures for kinship. (a) shows a mediation structure for a family with one child, and (b) shows a mediation structure for a family with two children with the inclusion of a sibling relation. (Figure 2 from [11])

Furthermore, a formal outline of how to generate algebraic kinship structures can be found in [13], where Read asserts that an algebraic structure constructed from kinship terminology is isomorphic to the kin term map structure. He presents a construction methodology[10] that maps the kin terminology with the kin term map. In Section 4, we show how the mediation structures presented in [10, 11] correspond to mathematical graph structures used to verify the kinship ontology presented in this paper.

In the sections that follow, we provide an overview of our axiomatization of Read's algebraic structures and show how our first-order ontology can describe kinship and familial relationships. Regardless of which anthropological kinship system is used to describe relationships, the ontology is sufficient to axiomatize the following: the terms found in each system, the intended semantics of the algebraic structures presented by Read in [11], and definitions in Canadian legal documentation.

3. The Kinship Ontology ($T_{kinship}$)

The idea of representing the *binary* relationships in [11] drives our interest in developing a first-order ontology that sufficiently captures these anthropological concepts of kinship. Herewith we present the kinship ontology, $T_{kinship}$, in first-order logic[11]. The ontology is designed with the mediation structures and kinship term maps from [11] in mind: our focus is on the various kinship relationships presented in anthropology, as these structures have been already established in that field. We emphasize here that we are axiomatizing Read's structures and do not introduce any bias in how these relationships should be axiomatized. Our approach differs from existing work done with ontologies and the

[10]This outlined in detail in Box 13.2 in [13], and in [9].
[11]Available online: `http://colore.oor.net/kinship/`

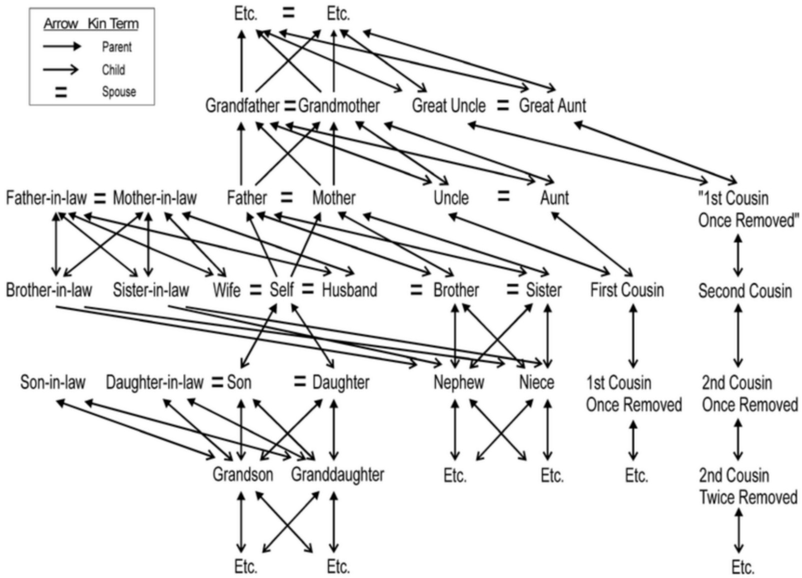


Figure 3. Kin term map that outlines the various relationships. (Figure 5a from [11])

discussion on the purpose and notion of 'roles' (such as role identities discussed in [14] and [15]).

Instead of focusing on which kinship relationship types should be considered as roles, we have taken the existing anthropological kinship terms and have axiomatized them as *binary* relations within the ontology. Further, we present a set of *atemporal* axioms for kinship as the algebraic models presented in [11] are independent of time; while spouses and other relationships may change over time, these changes need to be reflected in a temporal version of the kinship ontology: this is left for a future iteration of the ontology that includes the adoption of relevant time and event ontologies.

The algebra presented in [11] contains one substructure for consanguineal relations (which arise from ancestral lineage) and one substructure for affinal relations (which arise through marriage). The signature of $T_{kinship}$ therefore consists of the two *primitives*: the affinal $hasSpouse(x,y)$ and the consanguineal $ancestorOf(x,y)$ relations, which are read as "x has spouse y" and "x is the ancestor of y," respectively. The axioms of the ontology are organized into the following sets of Common Logic Interchange Format (CLIF) files depicted in Figure 4:

- $T_{ancestor}$ contains axioms pertaining to ancestors (Axioms (1) to (8) in Figure 5).
- T_{spouse} contains axioms pertaining to spouses (Axioms (9) to (12) in Figure 5).
- $T_{kinship}$ imports $T_{ancestor}$ and T_{spouse}, with additional axioms that combine spouses and ancestors.

3.1. Ancestors and Children

Approaches to define kinship relations, such as those presented in [11], begin with the parent/child relation, and then define all other relations through composition. However,

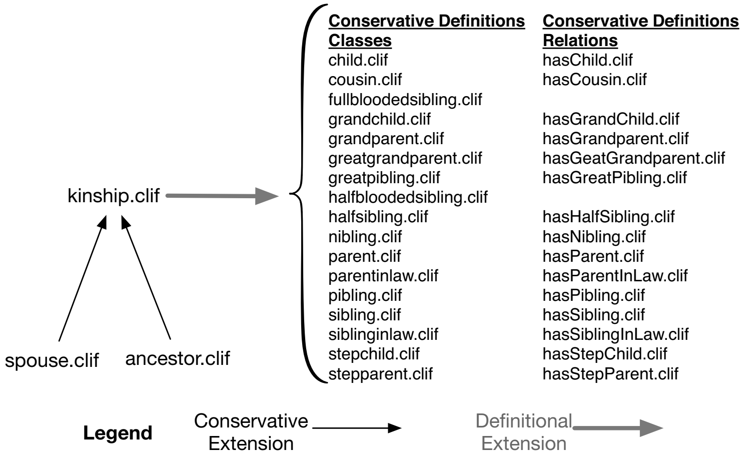


Figure 4. Hierarchy organization in COLORE. Theory names denote the CLIF file names found in the repository. Solid arrows denote conservative extension, dotted arrows denote non-conservative extension, and bolded solid grey arrows denote definitional extension.

the partial ordering over ancestors and descendants is not first-order definable using the $hasChild(x,y)$ relation (in the same way that a discrete linear ordering is not first-order definable using a successor relation). Partial orders are not first-order definable by a theory whose signature consists only of *successor* (see [16]). On the other hand, a successor relation, such as $hasChild(x,y)$, is definable in a discrete partial order using the $ancestorOf(x,y)$ relation. Consequently, $ancestorOf(x,y)$ was selected as a primitive in the ontology.

3.2. Ancestors, Spouses, and Unintended Models

The axioms in $T_{ancestor}$ and T_{spouse} alone are not sufficient. We also need to specify additional constraints between the $ancestorOf(x,y)$ and $hasSpouse(x,y)$ relations, to prevent scenarios such as where grandparents are the spouses of their grandchildren in the models of the ontology. We need axioms that limit how two persons in the domain of the ontology can be related by marriage using $hasSpouse(x,y)$ and by parentage using $ancestorOf(x,y)$. In order to eliminate such unintended models where people with familial relationships would become spouses or parents of each other, we need to introduce an ordering relation to differentiate between ancestors and descendants. Such unintended relationships would be having two siblings becoming spouses, or a grandchild marrying their grandparent. We want to best represent situations that are bound by Canadian laws and thereby adhere to the Criminal Code (R.S.C., 1985, c. C-46)[12], which outlines the conditions for incest when the relationships between two people are by blood. These sorts of relationships are illegal when they are between a person and their parent, child, sibling, grandparent, or grandchild.

Without constraints to limit incestuous relationships in $T_{kinship}$, this would cause models of the ontology to contain circular relationships. For example, this would result in models where a grandparent can be the child of their own child[13], or have grand-

[12] https://laws-lois.justice.gc.ca/eng/acts/C-46/section-155.html
[13] http://colore.oor.net/kinship/output/kinship_greatgrandparents_unintended.model

parents be the spouses of their own grandchild[14]. To eliminate such *unintended* models from the ontology, we use a *discrete partial order* to constrain how the elements of the $hasChild(x,y)$ and $hasSpouse(x,y)$ relations can interact. The $ancestorOf(x,y)$ relation is used to *order* the individuals in the ontology – we utilize the notion of *ordering* found in mathematics and order theory. Axioms (1) to (8) in Figure 5 outlines the axioms used to handle ancestor relationships in the $T_{ancestor}$ module.

We impose the rather strong condition of Axiom 13, (which requires spouses to have no common ancestors) in part because it will be needed to capture the structure of Read's algebra. If the ontology is used for data cleaning, this axiom can be relaxed to allow families with spouses that share a common great-great-grandparent. For example, the British Royal Family consists of third cousins who have married one another: Queen Elizabeth II and Prince Philip are both descendants of Queen Victoria.

$$(\forall x \forall y\, (ancestorOf(x,y) \supset (person(x) \wedge person(y)))). \tag{1}$$

$$(\forall x\, (\neg ancestorOf(x,x))). \tag{2}$$

$$(\forall x \forall y \forall z\, ((ancestorOf(x,y) \wedge ancestorOf(y,z)) \supset ancestorOf(x,z))). \tag{3}$$

$$(\forall x \forall y\, (ancestorOf(x,y) \supset \neg ancestorOf(y,x))). \tag{4}$$

$$\begin{aligned}(\forall x \forall y\, (hasChild(x,y) \equiv (&ancestorOf(x,y)\\ &\wedge \neg(\exists z\, (ancestorOf(x,z) \wedge ancestorOf(z,y)))))).\end{aligned} \tag{5}$$

$$(\forall x \forall y\, (ancestorOf(x,y) \supset (\exists z\, (hasChild(x,z) \wedge (ancestorOf(z,y) \vee (y=z)))))). \tag{6}$$

$$(\forall x \forall y ((ancestorOf(x,y) \supset (\exists z (hasChild(z,y) \wedge (ancestorOf(x,z) \vee (x=z))))))). \tag{7}$$

$$\begin{aligned}(\forall x \forall y \forall z \forall u\, (&ancestorOf(u,y) \wedge ancestorOf(z,y) \wedge ancestorOf(x,u) \wedge ancestorOf(x,z)\\ &\supset (ancestorOf(u,z) \vee ancestorOf(z,u) \vee (z=u)))).\end{aligned} \tag{8}$$

$$(\forall x \forall y\, (hasSpouse(x,y) \supset (person(x) \wedge person(y)))). \tag{9}$$

$$(\forall x\, (\neg hasSpouse(x,x))). \tag{10}$$

$$(\forall x \forall y\, (hasSpouse(x,y) \supset hasSpouse(y,x))). \tag{11}$$

$$(\forall x \forall y \forall z\, (hasSpouse(x,y) \wedge hasSpouse(x,z) \supset (y=z))). \tag{12}$$

$$(\forall x \forall y \forall z\, ((hasSpouse(x,y) \wedge ancestorOf(z,x)) \supset \neg ancestorOf(z,y))). \tag{13}$$

Figure 5. Axioms for $T_{kinship}$.

3.3. Doesn't Everyone Have A Parent?

We **do not** include this axiom in the ontology:

> A person has a parent who is a person. (For every person, there is another person who is their parent.)

$$(\forall x\, (person(x) \supset (\exists y\, (person(y) \wedge hasParent(x,y) \wedge (x \neq y))))). \tag{14}$$

[14] http://colore.oor.net/kinship/output/kinship_grandparentspouse_unintended.model

Intuitively, this axiom makes sense in real life. However, the knowledge base or ontology of persons should only reflect the elements one wants to examine. We might care about Bob and Alice, but not necessarily Bob's parents or Alice's parents: we do not necessarily need to know the parents of a particular person. It is possible to have an ancestor of a person without them being a parent of that person. Furthermore, this axiom creates *infinite* models when using a model finder like Mace4[15]: for every person in the knowledge base, the program will continually generate more and more elements in the model and will never terminate. Consequently, a model will never be outputted by a model finder.

3.4. What About Gender?

The approach we have taken to axiomatize $T_{kinship}$ is *independent* of gender. Within kinship systems in anthropology, a strict binary gender system of male and female is adopted: this is particularly noticeable in the kinship systems presented in Figure 1. Consequently, we can state that anthropologists have adopted an explicit binary gender ontology in their algebraic representations of relationships.

In contrast, $T_{kinship}$ does not have any inherent bias towards any gender ontology: in the axioms presented in Figure 5, we have made all binary relations as gender-neutral as possible. Due to the gender-neutral nature of the axioms of $T_{kinship}$, we can treat gender as an ontology module that can be imported into $T_{kinship}$ to allow us to make additional distinctions (such as male or female) in order for us to *faithfully interpret* existing work and models done by the anthropology community.

3.5. Kinship Relationships As Defined Relations

With $T_{kinship}$, we can axiomatize the kinship relationships presented in Figures 2 and 3 as *defined relations*. For example, this means that definitions for first cousins once- or twice-removed, and second and third cousins, can be easily axiomatized by extending the ontology with conservative definitions. In the hierarchy organization presented in Figure 4, these definitions are signified as definitional extensions with the bolded grey arrows and are in their own individual CLIF files in the repository. As we will see in Section 5, we can write definitions for classes (such as $grandparent(x)$, $cousin(x)$, $grandchild(x)$) and their corresponding binary relations (such as $hasGrandparent(x,y)$, $hasCousin(x,y)$, $hasGrandchild(x,y)$). For example, the $hasGrandparent(x,y)$ relation has the following definition:

$$(\forall x \forall z\,(hasGrandparent(z,x) \equiv (\exists y\,(hasChild(x,y) \land hasChild(y,z))))).$$

From this first-order definition, we can see that the bidirectional equivalence is not definable in OWL. In FHKB, the `grandparent` class is axiomatized as an OWL2 property chain, which allows an ontology user to infer the existence of a property from a chain of properties. This would appear as $hasParent \circ hasParent \sqsubseteq hasGrandParent$. With Read's algebraic approach presented in [10], the grandparent relationship is defined as $P^2 \leftrightarrow$ grandparent in algebraic logic, where P stands for parent. Note that the axioms that arise from the property chain and algebraic approaches correspond to *paths* within

[15] https://www.cs.unm.edu/~mccune/mace4/

the kin term maps shown in Figure 3. Consequently, we can state that defined relations in $T_{kinship}$ correspond to paths found in the kinship structures.

We can graphically depict this *model* of the ontology in Figure 6 as a consanguinity graph, where people are nodes and the lines between the nodes represent relationships between the people. Directional arrows indicate a parental relationship between nodes, where the tail-end is the parent node and the arrow-head is the child node (e.g., Lucy is the parent of Alice and Lucy is the child of Alice: $hasChild(Lucy, Alice)$ and $hasParent(Alice, Lucy)$). In order to determine whether two people are related to each other in the graph, all one needs to do is to find a *path*. It is possible for two elements in the graph to not be related at all: for example, Francisco has no relationship with anyone in Figure 6.

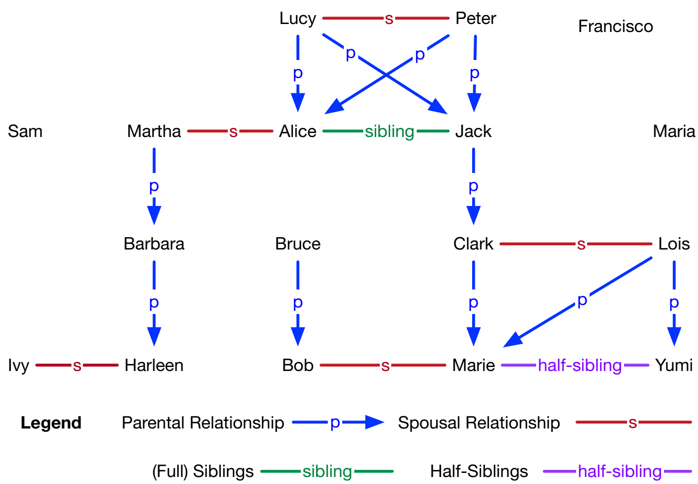


Figure 6. Graphical representation of how defined relations correspond to paths in the underlying consanguinity graph in the model of the ontology. Names of people are nodes of the graph and the lines between notes denote relationships.

To determine how Alice and Bob are related, we would simply have to find a path in the graph between *Bob* and *Alice*. From the example shown in Figure 6, this path would be the path from $hasSpouse(Bob, Marie)$, $hasChild(Clark, Marie)$, $hasChild(Jack, Clark)$, $hasChild(Peter, Jack)$, and $hasChild(Peter, Alice)$. Similarly, to determine if Sam and Jack are related to one another, examining the graph allows us to determine that no path between Sam and Jack can be found, so we can conclude that Sam and Jack are not related to each other.

3.6. Subgraphs as Examples

To show how defined relations can be further generalized, we consider the sibling relationships shown in Figure 7. These are *connected subgraphs* found from the example presented in Figure 6, which show how Alice and Jack are full siblings and that Marie and Yumi are half-siblings. With the ontology, we are able to *define new relations* to demonstrate these relationships. For example, we can define full-blooded siblings as having both parents in common. Conversely, half-siblings have one parent in common.

$$\forall x \forall y\, hasFullBloodedSibling(x,y) \equiv \exists w \exists y \exists z\, hasParent(x,y) \wedge hasParent(x,z) \wedge$$
$$hasParent(w,y) \wedge hasParent(w,z)$$
$$\forall x \forall w\, hasHalfSibling(x,w) \equiv \exists y \exists z\, hasParent(x,y) \wedge hasParent(x,z) \wedge$$
$$hasParent(w,y) \wedge \neg hasParent(w,z)$$

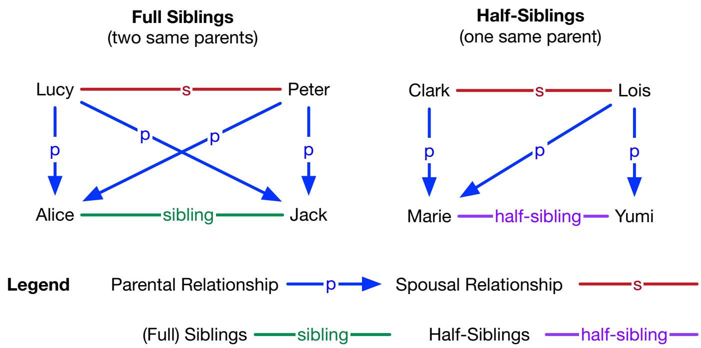


Figure 7. Differences between full- and half-siblings in the models of $T_{kinship}$.

Recall Figure 2 and notice how the models of $T_{kinship}$ resemble the mediation structures presented in [11]. This suggests that, in both cases, the intended structures focus on some underlying classes of graphs, leading to the twin issues of ontology verification and validation. In the sections that follow, we provide the verification of $T_{kinship}$, and the validation of $T_{kinship}$ with respect to the definitions of kinship relationships found in anthropology, the mediation structures developed by the KAES program, and StatsCan documentation.

4. Verification of $T_{kinship}$

Ontology verification is concerned with the relationship between the *intended* models of an ontology and the models of the axiomatization of the ontology. We characterize the models of an ontology up to isomorphism and determine whether these models are equivalent to the intended models of the ontology.

The intended structures for the $ancestorOf(x,y)$ relation is represented by a special class of partial orderings shown in Definition 1. We use the following notation for upper and lower sets from mathematics. For each $\mathbf{x} \in V$, the *upper set* is defined as:

$$U^{\mathbb{P}}[\mathbf{x}] = \{\mathbf{y} : \mathbf{x} \leq \mathbf{y}\}$$

The *lower set* is defined as:

$$L^{\mathbb{P}}[\mathbf{x}] = \{\mathbf{y} : \mathbf{y} \leq \mathbf{x}\}$$

$\mathscr{L}_{\mathbb{P}} = \langle V, E \rangle$ is the lower bound graph for $\mathbb{P}$:

$$(\mathbf{x},\mathbf{y}) \in E \quad L^{\mathbb{P}}[\mathbf{x}] \cap L^{\mathbb{P}}[\mathbf{y}] \neq \emptyset$$

Definition 1 *A partial ordering* $\mathbb{P} = \langle V, \leq \rangle$ *is lattice-free iff*

$$\langle L^{\mathbb{P}}[\mathbf{x}], \leq \rangle, \langle U^{\mathbb{P}}[\mathbf{x}], \leq \rangle$$

are semilinear orderings, for each $\mathbf{x} \in V$. $\mathfrak{M}^{lattice\_free}$ *denotes the class of discrete lattice-free partial orderings.*

Since the $hasSpouse(x,y)$ relation is symmetric and irreflexive, it is represented by a special class of simple graphs[16]:

Definition 2 *A scattered edge graph is a simple graph* $\mathbb{G} = \langle V, \mathbf{E} \rangle$ *such that*

$$\mathbb{G} \cong K_2 \cdot \overline{K_m}$$

$\mathfrak{M}^{scattered\_edge}$ *denotes the class of scattered edge graphs.*

Models of $T_{kinship}$ are represented by the amalgamation of lattice-free partial orderings and scattered edge graphs:

Definition 3 $\mathbb{P} \oplus \mathbb{G}$ *is a kinship mereograph iff:*

1. $\mathbb{P} = \langle V, \leq \rangle$ *such that* $\mathbb{P} \in \mathfrak{M}^{lattice\_free}$;
2. $\mathbb{G} = \langle V, \mathbf{E} \rangle$ *such that* $\mathbb{G} \in \mathfrak{M}^{scattered\_edge}$;
3. $\mathscr{L}^{\mathbb{P}}([\mathbf{x}]) \cap N^{\mathbb{G}}[\mathbf{x}] = \emptyset$, *for each* $\mathbf{x} \in V$.

$\mathfrak{M}^{kinship\_mereograph}$ *denotes the class of kinship mereographs.*

Examples of kinship mereographs can be seen in Figures 6 and 7, in which the red edges (spousal relationship) form the scattered edge graph $\mathbb{G}$ and the blue edges (parental relationship) correspond to the Hasse graph of the lattice-free partial ordering $\mathbb{G}$.

Theorem 1 *There exists a bijection* $\varphi : Mod(T_{kinship}) \rightarrow \mathfrak{M}^{kinship\_mereograph}$ *such that:*

1. $(\mathbf{x},\mathbf{y}) \in \mathbf{hasSpouse}$ *iff* $\mathbf{y} \in N^{\mathbb{G}}[\mathbf{x}]$;
2. $(\mathbf{x},\mathbf{y}) \in \mathbf{ancestorOf}$ *iff* $\mathbf{x} \in L^{\mathbb{P}}[\mathbf{y}]$.

We can use this characterization of the models of $T_{kinship}$ to exploit the correspondence between the connected substructure of a kinship graph structure and definable relations found in the ontology. If a connected substructure is identified in the graph, a definition for the model that corresponds to the substructure can be written down in first-order logic using the ontology. As a result, the verification of definitional extensions follows from the verification of the primitive kinship theory, $T_{kinship}$.

[16]Notation: K_n is the complete graph with n vertices. $\overline{K_n}$ is the complement of K_n.

5. Validation of $T_{kinship}$

In order to validate $T_{kinship}$, we can use the ontology to axiomatize the relationships found in the aforementioned kinship systems, along with definitions of the kinship relationships found in StatsCan and Canadian legal documents. In particular, we are now able to axiomatize the algebraic models presented by Read: we have taken this independently-derived work from anthropology about kinship and have formalized these intended models that were previously expressed in natural language and in relational algebra. This in contrast to previous approaches where we have formalized the axioms *and* then identified the intended models of the ontology based on *our* interpretations.

For example, we can axiomatize the relationships from the various kinship systems, from Read's algebraic models, and from definitions provided by Wallace and Atkins:

(EX-1) We can generalize the notion of *first cousin* as the child of a parent's sibling. Using the algebraic model from [11] (also in Figure 3), this notion is also captured in the definition for the binary $hasCousin(x,y)$ relation.

$$(\forall x \forall y\, (hasCousin(x,y) \equiv (\exists k \exists w \exists z\, (hasChild(k,z) \wedge hasChild(k,w) \wedge hasChild(z,x) \wedge hasChild(w,y) \wedge (w \neq z))))).$$

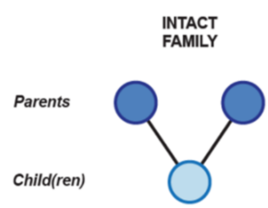
(EX-2) Similarly, we can do the same with concepts like grandchild.

$$(\forall x \forall y\, (hasGrandchild(x,z) \equiv (\exists y \exists z (hasChild(x,y) \wedge hasChild(y,z))))).$$

(EX-3) Additionally, an application of the ontology would be to axiomatize definitions found in StatsCan documentation. For example, StatsCan defines an *intact family* as a family unit where "all children are the biological or adopted children of both married spouses or of both common-law partners [17]." This is also graphically depicted by StatsCan in Figure 8. We can extend $T_{kinship}$ with a new module that contains the $inFamily(x,y)$ relation and the $familygroup(x)$ class to group people together to axiomatize these StatsCan definitions.

$$(\forall x\, (intactfamily(x) \equiv (familygroup(x) \wedge \exists y \exists z\, inFamily(y,x) \wedge inFamily(z,x) \wedge hasSpouse(y,z) \wedge (y \neq z) \wedge (\forall u\, (inFamily(u,x) \wedge (u \neq y) \wedge (u \neq z)) \supset hasChild(y,u))))).$$

(a) Axiom for intact family using $T_{kinship}$.



(b) Figure 3 in [17].

Figure 8. Intact family as defined by StatsCan in [17].

Read proposes a nonassociative algebra $\mathscr{K} = \langle L, \circ, * \rangle$ for kinship relations. The $\circ$ operator represents the composition of consanguineal relations and the $*$ operator represents the composition of spousal (also known as affinal) relations. In [18], Read presents structural equations to construct structural relationships using the American Kinship Terminology (AKT); for example, the sentences in Figure 9 indicate how the algebra can be used to develop the kin term map shown in Figure 3. The $\circ$ operator indicates the composition of kinship terms; for example, 'Self' is the identity term which can be determined using what Read calls a structural equation of 'Parent $\circ$ Child = Self'. A 0 in a structural equation indicates that the terms for 'parent of parent-in-law' (R-4) and 'parent of child-in-law' (R-5) are not valid kin terms in the AKT system.

$$\text{Parent} \circ \text{Child} = \text{Self} \tag{R-1}$$
$$\text{Spouse} \circ \text{Spouse} = \text{Self} \tag{R-2}$$
$$\text{Spouse} \circ \text{Parent} = \text{Parent} \tag{R-3}$$
$$\text{Parent} \circ \text{Parent} \circ \text{Spouse} = 0 \tag{R-4}$$
$$\text{Parent} \circ \text{Spouse} \circ \text{Spouse} = 0 \tag{R-5}$$
$$\text{Spouse} \circ \text{Child} \circ \text{Parent} = \text{Child} \circ \text{Parent} \circ \text{Spouse} \tag{R-6}$$

Figure 9. Example algebraic compositions for relationships in the American Kinship Terminology (AKT) from [9] and [18]. $\circ$ is the composition operator for the structural equation, and 0 indicates that this is not classified as a kin term in AKT.

In order to demonstrate that kinship structures (which are models of $T_{kinship}$) are the *right* class of structures, we show their relationship to Read's nonassociative algebra. The basis for this relationship lies in identifying the graphs that correspond to each class of structures, and then showing how these graphs are related to each other.

The central theorem that shows the relationship between kinship structures and Read's nonassociative algebra $\mathscr{K}$ for kinship relations relies on two classes of graphs that are associated with the respective structures.

Definition 4 *$\mathbb{G} = \langle V, \mathbf{E} \rangle$ is the Hasse graph for a partial ordering $\mathbb{P} = \langle V, \leq \rangle$ if $(\mathbf{x}, \mathbf{y}) \in E$ iff either $\mathbf{x}$ covers $\mathbf{y}$ or $\mathbf{y}$ covers $\mathbf{x}$ in $\mathbb{P}$.*

Definition 5 *Let $\mathbb{M} = \langle V, \circ \rangle$ be a semigroup such that S is a generating set for $\mathbb{M}$.*

A graph $\mathbb{G} = \langle V, \mathbf{E} \rangle$ is the Cayley graph for $\mathbb{M}$ iff $S \subseteq V$ and $(\mathbf{x}, \mathbf{y}) \in E$ iff there exists $\mathbf{z} \in S$ such that $\mathbf{y} = \mathbf{x} \circ \mathbf{z}$.

The idea is that there is a graph homomorphism that maps paths in the Hasse graph of a kinship structure to the kinship relations that are the vertices of the Cayley graph of Read's algebra.

Theorem 2 *Let $\mathbb{K} = \mathbb{P} \oplus \mathbb{G}$ be a kinship structure and let $H(\mathbb{P})$ be the Hasse graph for $\mathbb{P}$.*

$H(\mathbb{P}) \oplus \mathbb{G}$ is homomorphic to the Cayley graph $\Gamma(\mathscr{K})$ for the algebra $\mathscr{K}$.

For example, the path between Bruce and Peter in the kinship structure in Figure 6 is mapped to the following relation in Read's nonassociative algebra:

$$\text{Parent} \circ \text{Spouse} \circ \text{Child} \circ \text{Child} \circ \text{Child}$$

Note that Theorem 2 also means that substructures of kinship structures that are not paths (e.g., the relations depicted in Figure 7) are not mapped to relations in Read's nonassociative algebra.

With the verification and validation of $T_{kinship}$, we have shown that we have represented all the relationships captured in Read's algebraic models in anthropology, the legal context, and statistics collection agencies. Further, the benefit of having a first-order axiomatization of kinship allows us to define relationships that cannot be defined by Read in [9, 11]. The algebraic approach does not include constraints on how spouse and ancestors can be amalgamated. For example, Axiom 13 is a constraint we have included in $T_{kinship}$ based on the current Canadian law for marriage, but such a constraint cannot be represented using Read's approach. While the application of these axioms may depend on the legal context, it is equally important to be able to represent such constraints: algebraically, it is not possible to do so, whereas our first-order axiomatization allows us to further add onto Read's kinship algebra.

6. Lessons Learned & Future Work

We have shown how the kinship ontology can represent definitions developed by anthropologists and our commonsense intuitions of familial relationships. We have extracted these definitions of kinship found in anthropology and have axiomatized the algebraic structures presented within the anthropological community using first-order logic. Further, the ontology is more expressive than the algebraic approach and also supports reasoning. In contrast to existing OWL ontologies for kinship, we have presented an ontology that is not a toy ontology for reasoning in OWL and have been able to validate it with anthropological outside of the ontology community; this is significant since there are additional kinship systems found in anthropology that can be further examined through the use of ontologies.

Future work for this ontology would be to provide a more in-depth ontological analysis of kinship notions independent of anthropology and societal norms. We would like to explore representations of relationships that are weaker than Axiom 13, but stronger than the weakest set of axioms possible with $T_{kinship}$, and how unintended and anomalous models of the ontology interact with one another. As well, we would like to examine how changes in relationships affect the models of the ontology: how do life events, such as marriage and divorce, influence or change the axioms of the ontology? Furthermore, it would be interesting to examine the effects of temporal kinship and how this plays a role with making inferences from data: for example, how have relationships changed over time with census datasets? Using the ontology, can we make additional inferences with datasets from different years to analyze marriage or divorce rates?

References

[1] Stevens R, Matentzoglu N, Sattler U, Stevens M. A Family History Knowledge Base in OWL 2; 2014. OWL Reasoner Evaluation Workshop 2014. Available from: http://ceur-ws.org/Vol-1207/paper_11.pdf.

[2] Vacura M, Svátek V, Gangemi A. An Ontological Investigation Over Human Relations in Linked Data. Applied Ontology. 2016;11(3):227–254. Available from: https://doi.org/10.3233/AO-160169.

[3] Sazonau V, Sattler U, Brown G. General Terminology Induction in OWL. In: Tamma V, Dragoni M, Gonçalves R, Ławrynowicz A, editors. Ontology Engineering. Cham: Springer International Publishing; 2016. p. 1–13.

[4] Longo C, Gangemi A, Cantone D. Representing Kinship Relations on the Semantic Web. In: Proceedings of the 10th International Workshop on OWL: Experiences and Directions (OWLED 2013) co-located with 10th Extended Semantic Web Conference (ESWC 2013), Montpellier, France, May 26-27, 2013; 2013. Available from: http://ceur-ws.org/Vol-1080/owled2013_15.pdf.
[5] Morgan LH. Systems of Consanguinity and Affinity of the Human Family. Smithsonian Institution,; 1871.
[6] Murdock GP. Social Structure. Macmillan New York; 1949.
[7] Read DW. A New Approach to Forming a Typology of Kinship Terminology Systems: From Morgan and Murdock to the Present. Structure and Dynamics. 2013;6. Available from: https://escholarship.org/uc/item/0ss6j8sh.
[8] Wallace AFC, Atkins J. The Meaning of Kinship Terms. American Anthropologist. 1960;62(1):58–80. Available from: http://www.jstor.org/stable/666908.
[9] Read DW, Atkins J, Buchler IR, Fischer M, Meur GD, Lally E, et al. An Algebraic Account of the American Kinship Terminology [and Comments and Reply]. Current Anthropology. 1984;25(4):417–449. Available from: http://www.jstor.org/stable/2742902.
[10] Read DW. Kinship Algebra Expert System (KAES): A Software Implementation of a Cultural Theory. Social Science Computer Review. 2006;24(1):43–67. Available from: https://doi.org/10.1177/0894439305282372.
[11] Read D. Kinship, Formal Models of. In: Wright JD, editor. International Encyclopedia of the Social & Behavioral Sciences (Second Edition). second edition ed. Oxford: Elsevier; 2015. p. 53 – 60. Available from: http://www.sciencedirect.com/science/article/pii/B9780080970868431144.
[12] Guindi FE, Read DW, Boyd JP, Burton M, Crumrine NR, Davis R, et al. Mathematics in Structural Theory [and Comments and Reply]. Current Anthropology. 1979;20(4):761–790. Available from: http://www.jstor.org/stable/2741685.
[13] Read D. 13. In: Mathematical Representation of Cultural Constructs. John Wiley & Sons, Ltd; 2011. p. 227–253. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781444394931.ch13.
[14] Guarino N. The Role of Identity Conditions in Ontology Design. In: Freksa C, Mark DM, editors. Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science, International Conference COSIT '99, Stade, Germany, August 25-29, 1999, Proceedings. vol. 1661 of Lecture Notes in Computer Science. Springer; 1999. p. 221–234. Available from: https://doi.org/10.1007/3-540-48384-5_15.
[15] Masolo C, Vieu L, Bottazzi E, Catenacci C, Ferrario R, Gangemi A, et al. Social Roles and their Descriptions. In: Dubois D, Welty CA, Williams M, editors. Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004), Whistler, Canada, June 2-5, 2004. AAAI Press; 2004. p. 267–277. Available from: http://www.aaai.org/Library/KR/2004/kr04-029.php.
[16] Enderton HB. A Mathematical Introduction to Logic. Academic Press; 1972.
[17] Minister of Innovation, Science, and Economic Development. 2016 Census of Population: Families, households and marital status; 2017. Accessed 2020-01-17. Available from: https://www12.statcan.gc.ca/census-recensement/2016/ref/98-501/98-501-x2016004-eng.cfm.
[18] Read DW. Formal analysis of kinship terminologies and its relationship to what constitutes kinship. Anthropological Theory. 2001;1(2):239–267. Available from: https://doi.org/10.1177/146349960100100205.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200664

Ontological Analysis and Modularization of CIDOC-CRM

Emilio M. SANFILIPPO [a,b,1], Béatrice MARKHOFF [c], and Perrine PITTET [b]
[a] *Le Studium Loire Valley Institute for Advanced Studies, Orléans and Tours, France*
[b] *CESR – UMR 7323, University of Tours, France*
[c] *LIFAT, University of Tours, France*

Abstract. The CIDOC-CRM ontology is a standard for cultural heritage data modeling. Despite its large exploitation, the ontology is primarily maintained in a semi-formal notation, which makes it difficult to homogeneously exploit it in digital environments. In addition, the ontology consists of several classes and relations, whereas one sometimes wishes to reuse it but only partially. The purpose of the paper is to contribute to the use of CIDOC by strengthening its foundations. On the basis of formal ontology theories, we propose a first analysis of the ontology to enhance its conceptual structure. We also present a preliminary modularization of CIDOC aimed at enhancing both its formalization and usage.

Keywords. CIDOC-CRM, cultural heritage data modeling, modularity

1. Introduction

The CIDOC Conceptual Reference Model (hereafter CIDOC) is a standard ontology (ISO 21127) for cultural heritage data modeling [1]. CIDOC has been adopted in several research projects and it constitutes the conceptual architecture for archives, libraries, and museums, among other institutions, to organize data in information systems [2].

Despite its large exploitation, CIDOC is only weakly axiomatized and some of its modeling choices remain opaque. Existing works like [3] have improved its formal treatment but they have only partially contributed to improve its conceptual framework. For instance, as we will see in the next sections, the ontology adopts a representational approach at the intersection between three- (3D) and four-dimensionalism (4D), which – apart from being controversial from a theoretical standpoint [4] – does not seem to bring any advantage from a modeling perspective. In addition, by working with end-users in the exploitation of the ontology, we have observed that the intended meaning of some of its elements is open to alternative interpretations (e.g., the class *E5 Event*),[2] which is a fact running the risk of compromising its uniform usage across applications.

The purpose of the paper is to contribute to the exploitation of CIDOC by strengthening its ontological and formal foundations. We attempt in this way at making the on-

[1]Corresponding Author: CESR - Université de Tours, 59, rue Néricault-Destouches, 37020 Tours, France. Email:emiliosanfilippo@gmail.com (permanent address).
[2]Each class in CIDOC is prefixed by a unique ID starting with 'E', whereas relations' IDs start with 'P'.

tology more robust and transparent to its users. In order to achieve this goal, we present a first ontological analysis of (some parts of) CIDOC based on well-known approaches in applied ontology. In particular, we rely on both the OntoClean methodology [5] to analyze the taxonomic relations of CIDOC and theories of formal ontology (e.g., 3D, 4D, etc.) to improve its overall conceptual framework. Since many of the latter theories have been already adopted in foundational ontologies like UFO [6] and DOLCE [7], among others, we will rely on these ontologies, too, to analyze CIDOC.

The paper is structured as follows. We present and analyze in Sect. 2–Sect.5 some of the core modeling elements of CIDOC. On the basis of the analysis, we propose in Sect. 6 a modularization of the ontology which revises an existing formalization. By splitting CIDOC in various (inter-connected) modules, we attempt to allow for its *selective* reuse depending on specific application scenarios. Sect. 7 concludes the paper by addressing future work needed to strengthen our proposal.

2. Overview of CIDOC-CRM

The CIDOC ontology (version 6.2.1)[3] [1] consists of 94 taxonomically organized classes and 168 horizontal relations (called *properties*). It is mainly conceived and maintained in a semi-formal and application-independent notation, although the ontology is nowadays largely exploited in Semantic Web environments through languages like RDF and OWL (see, e.g., [8,9]). For each class, the original specification provides 1) its parent and child classes (if the latter are present), where only *direct* taxonomic relations are specified in first-order logic (FOL); 2) a natural language definition, which is associated to comments and examples to facilitate the understanding of the class; 3) in some cases, the horizontal relations by which the class can be linked to other classes. Similarly, for each relation the specification provides 1) domain and range information (in both natural language and FOL); 2) taxonomic relations (with respect to other relations); 3) natural language comments and examples; 4) cardinality restrictions (called *quantification*). According to CIDOC, the latter "are provided for the purpose of semantic clarification only, and should not be treated as implementation recommendations" [1, p.XIII]. Hence, given a relation associated with a cardinality, it is not mandatory to comply with the latter when the ontology is represented in a specific formal notation.[4]

For the sake of clarity, consider the following example. The class *E5 Event* is subsumed by *E2 Temporal Entity*. Among others, the relation *P11 had participant* is used to relate *E5 Event* to *E39 Actor*. The cardinality of P11 is set to (0,n) on both sides. CIDOC is however liberal to alternative interpretations. This choice is unfortunate since divergent formalizations may lead to scarcely interoperable data models. For instance, consider two alternative formalizations; the first one, call it O_1, implements cardinalities as they are given in [1]; the second one, O_2, where the cardinality of P11 is restricted to (1,n) on the side of *E39 Actor* so that an instance of *E5* must have at least one actor as participant. While O_2's models are O_1's models, too, the vice-versa does not hold. In this sense, by leaving open to users the choice of how to interpret cardinalities, the CIDOC's approach runs the risk of making it hard for applications to interoperate.

[3]CIDOC version 6.2.1 is the most recent stable version of the ontology; see http://www.cidoc-crm.org/versions-of-the-cidoc-crm, last accessed March 2020.

[4]In the work presented in [3], cardinalities are interpreted as suggested in [1].

Figure 1 shows the most general classes of CIDOC.[5] We discuss the representation of persistent items and spacetime volumes in Sect.3, temporal entities and time spans in Sect. 4, dimensions in Sect. 5. The analysis of places is left to future work.

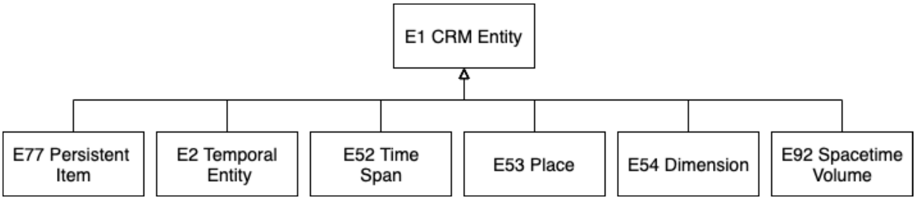


Figure 1. Upper-level taxonomy of CIDOC (v.6.2.1)

Before moving to the next sections, note that the distinction between *E77 Persistent Item* and *E2 Temporal Entity* is the core dichotomy of CIDOC. Instances of the former are *endurants* keeping their identity through time [1, p.35], whereas instances of the latter are *perdurants* unfolding in time [1, p.2]. These classes are therefore disjoint.[6] Also, CIDOC adopts a so-called *event-oriented* approach (in the terminology of [2]), according to which the representation of events is fundamental in the scope of the ontology. For example, representing a person's birth date means, first, to represent the person's birth event and, second, to label the time span of this event by a date.

3. Analysis of Persistent Items

We analyze in this section the taxonomy of persistent items, see Fig. 2. We first provide a general overview on the taxonomy by introducing some of its classes and we then analyze the taxonomy while introducing the remaining classes.

Looking at Fig. 2, CIDOC models a high-level distinction between *E39 Actor* and *E70 Thing*. Instances of *E39 Actor* are either individual persons (*E21 Person*) or groups (*E74 Group*) "who have the potential to perform intentional actions" [1, p.20]. The class *E40 Legal Body* extends *E74 Group* to model "institutions or groups of people that have obtained a legal recognition [...] and can act collectively as agents" [1, p.21].

E70 Thing is a generic class subsuming different types of entities. A first distinction is between man-made (*E71 Man-Made Thing*) and non-man-made things (*E19 Physical Object*, *E26 Physical Feature*); as the terminology suggests, only the former are intentionally produced by actors. A second distinction is between *E18 Physical Thing* and *E28 Conceptual Object*. Instances of the former class exist in space, whereas instances of the latter are "non-material products of our minds" [1, p.16] such as natural languages (*E56 Language*), the 'contents' of physical books (*E89 Propositional Object*), or types (*E55 Type*, e.g., material types), among others. According to CIDOC, conceptual objects "exist as long as they can be found on at least one [physical] carrier or in at least one human memory" (ibid.). Since *E28 Conceptual Object* is not subsumed by *E18 Physical Thing*, it follows that its instances do not reside in space.[7]

[5] CIDOC includes also *E59 Primitive Value* at the same level of *E1 CRM Entity* to represent data types. We comment on E59 in Sect. 6.

[6] Apart from the disjointness between E77 and E2, there is only another disjointness declaration in CIDOC between *E18 Physical Thing* and *E28 Conceptual Object*, see Sect.3.

[7] The analysis of conceptual objects is left to future work.

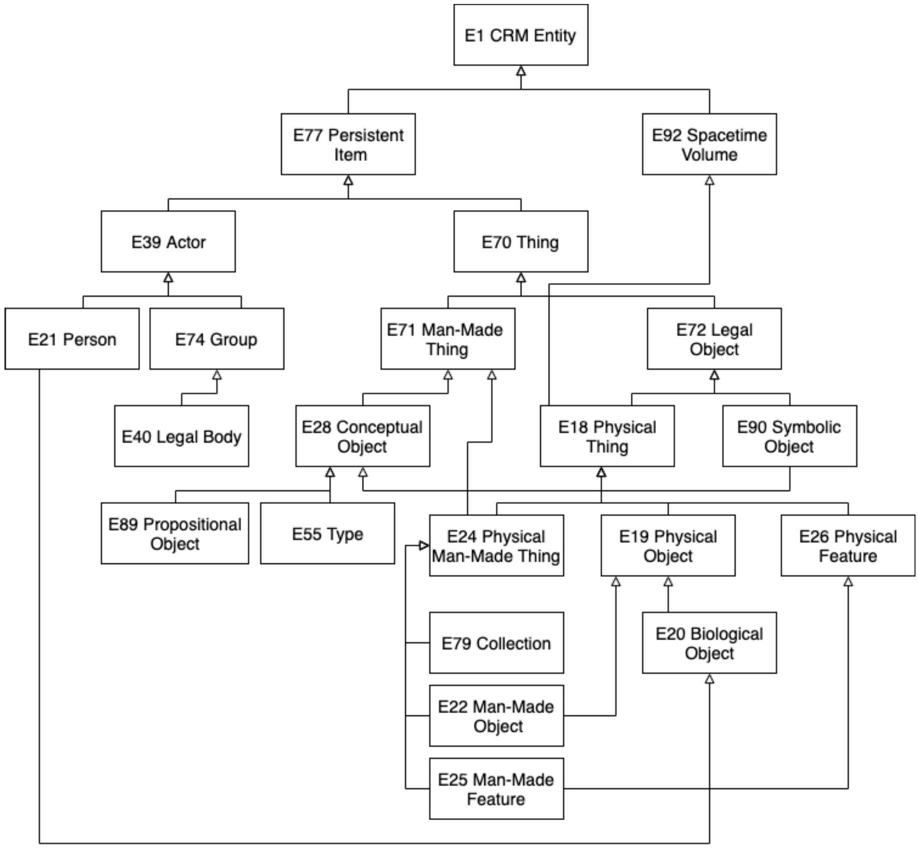


Figure 2. Partial taxonomy of persistent items in CIDOC (v.6.2.1)

To comment on the taxonomy, first, the distinction between *E39 Actor* and *E70 Thing* is not so sharp. Looking at Fig. 2, *E21 Person* is subsumed by *E20 Biological Object*, which is subsumed by E70. In addition, the scope of E70 is broad enough to cover E39 and all its subclasses.

Second, *E72 Legal Object* subsumes all physical things, amongst other classes. Its instances are material or immaterial items to which legal rights, such as property rights, apply. In our understanding, from a formal ontology perspective, *E72 Legal Object* models *anti-rigid* properties – in the sense of OntoClean [5], i.e., properties that entities only possibly satisfy and whose acquisition or loss does not alter their identities. For instance, a human being is subject to legal rights and duties in the scope of a specific socio-legal system, independently from which she always remains a human being for the entire duration of her life. On the other hand, it is reasonable to assume that *E18 Physical Thing* models *rigid* properties, i.e., properties that entities necessarily satisfy and whose loss *does* affect identity. Assuming these considerations along with the formal treatment of anti-/rigidity in OntoClean, physical things can not be subsumed by legal objects.

Finally, the class *E92 Spacetime Volume* deserves some discussion. CIDOC has inherited this class from the CRMgeo [10], which extends CIDOC for geo-spatial applications. According to [1], E92 “comprises 4 dimensional point sets (volumes) in physical spacetime [...]. An instance of *E92 Spacetime Volume* is either contiguous or composed of a finite number of contiguous subsets " [1, p.41]. Apart from *E4 Period* (see Sect. 4) and *E18 Physical Thing*, this class subsumes *E93 Presence*, i.e., "*snapshots* of a Spacetime volume, i.e. intersections of a Spacetime volume with all space restricted to a particular time-span, such as the extent of the Roman Empire during 33 B.C. " [10].

If we interpret it properly, instances of E92 correspond to *four-dimensional worms* in the sense of ontological four-dimensionalism (4D) [11]. This seems clear from its definition as something that has both temporal and spatial extents but also from the examples in [1,10]; e.g., the fact that an individual spacetime volume can be cut in different parts, each one standing for a spatio-temporal 'snap-shot' of the entity at stake like the Roman Empire during 33 B.C. If this consideration is correct, CIDOC mixes 4D with a standard three-dimensionalism (3D) view.[8] From a foundational perspective, this approach is controversial. Despite the hot debate on 4D and 3D in formal ontology, these remain indeed alternative and perhaps even incompatible positions (see [4] for some discussion). The situation is not better from a modeling perspective, since the benefits of introducing spacetime volumes is unclear. According to [1], a reason for having these entities is to simplify data models; e.g., to represent "an [instance of] *E18 Physical Thing* without representing each instance of it together with an instance of its associated spacetime volume" [1, p.12]. What the specification seems to suggest is that one can represent physical (or temporal) entities without necessarily modeling their spatial or temporal locations. This because they inherit their spatio-temporal dimension by being instances of E92. In our view, this consideration is not fully correct. First, it can be relevant for application purposes to explicitly model, e.g., the space region occupied by an individual object at a certain time. Second, even by assuming the distinction between space regions, temporal regions, perdurants, and endurants, it is not necessary – at the instance level – to represent all (spatial, temporal) regions which an object occupies during its entire life or all perdurants where it participates.

On the basis of this analysis, Fig. 3 shows the restructuring of the taxonomy of persistent items. Classes with dashed lines are new;[9] also, the taxonomy does not include *E70 Thing*, *E72 Legal Object*, and *E92 Spacetime Volume*. Some comments are due.

First, *E18 Physical Thing* is now directly subsumed by *E77 Persistent Item* and it is disjoint with *Non-Physical Thing*. This latter class is introduced to sharply distinguish between physical and non-physical items. *Non-Physical Man-Made Thing* extends *Non-Physical Thing* to explicitly classify non-physical items resulting from human actions.[10] *E70 Thing* has been removed because it was only a generic umbrella without any specific intended meaning. The class *E71 Man-Made Thing* is directly subsumed by *E77 Persistent Item*. It is neither disjoint nor subsumed by E18 or *Non-Physical Thing*, because it subsumes both physical and non-physical man-made entities.

Second, looking at physical things, we introduce *Aggregation* to distinguish between general collections of physical things (e.g., all objects on my desk) and instances of *E78 Collection*, among others. Aggregations should not be confused with physical objects having multiple and physically connected parts such as potteries or statues (both

[8]Recall that *E2 Temporal Entity* and *E77 Persistent Item* are disjoint classes.

[9]Following CIDOC's minimality principle (see [1, p.XVI]) each new inserted class is used either as domain or range for a relation.

[10]The disjointness between *Non-Physical Man-Made Thing* and *E24 Physical Man-Made Thing* can be logically derived. It is included in the diagram to facilitate understanding.

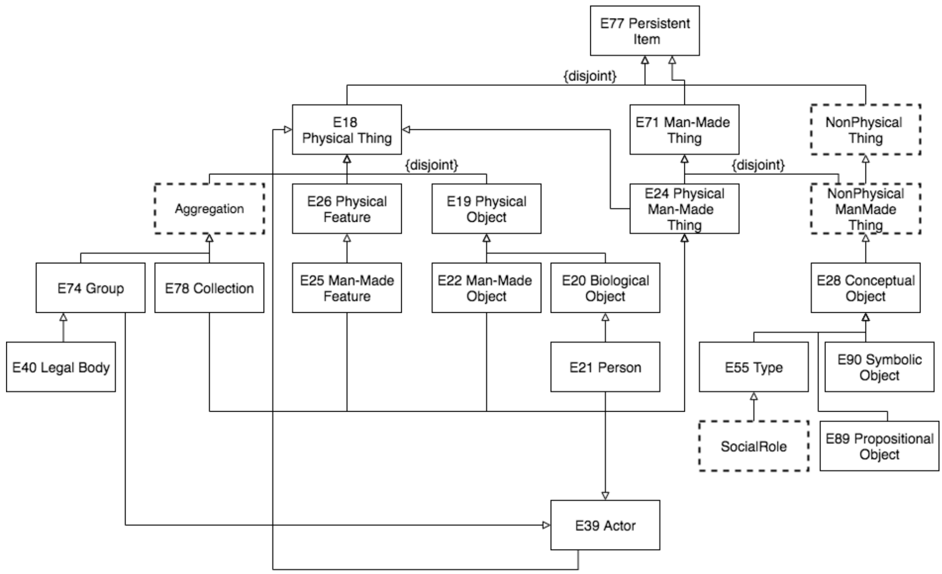


Figure 3. Revised taxonomy of persistent items

instances of *E22 Man-Made Object*). Aggregations bear indeed unity conditions other than topological ones. For instance, according to [1], museum collections, which are represented as specific types of aggregations in Fig. 3, are "assembled and maintained by one or more instances of E39 Actor over time for a specific purpose and audience" [1, p.36]. An example is the collection of the British Museum, which qualifies as a collection because it consists of objects collected and owned by the museum, and possibly used during its exhibitions. Its unity could be therefore defined in legal terms. *E74 Group* and *E40 Legal Body* are both subsumed by *Aggregation*, following CIDOC's understanding of groups as collection of individual persons satisfying (non-topological) unity conditions.[11] In addition, both *E74 Group* and *E21 Person* are subsumed by *E39 Actor*, which is a direct subclass of *E18 Physical Thing*. The revision of CIDOC concerning agents is based on and simplifies the ontology of groups and institutions presented in [12,13]. In these works, the authors distinguish between arbitrary collections of individuals and social groups. In addition, differently from CIDOC, the approach in [12,13] allows to explicitly represent the membership conditions that individuals must satisfy to form groups. This approach could be adopted to enhance the ontology of actors in CIDOC, which remains only weakly characterized at the current state.

Third, *E92 Spacetime Volume* has been removed from the taxonomy because of its ambiguity. However, since CIDOC covers both places, temporal regions, and temporal entities, even by removing E92, one still has the possibility of linking persistent items to space, time, and temporal entities.

Finally, by conceiving legal objects as social roles, instances of *E72 Legal Object* can be represented in different ways. A proposal, based on [14], consists in introducing a

[11] Since CIDOC understands legal bodies as groups with legal status, legal bodies constituted by single persons are not covered by the ontology. An extension in this direction could be needed.

new class, *Social Role*, for properties like *being a student* or *being a professor* that entities satisfy within specific contexts. From this perspective, legal objects can be (roughly) understood as roles that entities acquire in socio-legal systems or events. Following [14], the property of *being a legal object* is reified in the domain of discourse as an instance of *Social Role*, whereas the CIDOC's relation *P2 has type* can be used to link an entity to it (e.g., a statue *has type* legal object); alternatively, a new relation can be easily introduced.[12]

4. Analysis of Temporal Entities

Figure 4 shows the highest classes in CIDOC for the representation of temporal entities. For the sake of the analysis, we limit to show the taxonomic relations between these classes while providing a general overview on their subclasses to facilitate the understanding of the modularization of the ontology presented in Sect. 6.

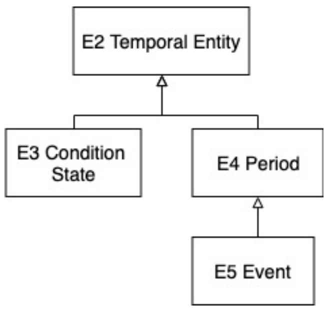


Figure 4. Top-level temporal entities in CIDOC (v.6.2.1)

The class *E3 Condition State* "comprises the states of objects characterized by a certain condition over a time-span" [1, p.3]. An example provided in [1] is the "condition of the SS Great Britain between 22 September 1846 and 27 August 1847 [as being] *wrecked*" (ibid). From a formal ontology perspective, this class matches well with the notion of *state*, e.g., in the DOLCE ontology [7] (e.g., *being sitting*, *being open*, etc.).

E4 Period subsumes all temporal entities other than condition states. It is defined as comprising "sets of coherent phenomena or cultural manifestations occurring in time and space. It is the social or physical coherence of these phenomena that identify a *E4 Period* and not the associated spatiotemporal extent. [...] Often, this class is used to describe prehistoric or historic periods such as the *Neolithic Period*, the *Ming Dynasty* or the *McCarthy Era* [...]" [1, p.3]. E4 subsumes *E5 Event*, whose instances are "changes of states in cultural, social or physical systems, regardless of scale, brought about by a series or group of coherent physical, cultural, technological [...] phenomena" [1, p.5]. E5 directly subsumes *E7 Activity*, i.e., intentional actions performed by actors; *E63 Beginning of Existence*, i.e., events that bring into existence persistent items; and *E64 End*

[12] We model legal object as an individual rather than a class to avoid multiplying roles for specific entities, e.g., the legal-object-role$_1$ of statue$_1$ *vs* the legal-object-role$_2$ of statue$_2$. The reader can refer to [15] for various approaches on the modeling of roles.

of Existence, i.e., events that end the existence of persistent items. These classes are not mutually disjoint (e.g., *E12 Production* is subsumed by both E7 and E63).

Classes like *E66 Formation*, *E66 Dissolution*, *E86 Leaving*, *E85 Joining*, *E67 Birth*, and *E69 Death* are related to actors, in particular, to the formation and dissolution of groups, to persons leaving and joining groups, and to persons' birth and death, respectively. *E11 Modification* and *E65 Creation* are related to the production of physical man-made things and conceptual objects, respectively. *E6 Destruction* models intentional or natural events that destroy physical things. Instances of *E81 Transformation* are events resulting in the destruction of a persistent item and the creation of another item which is different in both nature and identity in comparison to the destroyed one. *E13 Attribute Assignment* concerns the attribution of properties to entities; among its subclasses, it covers measurement events. Finally, *E9 Move*, *E10 Transfer of Custody*, *E8 Acquisition*, and *E87 Curation Activity* are specific to the cultural heritage domain; e.g., they can be useful to describe the transfer of ownership of goods from one museum to others.

Let us now comment, in particular, on the notions of *E4 Period* and *E5 Event*. A first issue is that E4 captures temporal phenomena bearing a cultural nature (e.g., Italian Renaissance, Cubism, etc.). Instances of E5, however, are not necessarily relevant from a cultural standpoint according to CIDOC (see, e.g., the class *E6 Destruction* in [1]). The subsumption of E5 under E4 is therefore misguided. A second issue concerns the mereological structure of periods and events. At first glance, instances of E4 are complex temporal entities consisting of multiple (temporal) parts. At the same time, CIDOC does not take any explicit commitment on the structure of events, which can be either complex or atomic (see [1, p.3]). This is unfortunate because if periods are complex, considering the subsumption of E5 under E4, it cannot be the case for events to be atomic.[13]

On the basis of these considerations, we propose to detach the classes E4 and E5, and to subsume the latter directly under *E2 Temporal Entity*. In this perspective, E5 is a general umbrella for temporal entities that are neither condition states nor periods. A mereological relation of *parthood* between temporal entities can be used to model atomic and complex temporal phenomena (see, e.g., [7]). Finally, E4, E5, and E3 are disjoint.

5. Analysis of Dimensions

The class *E54 Dimension* is directly subsumed by *E1 CRM Entity* (see Fig. 1) to capture "quantifiable properties that can be measured by some calibrated means and can be approximated by values, i.e. points or regions in a mathematical or conceptual space, such as natural or real numbers, RGB values etc." [1, p.26]. The relationship *P43 has dimension* links things to dimensions; *P90 has value* relates dimensions to numeric values, whereas *P91 has unit* models the link between a dimension and its measurement unit, the latter being represented via *E58 Measurement Unit*, a subclass of *E55 Type*.

From a formal ontology perspective, CIDOC's dimensions correspond to a restricted understanding of *qualities* in foundational ontologies like DOLCE or UFO, 'restricted' because limited – at first glance – to classes of qualities for sizes, e.g., lengths or widths.

[13] It should be noted that the distinction between events and periods is partially a question of scale of observation: "Viewed at a coarse level of detail, an *E5 Event* is an instantaneous change of state. At a fine level, the *E5 Event* can be analysed into its component phenomena within a space and time frame, and as such can be seen as a *E4 Period* [1, p.4] (emphasis is ours). CIDOC however lacks a framework to handle granularity.

Also, similarly to these ontologies, CIDOC assumes that a dimension characterizes a single entity. In addition, a dimension can have exactly one value. It is not however clear whether changes in dimensions' values affect changes in their identities.

A drawback in the CIDOC's conceptualization of dimensions is the restriction of their values to numerical terms only, whereas one may wish to represent also *qualitative* values.[14] For instance, representing a man-made object's color, one may wish to say that it is red without specifying its exact shade in quantitative terms. Our proposal is to revise CIDOC on the basis of the work done in [7,16], therefore, by allowing for the representation of dimensions' qualitative values, too. This is done by introducing the class *Qualitative Quality Space*, which provides a way to organize and represent qualities' values in terms of, e.g., mereological or topological structures, among others (see the `cidoc:dimension-module` described in Sect. 6).

6. Towards the Modularization of CIDOC

We discuss in this section a preliminary modularization of CIDOC; we do not cover the entire input ontology and future work in this regard is required. By the end of the section, we present examples about cultural heritage data modeling showing the (potential) advantages of using CIDOC in different inter-connected modules.

Before presenting the modular structure, let us recall some core ideas about ontology modularization. Following [17] "ontology modularization can be interpreted as decomposing potentially large and monolithic ontologies into (a set of) smaller and interlinked components (modules)." An ontology module *M* corresponds to "[...] a subset of a source ontology O, $M \subset O$, either by abstraction, removal or decomposition, or module M is an ontology existing in a set of modules such that, when combined, make up a larger ontology" [18]. Also, despite the amount of research work, at the current state of the art "*there is no universal way to modularize an ontology*" [19] (emphasis is ours). Hence, according to the same authors, "the choice of a particular technique or approach should be guided by the requirements of the application or scenario relying on modularization" (see [18] for similar considerations in a more recent publication).

For our application and research purposes the modularization of CIDOC is primarily aimed at facilitating its *selective* use. For example, when modeling (social) groups, one may be interested in their members without necessarily describing the events by which the groups are created (or destroyed). Similarly, when working with man-made objects, one may wish to represent only their physical structure without necessarily relating them to temporal information. Because of usability requirements, we rely on Semantic Web (SW) languages, namely, the Web Ontology Language (OWL). Recall that OWL is indeed the leading formalism for the exploitation of ontologies in the Digital Humanities (see, e.g., [20]). In addition, by using OWL, we aim at enhancing the (computational) representation of the ontology. For this purpose, we reuse and (partially) revise the Erlangen release of CIDOC,[15] which formalizes the latter (version 6.2.1) in OWL.

In addition to usability criteria, the modularization of the ontology has been driven by *functional* and *subject similarity* considerations between its various modeling elements. Accordingly, we group classes (and relations) which are aimed at a common goal

[14] This is a further restriction of CIDOC in comparison to DOLCE or UFO.

[15] https://github.com/erlangen-crm/ecrm, last accessed in March 2020.

(e.g., facilitating the integration of other modules) or at covering the same portion of reality. For example, considering persistent items (see Fig. 3), one can distinguish between physical things that are not man-made (*Aggregation*, *E19 Physical Object*, and *E26 Physical Feature*) from their man-made counterparts. On the same lines, looking at temporal entities, one can identify and distinguish between, e.g., events concerning the creation or destruction of man-made things (e.g., *E11 Modification* and *E6 Destruction*, among others), and similar events about actors (e.g., *E67 Birth*, *E69 Death*, etc.).

Moving to the technique for the modularization, Kahn and Keet [18] present various automatic approaches based on computational techniques. We have adopted a *manual* approach (an option discussed in [18] as well), because, as a result of the analysis presented in the previous sections, we modularize but also revise CIDOC. We therefore need to look at its conceptual and formal structure and change it wherever necessary.

At the current development stage, the modular architecture comprises 18 modules including the module called `cidoc:whole` which is the union of all modules used to build the whole ontology.[16] For data organization in, e.g., RDF triplestores, this module should be always imported for first to guarantee the integration and interoperability of data instantiating the other modules. For the sake of shortness, we provide here only a general overview of the modules; Tables 1 – 4 give a schematic view on the entire library, including the structure of imports (`owl:imports`).

Besides *E92 Spacetime Volume*, which has been removed, all classes in Fig. 1 constitute the `cidoc:top-module`. This also includes the new class *Qualitative Quality Space* (see below) to represent non-numerical dimensions' values (e.g., the space of weights having values such as *heavy*, *medium*, *light*, etc.). The purpose of the `cidoc:top-module` is to represent the highest classes of the ontology to allow for the consistent integration of all other modules; e.g., to guarantee the disjointness between persistent items and time-spans when these are integrated.

Table 1. General modules

Module name	**Goal**	**Direct imports** (*owl:imports*)
`cidoc:top-module`	To represent the highest classes of the ontology to allow for the consistent integration of all other modules	–
`cidoc:whole module`	The union of all modules in the CIDOC's library	`cidoc:top-module;` `cidoc:persistent-item-whole-module;` `cidoc:temporal-entity-whole-module`

We spend some words on the `cidoc:dimension-module` to explain its differences with the standard CIDOC. First, the module covers the classes *E54 Dimension*, *Qualitative Quality Space*, and *E77 Persistent Item*; the latter is used to characterize dimensions in relation to E77's instances. For instance, one may characterize a pottery as bearing a color-dimension with value *black*, the latter being a region within a space for colors. Note that the intended meaning of *Qualitative Quality Space* is more restricted than the notion

[16]The library of CIDOC's modules is available at: https://github.com/emiliosanfilippo/cidoc-modularization. The repository also contains some diagrams to facilitate the understanding of the modular architecture.

Table 2. Modules about places and dimensions

Module name	**Goal**	**Direct imports** (*owl:imports*)
`cidoc:place-module` `module`	To represent places (*E53 Place*)	–
`cidoc:dimension-module` `module`	To represent dimensions (e.g., *E54 Dimension*, *Qualitative Quality Space*)	–

of *quality space* in [7], where the authors use such spaces for both qualitative and quantitative values. In our case, the latter are simply represented through OWL *data properties* and their *value spaces* (e.g., integers) to express numerical values. This approach weakens the expressivity of the ontology in comparison to [7] (e.g., we can not say that 8kg is a value within a space for weights), but it takes the benefits of a Description Logic based formalism to model quantitative dimensions' values. In addition, end-users can introduce data properties like *hasWeightInKg* to characterize the intended meaning of numerical values attached to dimensions (see [16]). With this approach, differently from the original spirit of CIDOC, dimensions can be now characterized in terms of either quantitative or qualitative values.

The taxonomy of persistent items (see Fig. 3) is split into 6 modules, see Table 3. Since the taxonomy covers both physical and non-physical entities, man-made and non-made-made entities, the `cidoc:persistent-item-top-module` is created to provide the most general classes and, therefore, to facilitate the consistent integration of more specific modules. Also, this module is (indirectly) imported by all modules about persistent items besides the `cidoc:concept-module`.[17]

Table 3. Modules about persistent items

Module name	**Goal**	**Direct imports** (*owl:imports*)
`cidoc:persistent-item-top` `module`	To integrate modules about persistent items	–
`cidoc:physical-thing-module`	To represent non-man-made physical things (e.g., *E19 Physical Object*)	`cidoc:persistent-item-top-module`; `cidoc:place-module`
`cidoc:artifact-module`	To represent physical man-made entities (e.g., *E22 Man-Made Object*)	`cidoc:physical-thing-module`
`cidoc:actor-module`	To represent actors (e.g., *E21 Person*, *E74 Group*)	`cidoc:physical-thing-module`
`cidoc:concept-module`	To represent non-physical conceptual entities (e.g., *E28 Conceptual Object*)	–
`cidoc:persistent-item-whole-module`	The union of all persistent items modules	All modules about persistent items

The modular architecture of temporal entities is organized in 8 modules, see Table 4. The `cidoc:temporal-entity-top-module` covers the most general classes for tem-

[17]The design of the `cidoc:concept-module` is incomplete because further work on the analysis of conceptual entities is required.

poral entities plus the direct subclasses of *E5 Event*, i.e., *E7 Activity*, *E63 Beginning of Existence*, and *E64 End of Existence*, as well as *E52 Time Span*. This module is imported by all modules about temporal entities to guarantee their consistent integration. Looking at the table, note that modules about temporal entities import modules about persistent items. Following [21], an alternative approach would consist in splitting between persistent items and temporal entities, and creating *bridging modules* for their integration. We avoid this approach, first, to keep a simple modular architecture and to avoid the proliferation of modules, second because the representation of temporal entities in cultural heritage scenarios often requires the representation of their participants (see, e.g., [10]).

Table 4. Modules about temporal entities

Module name	**Goal**	**Direct imports** (*owl:imports*)
`cidoc:temporal-entity-top module`	To integrate modules about temporal entities	`cidoc:persistent-item-top-module; cidoc:place-module`
`cidoc:actor-activity-module`	To represent activities related to the life of individual actors or groups (e.g., *E67 Birth*, *E68 Dissolution*)	`cidoc:temporal-entity-top-module; cidoc:actor-module`
`cidoc:attribute-assignment-activity-module`	To represent activities for attributes assignment (e.g., *E16 Measurement*)	`cidoc:temporal-entity-top-module`
`cidoc:creation-activity-module`	To represent the creation of conceptual objects (e.g., *E65 Creation*)	`cidoc:temporal-entity-top-module; cidoc:concept-module`
`cidoc:cultural-heritage-activity-module`	To represent temporal entities relative to cultural heritage (e.g., *E87 Curation Activity*	`cidoc:temporal-entity-top-module; cidoc:actor-module`
`cidoc:modification-activity-module`	To represent the production, modification or destruction of physical entities (e.g., *E79 Part Addition*, *E6 Destruction*)	`cidoc:temporal-entity-top-module`
`cidoc:move-activity- module`	To represent movements of physical objects (*E9 Move*)	`cidoc:temporal-entity-top-module`
`cidoc:temporal-entity-whole module`	The union of all temporal entities modules	All modules about temporal entities

Let us now add some comments. First, CIDOC employs relations which contain disjunctive terms. An example is *P53 has former or current location* between *E18 Physical Thing* and *P53 Place*. This subsumes the relation *P55 has current location* whereas no counterpart for *has former location* is available. From a semantic perspective, the meaning of *having former location* is not the same as *having current location*. It is therefore unclear why a unique modeling element is used, since a relation like P53 can easily lead to misunderstandings. In the ontology modules, we have not reused CIDOC's relations employing disjunctions; rather, we have split each of these relations in further relations while maximizing the reuse of existing elements (e.g., we reuse P55 but not P53).

Second, as said in Sect. 2, CIDOC relies on temporal entities to represent information about persistent items such as birth dates. A similar position is adopted in ontologies like DOLCE or UFO. From a data modeling perspective, however, this approach forces users to create entities which may not be required. Our proposal is to introduce *shortcuts* to enhance data modeling tasks, a strategy which is adopted by CIDOC itself [1]. For example, a new binary predicate *createdAt(o,d)* between a physical man-made object and its production date can be defined (in FOL) as in (Def1), where all defining predicates belong to the CIDOC's signature.[18]

Def1 $createdAt(o,d) \equiv PhysicalManMadeThing(o) \wedge Date(d) \wedge \exists e,t(Production(e) \wedge hasProduced(e,o) \wedge hasTimeSpan(e,t) \wedge identifiedBy(t,d))$

Because of expressivity restrictions, definitions similar to (Def1) can not be employed in SW ontologies. One can however use OWL data properties – possibly by importing them from existing SW vocabularies – while characterizing their formal interpretations in external FOL theories.[19] Following this consideration, we have included in the modules some data and object properties to facilitate data representation.

A third observation is about CIDOC's use of *appellations* (e.g., names, dates) and *primitive values* (strings, numbers). As a formalism-independent model, the relevance of these elements can not be dismissed. When choosing a specific formalism, however, they need to be handled with care (see, e.g., [3]). In the case of OWL, it is reasonable to rely on data types and data properties to handle primitive values and appellations, respectively, rather than representing them as domain instances as it is done in existing OWL releases of CIDOC such as the Erlangen release (see above for references). In this way, one can rely on value spaces to characterize values' meanings and can enable the use of algorithmic procedures to manipulate data (e.g., the use of regular expressions on strings or arithmetic operations on numbers). A deeper analysis of appellations is however required to strengthen their representation.

Finally, the use of cardinality restrictions and axioms in CIDOC deserve attention. For example, physical things are characterized by material types in both [1] (therefore in [3]) and the Erlangen formalization; see (Ax1) for a representation in FOL.[20]

Ax1 $PhysicalThing(x) \rightarrow \exists y(Material(y) \wedge consistsOf(x,y))$

Considering that *E18 Physical Thing* subsumes *E26 Physical Feature*, (Ax1) is misguided, at least if CIDOC understands features like holes as *immaterial* entities (as it seems). Hence, we have not included in the modules the entirety of axioms that are present in the CIDOC-Erlangen; further work on their analysis and the analysis of CIDOC's cardinalities is required.

As a first example, let us assume that we need to represent museological data about statues. These can include data about statues' dimensions, creators, creation dates, material types, identifiers, and the museums where they are preserved. To represent these data in our framework it is sufficient to use the `cidoc:artifact-module` and the `cidoc:actor-module`. The former contains the basic modeling elements for statues, whereas the latter is required to represent the statues' creators. Hence, differently from the current release of CIDOC, end-users can now exploit the ontology by reusing only the modules that are relevant for their tasks. In addition, as said, in a data modeling

[18] For simplicity, we omit CIDOC's identifiers. Also, looking at (Def1) some unary predicates can be derived from relations' domain/range restrictions. We include them to facilitate the understanding of the formula.

[19] Recall that the Distributed Modeling Language (DOL) [22] can be used to handle and link alternative formalizations of the same conceptual model.

[20] Looking at (Ax1), *Material* stands for material types and not for amounts of matter in sense of, e.g., [7].

scenario one may not desire the explicit representation of temporal phenomena like the production events leading to the statues and their time-regions. Although ontologically coherent, this approach would lead to verbose data models at the expenses of computational resources. By introducing shortcuts on the line of (Def1) we can link statues to their creation dates and creators while keeping a simple data representation.

As a second example, we consider the design of a domain-specific ontology based on CIDOC. OpenArchaeo is a semantic mediator for archaeological datasets currently hosted by the French infrastructure Huma-Num.[21] It interconnects multiple datasets by using an ontology dedicated to archeology [8,23], which is based on CIDOC plus some of its extensions, e.g., CRMsci[22] and CRMba,[23] among others. One of the most relevant classes is the event of (archeological) site discovery represented by *S19 Encounter Event* from CRMsci, which is a subclass of *S4 Observation*, the latter subsumed by *E13 Attribute Assignment*. The site discovery event (*i*) is carried out by a *E21 Person* who is member of a *E40 Legal Body*; (*ii*) took place on a *E27 Site*, which has a place as location; (*iii*) is linked to a *E52 Time-Span* with dates; and (*iv*) found some artifacts. This ontology was developed by taking into account the whole CIDOC, whereas with our approach one would require the `cidoc:actor-module`, the `cidoc:artifact-module`, and the `cidoc:attribute-assignment-activity-module` including both their imported modules and the `cidoc:top-module`, the latter used to consistently integrate all modules. In principle, from an ontology design perspective, the selective reuse of CIDOC could facilitate the development of the ontology, since one would not need to go through its entire taxonomy. For end-users, this may also facilitate the understanding of the ontology, since many of CIDOC's modeling elements would be left out.

7. Conclusions

In order to foster the use of ontologies for knowledge representation and data management in the area of cultural heritage, we presented in the paper a first ontological analysis and modularization of the CIDOC ontology. We focused on the latter because of its wide use in both research projects and institutions. Our contribution is twofold: first, by analysing CIDOC, the goal is to enhance and make transparent its ontological commitment. As a result, we have proposed to remove some classes from the ontology and to introduce some new modeling elements. Second, by modularizing it, the purpose is to facilitate its selective reuse, maintenance, and extension with domain-specific modules.

Future work to strengthen our proposal is required. First, both the analysis and modularization have to be extended to the whole ontology, conceptual objects and relations included. The analysis of relations requires a careful evaluation of their cardinalities to check whether these are consistent with the intended meaning of the related classes. Second, a testing benchmark is necessary to evaluate both the ontology resulting from the analysis and its modular architecture. From a usability perspective, we plan to exploit the ontology modules in research projects and to test their impact on data management practices. Finally, a stable formalization of CIDOC in a language like FOL is a desiderata to unambiguously characterize its elements. This could be based on the work presented

[21] `http://openarchaeo.huma-num.fr/explorateur/sourcesSelect`, last accessed in March 2020.
[22] `http://www.cidoc-crm.org/crmsci/`, last accessed in March 2020.
[23] `http://www.cidoc-crm.org/crmba/`, last accessed in March 2020.

in [3] possibly revised and extended by the work we presented. This formalization could be then used as a foundational basis for the computational treatment of the ontology.

References

[1] Boeuf PL, Doerr M, Ore CE, Stead S, et al. Definition of the CIDOC Conceptual Reference Model. Version 6.2.1. ICOM/CIDOC Documentation Standards Group CIDOC CRM SIG. 2015;.

[2] Bruseker G, Carboni N, Guillem A. Cultural heritage data management: the role of formal ontology and CIDOC CRM. In: Heritage and Archaeology in the Digital Age. Springer; 2017. p. 93–131.

[3] Meghini C, Doerr M. A first-order logic expression of the CIDOC Conceptual Reference Model. International Journal of Metadata, Semantics and Ontologies. 2018;13(2):131–149.

[4] Wahlberg TH. The endurance/perdurance controversy is no storm in a teacup. Axiomathes. 2014;24(4):463–482.

[5] Guarino N, Welty CA. An overview of OntoClean. In: Staab S, Studer R, editors. Handbook on ontologies. Springer; 2009. p. 201–220.

[6] Guizzardi G. Ontological foundations for structural conceptual models; 2005.

[7] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. WonderWeb Deliverable D18. Laboratory for Applied Ontology ISTC-CNR; 2003.

[8] Marlet O, Francart T, Markhoff B, Rodier X. OpenArchaeo for usable semantic interoperability. In: Proceedings of the 1st International Workshop on Open Data and Ontologies for Cultural Heritage (ODOCH19). vol. 2375. CEUR; 2019. p. 5–14. Available from: http://ceur-ws.org/Vol-2375/paper1.pdf.

[9] Moraitou E, Aliprantis J, Christodoulou Y, Teneketzis A, Caridakis G. Semantic Bridging of Cultural Heritage Disciplines and Tasks. Heritage. 2019;2(1):611–630.

[10] Hiebel G, Doerr M, Eide Ø. CRMgeo: A spatiotemporal extension of CIDOC-CRM. International Journal on Digital Libraries. 2017;18(4):271–279.

[11] Sider T. Four Dimensionalism. An Ontology of Persistence and Time. Oxford University Press; 2001.

[12] Bottazzi E, Ferrario R. Preliminaries to a DOLCE ontology of organisations. International Journal of Business Process Integration and Management. 2009;4(4):225–238.

[13] Porello D, Bottazzi E, Ferrario R. The Ontology of Group Agency. In: Proc. of the 8th Int. Conf. on Formal Ontology in Information Systems. IOS Press; 2014. p. 183–196.

[14] Masolo C, Vieu L, Bottazzi E, Catenacci C, Ferrario R, Gangemi A, et al. Social roles and their descriptions. In: Principles of Knowledge Representation and Reasoning. AAAI Press; 2004. p. 267–277.

[15] Masolo C, Guizzardi G, Vieu L, Bottazzi E, Ferrario R, et al. Relational roles and qua-individuals. In: AAAI Fall Symposium on Roles, an interdisciplinary perspective. AAAI PRESS-MIT PRESS; 2005. p. 103–112.

[16] Sanfilippo EM. Feature-based product modelling: an ontological approach. International Journal of Computer Integrated Manufacturing. 2018;31(11):1097–1110.

[17] Ben Abbès S, Scheuermann A, Meilender T, D'Aquin M. Characterizing Modular Ontologies. In: Proc. of the 7th Int. Conf. on Formal Ontologies in Information Systems. IOS Press; 2012. p. 13–25.

[18] Khan ZC, Keet CM. An empirically-based framework for ontology modularisation. Applied Ontology. 2015;10(3-4):171–195.

[19] dAquin M, Schlicht A, Stuckenschmidt H, Sabou M. Criteria and evaluation for ontology modularization techniques. In: Modular ontologies. Springer; 2009. p. 67–89.

[20] Carriero VA, Gangemi A, Mancinelli ML, Marinucci L, Nuzzolese AG, Presutti V, et al. Arco: the italian cultural heritage knowledge graph. In: International Semantic Web Conference. Springer; 2019. p. 36–52.

[21] Rector A, Brandt S, Drummond N, Horridge M, Pulestin C, Stevens R. Engineering use cases for modular development of ontologies in OWL. Applied Ontology. 2012;7(2):113–132.

[22] Mossakowski T, Codescu M, Neuhaus F, Kutz O. The distributed ontology, modeling and specification language–DOL. In: The road to universal logic. Springer; 2015. p. 489–520.

[23] Marlet O, Rodier X. A way to express the reliability of archaeological data: data traceability at the Laboratoire Archologie et Territoires (Tours, France). International Journal on Digital Libraries. 2019;.

This page intentionally left blank

III. Intentionality and Embodiment

This page intentionally left blank

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200666

Needs and Intentionality

An Ontological Analysis and an Application to Public Services

Luca BICCHERI [a], Roberta FERRARIO [b] and Daniele PORELLO [b]
[a] *University of Urbino (Italy)*
[b] *ISTC-CNR, Laboratory for Applied Ontology, Trento (Italy)*

Abstract. A thorough understanding of what needs are is fundamental for designing well-behaved information systems for many social applications and in particular for public services. Talking about needs pervades indeed the jargon of Public Administrations when motivating their service offering. In this paper, we propose an ontological analysis of needs, aiming at a principled disentangling of the different uses of the term. We leverage philosophical tradition on intentionality, for its rich understanding of mental entities, we compare it with the well-established BDI (Belief-Desire-Intention) tradition in knowledge representation, and we propose a formalisation of needs within the foundational ontology DOLCE. Throughout the paper, we motivate our analysis focusing on needs in public services.

Keywords. Need, satisfier, mental attitude, intentionality, intentional content, intentional object, public service, Public Administration.

1. Introduction

The mission of every Public Administration (PA) of a Welfare State is that of promoting the fulfillment of human rights and the well-being of the citizens of the community within which it operates. Every citizen has the right to food, health, shelter, justice, safety. Public services are thus aimed at improving the conditions of citizens, by granting a decent level in all the fundamental aspects of living.

A way of phrasing the PA mission is by saying that, whenever a citizen's right fails to be actualised, a citizen's *need* arises, and the satisfaction of that need is what should drive the PA in the first place.

Nonetheless, citizens' fruition of rights is not the only purpose for the implementation of public services, as well as the violation of fundamental rights is not the only trigger of citizen's needs. Citizens have life plans, goals, desires that drive their actions and their interactions within a society. In many cases, to pursue their own private goals within a society, people have to interact with the PA. For instance, when a citizen wants to buy a car to freely circulate, they have to interact with the authority for motor vehicles in order to satisfy their goal. That is, one *needs* a driving licence to circulate. Hence, a second view of need emerges here, not arising from violations or rights, rather emerging from the goals and the viable means to satisfy those goals[1]. Accordingly, a demand for

[1]An alternative way to present this example is to say that holding a driving license is a *requirement* to circulate in public streets. It is nonetheless a state that has to be reached in order to be able to execute the action

public services arises, requiring the PA to respond, hence public services may also be motivated by citizens' need to satisfy their goals.

We are here embracing a view of public services that can be termed *need-driven*. To properly articulate such a view, a thorough analysis of needs is in fact required.

Even before disentangling the heterogeneous notions of need at work, we have to separate *self-ascribed* and *hetero-ascribed* (e.g. from the PA) needs [1]. In the former case, a citizen believes to need something for which an action of the PA is required. Citizens can be entitled to request to the PA to enact a service and the PA may respond to such request; we will call this kind of services *reactive*. In the latter case, it is the PA that attributes a need to a citizen, usually by proposing a service to satisfy it; we will call these *proactive* services. Proactive services are considered particularly desirable as they reduce the charge on the side of the citizens and allow the PA to fully achieve its mission; on the other hand, they require the PA to be able to correctly ascribe needs to citizens.

In principle, citizens know better their own needs, as they have first person experience of what they miss, while the PA knows better which are its means to cope with needs. However, this is not always the case, for various reasons. First of all, some citizen (e.g. cognitively impaired citizens or unaccompanied minors) could be not aware – while the PA is – of being in need. Secondly, citizens may know to be in need, but they do not know what to demand to the PA, because they are not aware of the services the PA may offer. Also, sometimes the state of need is associated with pain, is pressing, and prevents people from taking prompt action (e.g. health conditions). Thus, in some situations proactive services are required. Moreover, there is a series of needs that are easily predictable for the PA, either because they emerge from the demands of the PA itself (e.g. the issuance of a driver license to allow citizens to drive) or because they are associated with the occurrence of regular life-events (as the birth of a child).

Correctly ascribing needs to citizens is the first step to implement useful proactive (or even predictive) services.[2] The second and fundamental step is to design services for each particular kind of citizen (youngsters, families, elderly, people with disabilities, foreigners, citizens belonging to minorities etc.) to help them satisfy their specific needs. The final phases are the delivery of services (specific deliveries to specific persons) and their ex-post evaluation. When designing a service system, all these steps are important.

While most of these phases require massive data gathering, the design phase, whose importance is well acknowledged and explained (for instance in [3]), presupposes a thorough analysis in (at least) two directions. The first one is prominently sociological and is dedicated to understand what people in certain conditions need. The other is conceptual and focuses on how the information required by the system can be organised and represented in a structured and well-founded manner, to promote accessibility and interoperability. This paper provides a contribution to the conceptual analysis at the design phase, by concentrating on the notion of need. When one talks about need-driven services, at least three different concepts are connected with the use of the term "need", both in com-

of circulating in a public street. On the other hand, it is in virtue of the final goal of circulating on public streets that citizens aim at holding a driving license, that is thus instrumental with respect to the end goal. We thank an anonymous reviewer for the suggestion.

[2]A fairly intuitive definition of proactive vs. predictive services is provided in [2], where the former are the services connected to needs whose emergence can be foreseen, but without an exact timing, while the latter are those triggered by needs for which the PA can foresee not only that they will emerge, but also when.

mon sense and in scientific discourse [4]. Three very common uses of the term that can be found in the literature are listed below:

1. an event or a state that, if realised, brings about an end goal for an agent (these are referred to in some literature as “satisfiers” – see, e.g., [5] and [6]), for instance the need for a train, bus, or shuttle service to connect the airport with the city center;
2. an event or a state that is, besides sufficient, necessary to realise for the achievement of an end goal, i.e. if not realised, it prevents the end goal from being achieved (in the philosophical debate these have been defined “instrumental needs” [7], i.e. needs whose satisfaction is instrumental to the achievement of some end goal), as for example the need of having a driver license issued for being allowed to circulate on public streets with a car;
3. a very fundamental condition, whose absence causes an important damage for persons, universalisable goals [8] (these are sometimes called “absolute” or “basic” needs, sometimes “pre-conditions”). Examples are the need for food or freedom.

Need-driven services are therefore conjugated differently, according to the notion of need they refer to. In the first case, satisfying needs would mean finding solutions to help citizens to realise their own goals. In the second case, satisfying needs serves to provide the citizens with those means without which they could not achieve some of their goals. In the third case, copying with needs would translate into prioritizing the intervention on the most basic and important aspects of their citizens’ lives. For sure all these aspects are desirable and should be included in the implementation of need-driven services, but a conceptual clarification is mandatory to be able to reason on such complex scenarios.

The objective of this paper is thus to provide a clear map of the various notions of need, by proposing a conceptual analysis as well as a formal ontology of needs. As suggested by the previous discussion, there is a connection between goals, as well as other mental attitudes, and needs. For this reason, we shall confront our treatment with the BDI (Belief-Desire-Intention) approach to mental attitudes, which provides well-established models and tools in knowledge representation. However, as we will see, to cope with the complexity of the notion of need, we shall complement the BDI view with the rich tradition of intentionalty in philosophy of mind. To provide a formalisation of our approach, we shall place our treatment within the foundational ontology DOLCE, as it is expressive enough for our purposes. The remainder of this paper is organised as follows. Section 2 approaches mental states on the basis of the philosophical view of intentionality, relates this view to BDI, and discusses this view wrt. the assumptions of DOLCE. Section 3 focuses then on needs, by discussing them as mental states and articulating the intentional nature of needs. Section 4 proposes our formal ontology of needs. Section 5 illustrates our view by means of an example of needs in public services. Section 6 concludes.

2. Mental states and intentionality

In this paper, we locate needs, from an ontological point of view, among those mental states that exhibit the feature of intentionality, namely the property of *being about* something, i.e. about intentional objects. Although there are different types of mental states (e.g. beliefs, desires, expectations, perceptions etc. and also needs), they all share the property of *intentionality*, which somehow links them to their *intentional object* (e.g. if Maria loves Luigi, then "Luigi" is the intentional object of Maria's intentional state of love). In contemporary philosophical literature, reams of paper have been written on the notion of "intentional object"; nevertheless, there is currently no consensus about the ontological status of such objects. In fact, mental states can be about very different kinds of entities, for example, ordinary objects, properties or qualities, events, states of affairs. Besides, even nonexistent and impossible objects might be mental states' intentional objects (in literature, classic examples are Zeus, Pegasus, the golden mountain, the round square, etc.). Anyway, before taking into consideration intentional objects, we have to introduce some philosophical positions that we endorse in this paper regarding intentionality.

2.1. Aboutness as directedness or reference

According to a traditional picture of intentionality, which stretches back to Twardowski, this relation can be subdivided into two types of intentional binary properties that can be instantiated by mental states, namely *having a content* (*directedness* or *aboutness*$_1$) and *being about an object* (*reference* or *aboutness*$_2$). As Haldane suggests [9, pp. 17–18], the difference between directedness and reference is that the former puts a mental state in connection with an intentional content, an entity whose kind is typically classified as abstract, the latter links a mental state with an intentional object. It is worth just briefly outlining that the intentionality of a mental state is independent of the existence of its objects, i.e. we can think about existent objects as well as nonexistent ones. How shall we explain this fact?

To answer this question, we should understand how directedness and reference interact. Regardless of the fact that the objects of mental states exist, we are able to cognitively grasp something, namely a contentful meaning. Directedness is equivalent to *contentfulness* and pertains to every mental state. Having an intentional content is a feature which is independent of the existence of objects because, at least in some cases, there are no objects which we can refer to, but still our mental states have a content. Thus, if Maria fears Boogymen, then Maria's mental state is related by means of directedness to the intentional content expressed by the term "Boogymen". Instead, when mental states' objects exist, directedndess has the "power", so to speak, to identify a specific object, an intentional object, which we can refer to. For example, imagine that Paul admires Varenne, the best trotter of all time. In this case, it is in virtue of directedness that his mental state is related to an intentional object through a reference relationship, because Varenne, among all existent entities, is the one which satisfies the intentional concept expressed by the definite description "the best trotter of all time". So, it is in virtue of having a content that mental states can refer to objects, provided that such objects exist. By now it should have become clear that the intentional picture sketched above is the counterpart, at the level of philosophy of mind, of Frege's notion of sense[3]: the sense of an expression is, in the Fregean view, what enables to determine its reference.

[3]Crane stressed [10, p. 21] that there is a complex relation between *intentionality* and *intensionality*, namely, there are linguistic contexts in which sentences do not satisfy certain extensional criteria.

2.2. Intentional objects as roles

Now let's turn our attention to intentional objects. As already mentioned at the beginning of this section, the locution "intentional object" can be used in different senses. Trivially, every kind of entity can be an intentional object inasmuch as an entity is thought of. This leads Crane to affirm that an intentional object is not a particular kind of entity among others since, as long as something is the "target" of a mental state, every kind of entity counts as an intentional object. In other words, intentional objects are not a genuine category to be included in the ontological inventory, in Crane's words "they have no nature of their own" [9, p. 16]. We agree with Crane that our mental states can be about (in terms of reference) different types of entities and, in our account, we treat them as *roles*.

The notion of "role" has been discussed from several perspectives in quite different disciplines, from sociology to philosophy, as well as knowledge engineering and formal ontology[4]. We take here the classical approach for analyzing roles based on the works of Sowa [17,18], and Guarino [19]. In short, Sowa claims that a role is a monadic property which can be predicated of different entities; in technical terms, different entities can play the same role. For example, the role "customer" can be played by a person as well as by a company. Furthermore, an entity plays a role only with respect to a "pattern of relationships". For instance, the role "university student" holds only within a binary relation of "enrollment" to a university. Guarino adds constraints to Sowa's theory affirming that roles must be *founded* and *anti-rigid*. The notion of foundation or, better, generic foundation, is a binary relation between species (kinds of objects). It has been formulated by Husserl and expresses the idea that a property[5] α cannot exist as such except in a more comprehensive unity whit a property β[6]. For example, a university student is such as founded on a university, and a wife is such as founded on a husband and vice versa. The former is a one-sided foundation relation (not symmetric), the latter is a two-sided or mutual foundation (symmetric) [20, p. 128]. Simons defines [9, p. 125] generic foundation as follows: property α is founded on property β if and only if any instance x of α is necessarily associated with an instance y of β which is not related to x by a part-of relation.

Thus roles are founded properties, for example, "supplier" is founded on "customer" and vice versa (besides, the same person could play different roles simultaneously or at different times). It is worth noting that if an entity y instantiates the property β = person and the property α = student, then y can cease to be a student (or whatever role you

[4]Some important works on roles are: [11], [12], [13],[14]. In 2007 the journal *Applied Ontology* published a whole special issue, in which the notion of role has been analyzed under an interdisciplinary perspective: [15]. A more recent approach on roles has been proposed by Mizoguchi and colleagues [16].

[5]Simon and Correia use the terms "kind" or "species", but we rather preferred to use the more generic term "property", in order to prevent confusion with how such terms are used in the literature on knowledge representation.

[6]The notion of "foundation" is used by Husserl to characterise the concept of "pregnant whole", that is a whole in which each part is "foundationally connected, directly or indirectly, with every other, and no part of the whole so formed is founded on anything else outside the whole" [20, p. 122]. Husserl interpreted the foundation relation as a *necessary association* between kinds. The father of phenomenology conceptualized a pregnant whole as a more comprehensive union between kinds, differently from a mereological aggregate. Simons has stressed [20, pp. 122–125] that if we formulated the connection between instances of different kinds in terms of parthood or proper parthood, we would not be able to capture Husserl's original intuition about pregnant whole and foundation.

like) without changing its identity conditions, but can't cease to be a person, otherwise *y* would be a different entity. In other words, "being a person" is a rigid property, that is in every possible world this property applies to all its instances necessarily. Instead, "being a student" is an anti-rigid property, namely, it doesn't apply necessarily to all its instances.

This being said, let us come back to the analysis of intentional objects. What is it to be an intentional object? With respect to a certain entity, we suggest that being an intentional object is nothing but playing a certain role within an intentional relationship, that is aboutness$_2$.

We believe that aboutness$_2$ is a one-sided relation holding between an agent's mental state 'X' and an entity 'Y'. Now, the point is that the role of intentional object z is played by y to the extent that x is about$_2$ y. For instance, suppose that Paul admires Varenne. Given that Varenne is related to Paul's mental state by means of aboutness$_2$, Varenne as such plays the role of intentional object. It is worth noting that we are not embracing here an ontological multiplicative approach, that is we are not stating that the intentional object Varenne is a different entity from Varenne. All we are saying is that only insofar Varenne is the "target" of Paul's thought it instantiates a certain quality or property, which is the property of being an intentional object. We believe that two interesting consequences can be deduced from these observations.

First of all, let the property Φ stand for the property "being an intentional object" and Ψ for the property "being a mental state". We can affirm that any instance x of Φ is necessarily associated with an instance y of Ψ which is not related to x by a part-of relation (where the association relationship between instances corresponds to aboutness$_2$). So Φ is founded on Ψ. In addition, the property Φ is anti-rigid. In fact, it is only to the extent to which an entity is involved in an intentional relation (aboutness$_2$) with a particular mental state that this entity instantiates the property Φ. Since Φ is a founded and anti-rigid property, Φ is a role. To sum up, Φ is a monadic property that can be predicated of different entities and this is a welcome result if one wants to maintain the idea that ontologically different kinds of objects can be intentional objects[7].

2.3. *Intentional contents and social concepts in* DOLCE

It is time to make some clarifications on the approach we are developing on intentionality, adapting the traditional philosophical picture introduced above to DOLCE, that is the top-level ontology on which we decided to ground this work. The choice of DOLCE is motivated by its ontological commitment to its being tailored to common-sense representations of cognitive agents, rather than on the constitution of the "reality" as prescribed

[7] Sometimes individuating entities that play the role of being an intentional object could be a tricky matter. For instance, if Maria believes that Naples is the largest city of Southern Italy, one may claim that the intentional object wrt. Maria's mental state is the state of affairs "Naples is the largest city of Southern Italy". But others could argue that, actually, we can individuate two different intentional objects, that is "Naples" and the property "being the largest city of Southern Italy". In fact, given that a state of affairs is made up of its constituents, holding this thesis would be equivalent to saying that Maria's intentional state of belief is about (in terms of reference or aboutness$_2$) the property "being the largest city of Southern Italy" instantiated by "Naples", namely the state of affairs "Naples is the largest city of Southern Italy". Hence we suggest that it would be more accurate to say that Maria's intentional state is derivatively about the state of affairs *qua* complex entity. See [21, p. 29].

by science. This feature brings to the core of DOLCE the importance of modelling the mental and the social realms.

With this in mind, we know that intentionality is a relation that can be subdivided into two types of intentional binary relations, namely having a content (directedness or aboutness$_1$) and being about an object (reference or aboutness$_2$). Concerning the first relation, we said that the intentional content is a contentful meaning, something that we are able to grasp cognitively, and that a mental state can succeed in individuating intentional objects (in terms of reference) in virtue of its having an intentional content. Ontologically speaking, we interpret intentional content with the notion of *Social Concept*[8]. Social concepts are described in [22][9] as particulars that are created and accepted by a community of intentional agents and defined by descriptions encoded in linguistic expressions. Concepts can classify different kinds of entities, for example, objects as well as events[10]. Since our aim in this paper is representing the social domain of public services that, similarly as in [23], we see as events, we will specifically take into consideration concepts that classify events.

For example, in some countries parents are automatically entitled to receive a child benefit when a child is born. "Providing a child benefit" is a proactive service that can be seen as a concept whose description could be recorded in a PA's official document. Each service provision counts as a single event that is classified by the concept defined by the description. Concepts classify entities that satisfy all the constraints in the concept definition. Furthermore, the classification relation is associated with a time parameter that identifies a specific interval in which the classified entity satisfies the definition, more formally $\mathsf{cf}(x,y,t)$ stands for "the social concept x classifies the event y at time t". In the example, if the social concept is "Providing a child benefit", all the events that satisfy the constraints set by it at a certain time are the events classified by it at that time, for example the event "Giovanni sends the payment on Lucia's bank account" at 3.52 PM on April 14, 2020 satisfies such constraints and is thus classified as a social concept of "Providing child benefit".

This being said, from an intentional standpoint, when a citizen needs that the PA provides them with a child benefit, their mental state of need will be directed (*directedness*) towards the concept "PA providing a child benefit". Furthermore, in virtue of this concept, the citizen's mental state will refer to (*reference*) an intentional object that is one among a set of events that are classified by that concept (provided there are any) .

It is worth noting that two different concepts can classify the same entity, for instance, the event "Giovanni sends the payment on Lucia's bank account" could be classified by two different concepts, for example "PA providing a child benefit" if certain conditions hold, like that Giovanni works in a PA and that payment is connected to child benefits etc. and "paying off a debt to a friend" if other conditions hold (at different times

[8]Following DOLCE, in this paper we endorse a *social* view of concepts, i.e. we see them as strictly connected to (natural) language and intersubjective. We do not take them to be "private" entities existing only in the mind of one agent. That concepts are dependent on the social nature and practices of language is very well established in the philosophical literature, and we are here endorsing this position, without entering in the debate on whether private concepts exist.

[9]A similar account was previously presented in [13] for social roles, defined as social concepts that classify enduring entities, like objects.

[10]With the term "event" we are referring here generally to perduring entities, including states, processes and events in a stricter sense. From now on we will use "event" in this wide sense.

or simultaneously). The same event can then be grasped in different ways, similarly as it happens with concepts in many theories of intentionality.

This highlights that concepts can be considered good candidates for being intentional contents. Another reason is that concepts are somehow "shareable" between different agents. Since descriptions, and therefore social concepts, are accepted by a community of intentional agents, different agents can cognitively grasp the same concept and hence refer to the same intentional object. Furthermore, it is worth noticing that, analogously as for intentional contents that fail to refer to nonexistent objects (like Pegasus), also social concepts can exist without classifying any entity (at a certain time); as an example, we can talk about the king of Italy, though at the moment there is no entity classified by such concept. The current discussion is depicted in Figure 1.

Lastly, intentional contents can be seen as roles just like intentional objects. In fact, "Being an intentional content" seems to be a founded and anti-rigid property. Briefly, we note that: A) any instance of the property "being an intentional content" is necessarily associated with an instance of the property "being a mental state" without involving a part-of relation between instances and through an association relationship that corresponds to the intentional relationship of directedness. Thus the former property is founded on the latter. B) If we see concepts as entities that play the role of intentional content, only as long as a concept is involved in a directedness relationship with a particular mental state this concept instantiates the property of being an intentional content. Hence, being an intentional content is an anti-rigid property.

To sum up, we have characterised mental states in DOLCE as being about$_1$ (directed to) social concepts and being about$_2$ one of the entities that are classified by those concepts. In the next section we will see how this analysis can be specified to account for needs.

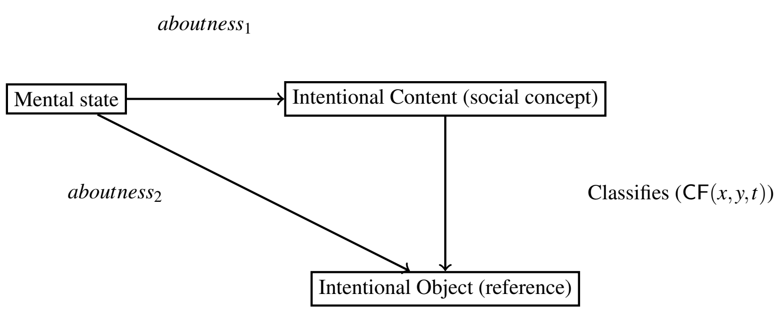


Figure 1. Aboutness relations

3. Needs as mental attitudes

As anticipated, to represent the notion of need, we start from the perspective on mental attitudes of BDI, because it is based on a long established tradition in knowledge representation[11], and it has been implemented in many widely used applications (e.g. JAM, Jason, and SPARK[12]).

Most approaches based on BDI have been developed for reasoning and planning with mental attitudes but, as far as we know, not much effort has been spent to define what they are and what they are about from an ontological point of view[13]. There seems to be a tension between the fact that mental attitudes are traditionally conceived of as being about propositions (they are also often identified with "propositional attitudes"), and the fact that a real planning agent, while realising a proposition, brings about changes in the real world. Moreover, the fact that mental attitudes are about something is implicit in the BDI representation framework. For this reason, the intentionality-based view of mental attitudes nicely complements the BDI view.

In classic BDI approaches, beliefs, desires and intentions are seen as the fundamental mental entities dependent on an intentional agent, with different features (which in the philosophical debates have been called "intentional modes" [31], [10]): while beliefs are informational – they constitute the information available to the agent –, desires are motivational – they motivate the agent to act – and intentions are deliberative – given bounded rationality, they allow the agent to commit to a course of action, until it is fulfilled or it fails.

Turning now to needs, in light of our discussion of intentionality, we can ask a number of questions: which kind of mental states are needs? What is their intentional content? What is their intentional object(s)? A first assumption, which is quite well established in the literature, is that needs are mental states separated from desires, beliefs, and intentions.

Let's now recall the different uses of the term "need" listed in the Introduction:

1. an event that, if realised, brings about an end goal for an agent (the need of taking a train/bus/shuttle to go to the airport);
2. an event that, if not realised, prevents the end goal from being achieved (the need of having a driver license issued to circulate in public streets);
3. a very fundamental condition, whose absence harms the person (the need for food)[14].

A first thing to notice, these are not definitions of what is a need, but of *what is needed*, of what such needs are about *prima facie*. In other terms, these correspond of what in the literature are called "satisfiers". As we shall see, some of the features of the satisfiers identify different concepts of need.

In [7], needs are construed as goals, where a goal is intended as a chosen desire, i.e. a desire that has been selected at a certain time among possibly conflicting desires, as the one to be pursued. Goals can be subdivided in end goals and instrumental goals; end goals are goals that are pursued *per se*, while instrumental goals are goals whose satisfier

[11] Seminal contributions are [24], [25] and [26].

[12] See http://www.marcush.net/IRS/, http://jason.sourceforge.net/wp/, and http://www.ai.sri.com/~spark/

[13] There are also approaches to BDI using BFO (Basic Formal Ontology), e.g. [27] and [28], and UFO (Unified Foundational Ontology), e.g. [29] and [30]

[14] One could object that in all these cases, the need is about an endurant, rather than a perdurant, for instance the need of a train, of a driving license or of food. But our intuition is that these are just elliptical ways to express needs of some events to happen. For example, someone does not just need a train to exist, they need to be able to take it.

– in this case an event –, if realised, brings about the satisfaction of the end goal (makes the event which the end goal is about occur).

If we look back at the three uses, the first two seem to refer to satisfiers of instrumental goals, while the latter of an end goal. But let's analyse them in light of the previous discussion on intentionality.

The first thing to be noticed is that all three refer to *an* event having certain properties. By viewing needs as mental attitudes, as we argued, they have: *i*) as a content a concept (i.e they are mental states about_1 a concept) that classifies events with certain characteristics; *ii*) as a reference (about_2) one among the classified events (the satisfier).

Going further, let's call need_1 and need_2 the mental attitudes associated with uses 1. and 2. respectively; they are apt to be represented by instrumental goals. But what distinguishes one from the other? Our proposal is to distinguish them depending on the relation that their satisfiers bear with the events that are the intentional object of the connected end goal. Instrumental goals have satisfiers that, if realised, can bring about an event that is the intentional object of the end goal.

The point is: instrumental goals that are need_1 require satisfiers that are *sufficient* for realising end goals; by contrast, instrumental goals that are need_2 require satisfiers that are, besides sufficient, also *necessary* for realising end goals. This seems to capture our common sense distinction between saying "I need to do x, y or z to obtain w" ("x, y or z could be of help") and "I need to do x to obtain w", the latter conveying a stronger message and an implicature ("I cannot do without it").

Point 3. seems instead to be connected to a different kind of need, what has been defined by some scholars an "absolute need". We thus introduce need_3 to talk about those needs whose satisfiers are states, conditions which are aimed at independently of any further goal and whose absence causes in the agent a consequent mental state (a state of need).

These three notions of need all play an important role for the PA: need_1 can be used to identify the different services that the PA may offer to help citizens to achieve their goal; need_2 can be used to single out those services that should be necessarily provided to citizens when the PA "legitimates" their goals on an institutional level and thus assumes them; need_3 can be used to drive the priorities of the PA towards goals which, if not reached, can compromise the well-being of the citizens.

4. A DOLCE-based ontology of intentional mental attitudes and needs

We develop our formal approach within DOLCE, so the following axioms are designed to be added to DOLCE, cf. [32][15]. Mental states (MS) are a type of states (ST, cf. [32], p. 24), i.e. a type of perdurants (aka events, which are disjoint from endurants, from time intervals, and from concepts)[16], cf. axiom (a1)[17]. Participation $\mathsf{pc}(x,y,z)$ (see [32], p.

[15]We implemented our ontology and we tested its consistency as well as its provable consequences, cf. https://github.com/diporello/DOLCE-mental_states-needs/blob/master/dolce_needs.p for the details. This treatment is also compatible with [33]. We present an excerpt of the axioms here.

[16]This is the view of concepts in DOLCE-CORE, cf. [22].

[17]States in DOLCE are perdurants that are cumulative and homeomeric. E.g. "sitting", the sum of two instances of sitting can still be a sitting and the parts of sitting are all sitting states. By viewing mental states as states (and not processes), we abstract here from the internal articulation of mental states. Moreover, here states are perdurants, so states happening at different times are different. A contrasting view of states is in [34].

20) is a ternary relation connecting an entity, a perdurant (i.e. also a state), and a time interval (T, cf. [32], p. 74). Existence is represented by the binary relation "present at", $\mathsf{pre}(x,t)$ (cf. [32], p. 37). Axiom (a2) states that every mental state depends on an agent who participates to that state (APO, agentive physical object, cf. [32], p. 24). Moreover, a mental state is private to the agent: for all time intervals, the same agent participates to the mental state, cf. (a3).

a1 $\mathsf{MS}(x) \to \mathsf{ST}(x)$

a2 $\mathsf{MS}(x) \to (\exists y\, t.(\mathsf{APO}(y) \wedge \mathsf{pc}(y,x,t)))$

a3 $\mathsf{MS}(x) \wedge \mathsf{APO}(x_1) \wedge \mathsf{APO}(x_2) \wedge \mathsf{pc}(x_1,x,t) \wedge \mathsf{pc}(x_2,x,t') \to x_1 = x_2$

a4 $\mathsf{about}_1(x,y,t) \to \mathsf{MS}(x) \wedge \mathsf{C}(y) \wedge \mathsf{T}(t)$

a5 $\mathsf{MS}(x) \to \exists y\, t.(\mathsf{about}_1(x,y,t))$

a6 $\mathsf{about}_1(x,y,t) \wedge \mathsf{about}_1(x,y',t') \to y = y'$

d1 $\mathsf{about}_2(x,z,t) \leftrightarrow (\mathsf{MS}(x) \wedge \mathsf{T}(t) \wedge \exists y.(\mathsf{about}_1(x,y,t) \wedge \mathsf{cf}(y,z,t)))$

d2 $\mathsf{intCon}(y,t) \leftrightarrow \exists x.\mathsf{about}_1(x,y,t)$

d3 $\mathsf{intObj}(z,t) \leftrightarrow \exists xy.(\mathsf{about}_1(x,y,t) \wedge \mathsf{about}_2(x,z,t))$

The intentional content of mental states is treated in (a4)–(a6): about_1 relates mental states and concepts, as we discussed in Section 2, cf. (a4). Every mental state must be about_1 a concept, cf. (a5), and by (a6) a mental state is associated to exactly one concept. Hence, two mental states that are about different contents must be different.[18]

The intentional object of mental states is expressed by the relation about_2, cf. (d1). According to (d1), the reference of a mental state can be undetermined, if more entities are classified by the concept, or even empty, if the concept does not classify anything at a time. Also, the reference may change through time, depending on which entities are classified by the content of the mental state at that time.[19]

As we discussed in Section 2, intentional objects and intentional contents are *roles* played, respectively, by concepts and general entities, cf. (d2) and (d3).

We start the treatment of the different notions of need and of the relationship between needs and goals, by introducing the view of needs as mental states. By specialising the type of mental states, we can represent beliefs, desires, intentions, as well as needs, as mental states, i.e. $\mathsf{BS}(x)$, $\mathsf{DS}(x)$, $\mathsf{IS}(x)$, $\mathsf{NS}(x)$ are disjoint subtypes of MS. This view assumes, for instance, that a state of need differs from a state of desire.[20]

To reason about mental states (*à la* BDI), we introduce the relations $\mathsf{bel}(i,y,t)$, $\mathsf{des}(i,y,t)$ and $\mathsf{int}(i,y,t)$, whose meaning is "i believes, desires, intends y at t". These relations are defined by constraints such as (d4).[21] Accordingly, to express the content

[18]The motivation for (a6) is that we do not want to enter the complex issue of the internal structure of a mental state and its relation with the structure of its content. Whether the internal structure of a mental state resembles the logical structure of its content is a very challenging open question of cognitive science.

[19]Concepts in DOLCE must be instantiated in at least one possible world. This excludes impossible concepts, e.g. the round square. To enable impossible concepts as content of mental states, we need to abandon that constraint of DOLCE.

[20]We are admittedly vague in not endorsing a specific view of mental states. For instance, we do not want to enter the debate on the relation between mental states and neural (physical) states. The motivation is to propose an ontology of need that is hospitable to different views of mental states. Also, we do not enter the debate on what separates a need state from a desire state, viewing this problem as a direct question for cognitive science.

[21]Since concepts may classify events and since events are in the domain of DOLCE, assuming that the content of a mental attitude is a concept is not restrictive wrt. assuming that the content is a proposition, as in BDI. A

and the agent of a need state, we use the relation termed need0, cf. (d5): i needs0 y at time t if and only if there exists a need state x to which i participates that is about y. The relation need0 expresses that a certain agent is in a mental state of need about y at a time t.

d4 $\mathsf{des}(i,y,t) \leftrightarrow \exists x(\mathsf{DS}(x) \wedge \mathsf{pc}(i,x,t) \wedge \mathsf{about}_1(x,y,t))$

d5 $\mathsf{need0}(i,y,t) \leftrightarrow \exists x(\mathsf{NS}(x) \wedge \mathsf{pc}(i,x,t) \wedge \mathsf{about}_1(x,y,t))$

d6 $\mathsf{goal}(i,y,t) \leftrightarrow ((\mathsf{des}(i,y,t) \vee \mathsf{need0}(i,y,t)) \wedge \neg\exists w.((\mathsf{des}(i,w,t) \vee \mathsf{need0}(i,w,t)) \wedge \neg\exists z(\mathsf{cf}(y,z,t) \wedge \mathsf{cf}(w,z,t))$

Goals are defined as desires or needs of an agent that are not incompatible with other desires or needs (thus they can be chosen). We express goal attitudes by the relation $\mathsf{goal}(i,y,t)$ and we define them in (d6): agent i has goal y at t iff i desires or needs y at t and i has no desires or needs incompatible with y (where desires and needs are incompatible if they are about concepts that cannot be simultaneously satisfied). Also notice that Definition (d6) assumes that goals can be associated to a state of desire as well as to a state of need.

We separate *instrumental goals* $\mathsf{igoals}(x,y,t)$ and *end goals* $\mathsf{egoals}(x,y,t)$. To define these types of attitudes, we firstly introduce the notion of *satisfier* of a goal, (d7), i.e. an (existing) event that satisfies the content of the goal (need).[22] Secondly, we suppose that time intervals can be strictly ordered by $\prec$. Finally, instrumental goals are those that, if satisfied, bring about an event that satisfies a subsequent goal (the end of the instrumental goal), cf. (a8). By contrast, end goals do not need a further goal to be satisfied, in this sense, they are pursued *per se*.[23]

d7 $\mathsf{sat}(e,y,t) \leftrightarrow \mathsf{pre}(e,t) \wedge \mathsf{cf}(y,e,t)$

d8 $\mathsf{igoal}(x,y,t) \leftrightarrow (\mathsf{goal}(x,y,t) \wedge \exists y't'.(\mathsf{goal}(x,y',t') \wedge t \prec t' \wedge (\exists e.\mathsf{sat}(e,y,t) \rightarrow \exists e'.\mathsf{sat}(e',y',t'))))$

d9 $\mathsf{egoal}(x,y,t) \leftrightarrow \mathsf{goal}(x,y,t) \wedge \neg\mathsf{igoal}(x,y,t)$

d10 $\mathsf{need1}(x,y,t) \leftrightarrow \mathsf{igoal}(x,y,t)$

d11 $\mathsf{need2}(x,y,t) \leftrightarrow (\mathsf{igoal}(x,y,t) \wedge \exists y't'.(\mathsf{goal}(x,y',t') \wedge t \prec t' \wedge (\exists e.\mathsf{sat}(e,y,t) \leftrightarrow \exists e'.\mathsf{sat}(e',y',t'))))$

d12 $\mathsf{need3}(x,y,t) \rightarrow \mathsf{egoal}(x,y,t) \wedge \mathsf{need0}(x,y,t)$

We can now introduce our refined notions of need, besides needs as mental states (need0): need1 views needs as instrumental goals cf. (d10), whereas need2 views needs as goals whose satisfiers are both necessary and sufficient for the end goal, cf. (d11). Accordingly, need2 entails need1 (i.e. igoal). Finally, (d12) introduces the view of needs as end goals which are associated to a need state, corresponding to the absolute view of needs; accordingly, need3 implies need0. By contrast, neither need1 nor need2 implies need0: need1 and need2 are both igoal, which might be associated to desire, and not to need, states.

---

proposition classifies states of affairs as here concepts classify states or events. A similar approach, that views the content of mental attitudes as *types*, is proposed in [35].

[22] DOLCE views the elements of the domain as *possibilia* (by adopting the QS5 modal axioms), so the domain is intended to include possible events.

[23] To enable goals that have future satisfiers (e.g. at t_0 my goal is to eat supper at t_1), we can enable temporal indexes in the definition of the concepts, expanding on the formalisation of concepts in [13]. We leave this aspect for a future work.

To sum up, we formalised the four uses of the term "need" that we discuss in this paper: *i*) needs as mental states of agents (cf. NS(x), whose content and agent can be expressed by (d5)); *ii*) needs as instrumental goals, (cf. need1 and (d10).); *iii*) needs as necessary instrumental goals (need2 and (d11).); *iv*) needs as end goals associated to a state of need (need3 and (d12).)

5. A guiding example

From the just presented perspective, the PA services offering to citizens can be seen as a manner to provide satisfiers to needs. When the PA ascribes a need to a citizen, the content of such need is a social concept classifying events with certain features. At execution time, the PA brings about an event (with actual individual agents, specific time, place etc.) so that the executed service (hopefully) ends up being a satisfier of the need. In the ex-post evaluation, the PA will check whether the specific instance of service that has been executed possesses in fact all the properties that characterise the concept of the service and if the final outcome of the execution of the service is in fact the achievement of the citizens' goal.

Let's see now with an example why all the different notions of needs that have been formalised are useful when representing public services, in particular proactive services, activated in conjunction with a "life event" generating new needs for citizens (like having a child, looking for a new job, starting education…).

Suppose that Maria is a low income single woman who just had a baby. Very likely, such event puts Maria in a number of states of need NS(x), whose contents can be specified by our relation need0. Allegedly, the life event "having a child" will generate a series of new goals for Maria. A first goal will likely be an end goal, namely that this child enjoys a state of well-being, so that they have food, clothes, diapers, etc. Having enough money to be able to make all the required expenses is for sure an instrumental goal with respect to Maria's end goal. But, by analysing Maria's income and her economic situation in general, the PA may understand that there is an only way to grant Maria the possibility of affording all the necessary expenses, and that is to provide Maria with some more income. So, Maria's need to receive some additional income will trigger a single parent's child allowance, one of the services that the PA offers. Since providing a single parent's child allowance is a necessary instrumental goal w.r.t Maria's end goal, this can be represented by need2.

A different PA could, on the other hand, provide in addition different services to meet the same needs, like for instance delivering food at home and temporarily lending clothes and diapers; in such case, all these services may satisfy the content of an instrumental goal, a need1.

Finally, in the example, Maria is not the only person acquiring a need with the birth's event: the child also acquires their "absolute" needs, like the need for food, personal autonomy, or freedom, and these are needs represented by need3.

6. Concluding remarks

This paper aims to contribute by building a dialogue between different areas of research: that of philosophy of intentionality, of BDI, and of formal ontology. With respect to the study of intentionality, the paper contributes, firstly, by characterising, with the help of ontological analysis, the notions of intentional object and intentional content as roles. Secondly, it leverages philosophy of intentionality to study the ontological nature of the intentional content and object of mental attitudes, identifying the former with social concepts and the latter with the entities that are classified by such concepts. Thirdly, it contributes to the studies on the notion of need, representing needs as mental attitudes and distinguishing different notions associated with the term "need". Fourthly, it extends the DOLCE ontology of mental entities, by integrating it with the treatment of intentionality and by extending it with the identified notions of need. Finally, it lays the bases for designing a service system in which services are triggered by various kinds of needs.

References

[1] L. Biccheri and R. Ferrario. An analysis of the notion of need for the representation of public services. In A. Barton, S. Seppälä, and D. Porello, editors, *JOWO 2019 – The Joint Ontology Workshops, Proceedings of the Joint Ontology Workshops 2019, Episode 5: The Styrian Autumn of Ontology, Graz, Austria, September 23–25, 2019*, volume 2518. CEUR Workshop Proceedings, 2019.

[2] H. Scholta, W. Mertens, A. Reeve, and M. Kowalkiewicz. From one-stop-shop to no-stopshop: An e-government stage model. In I. Ramos, H. Krcmar, and V. Tuunainen, editors, *Proceedings of the 25th European Conference on Information Systems*, pages 918–934. Association for Information Systems, http://aisel.aisnet.org/ecis2017/, 2017.

[3] R. Sirendi, A. Mendoza, M. Barrier, K. Taveter, and L. Sterling. A conceptual framework for effective appropriation of proactive public e-services. In *Proceedings of the European Conference on e-Government, ECEG*, pages 213–221. Academic Conferences and Publishing International Limited, 2018.

[4] D. Gasper. Conceptualising human needs and wellbeing. In I. Gough and J. MacGregor, editors, *Wellbeing in Developing Countries: New Approaches and Research Strategies*, pages 47–70. Cambridge University Press, 2007.

[5] S. K. McLeod. Two philosophies of needs. *Polish Journal of Philosophy*, 9(1):33–50, 2015.

[6] M. Max-Neef. *Human Scale Development*. Apex Press, New York, 1991.

[7] C. Castelfranchi. *To believe* and *to feel*: The case of "needs". Technical Report FS-98-03, AAAI, 1998.

[8] L. Doyal and I. Gough. *A theory of human need*. Critical perspectives. Macmillan, 1991.

[9] J. Haldane. Brentano's problem. *Grazer Philosophische Studien*, 35:1–32, 1989.

[10] T. Crane. *Elements of Mind: An Introduction to the Philosophy of Mind*. Oxford: Oxford University Press, 2001.

[11] F. Steimann. On the representation of roles in object-oriented and conceptual modelling. *Data and Knowledge Engineering*, 35:83–106, 2000.

[12] F. Loebe. *An Analysis of Roles. Toward Ontology-Based Modelling Master's Thesis*. Master's thesis, University of Leipzig, 2003.

[13] C. Masolo, L. Vieu, E. Bottazzi, C. Catenacci, R. Ferrario, A. Gangemi, and N. Guarino. Social roles and their descriptions. In D. Dubois, C. Welty, and M. Williams, editors, *Ninth International Conference on the Principles of Knowledge Representation and Reasoning*, pages 267–277, Whistler Canada, 2004. AAAI Press.

[14] G. Boella and L. van der Torre. A foundational ontology of organizations and roles. In M. Baldoni and U. Endriss, editors, *Declarative Agent Languages and Technologies IV*. Springer, Berlin / Heidelberg, 2006.

[15] G. Boella, L. van der Torre, and H. Verhagen. Special issue: Roles, an interdisciplinary perspective. *Applied Ontology*, 2:81–207, 2007.

[16] R. Mizoguchi, A. Galton, Y. Kitamura, and K. Kozaki. Families of roles: A new theory of occurrent-dependent roles. *Applied Ontology*, 10(3-4):367–399, 2015.

[17] J. F. Sowa. Using a lexicon of canonical graphs in a semantic interpreter. In M. W. Evens, editor, *Relational models of the lexicon*. Cambridge University Press, 1998.

[18] J. F. Sowa. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Brooks/Cole, Pacific Grove, CA, 2000.

[19] N. Guarino. Concepts, attributes and arbitrary relations: Some linguistic and ontological criteria for structuring knowledge bases. *Data and Knowledge Engineering*, 8(2):249–261, 1992.

[20] P. Simons. Three essays in formal ontology. In B. Smith, editor, *Parts and Moments. Studies in Logic and Formal Ontology*, pages 111–256. Philosophia Verlag, Munich, 1982.

[21] F. Dretske. *Naturalizing the Mind*. The MIT Press, Cambridge (MA), 1995.

[22] S. Borgo and C. Masolo. Foundational choices in dolce. In *Theory and Applications of Ontology, volume 2, chapter 12*. Springer Verlag, 2008.

[23] R. Ferrario and N. Guarino. Towards an ontological foundation for services science. In J. Domingue, D. Fensel, and P. Traverso, editors, *Future Internet – FIS 2008*, pages 152–169, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[24] M. Bratman. *Intention, plans, and practical reason*. Harvard University Press, Cambridge, MA, 1987.

[25] P. R. Cohen and H. J. Levesque. Intention = choice + commitment. In *Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 2*, AAAI'87, pages 410–415. AAAI Press, 1987.

[26] A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In *Principles of Knowledge Representation and Reasoning. Proceedings of the second International Conference*, pages 473–484, San Mateo, 1991. Morgan Kaufmann.

[27] J. Hastings, W. Ceusters, M. Jensen, K. Mulligan, and B. Smith. Representing mental functioning: Ontologies for mental health and disease. In *Towards an Ontology of Mental Functioning (ICBO Workshop), Proceeedings of the Third International Conference on Biomedical Ontology*. 2012.

[28] A. Barton, W. Duncan, F. Toyoshima, and J. Ethier. First steps towards an ontology of belief. In L. Jansen, D. P. Radicioni, and D. Gromann, editors, *Proceedings of the Joint Ontology Workshops 2018 Episode IV: The South African Spring co-located with the 10th International Conference on Formal Ontology in Information Systems (FOIS 2018), Cape Town, South Africa, September 17-18, 2018*, volume 2205 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018.

[29] J. C. Nardi, R. de Almeida Falbo, J. P. A. Almeida, G. Guizzardi, L. F. Pires, M. van Sinderen, N. Guarino, and C. M. Fonseca. A commitment-based reference ontology for services. *Inf. Syst.*, 54:263–288, 2015.

[30] D. Porello and G. Guizzardi. Towards an ontological modelling of preference relations. In C. Ghidini, B. Magnini, A. Passerini, and P. Traverso, editors, *AI*IA 2018 - Advances in Artificial Intelligence - XVIIth International Conference of the Italian Association for Artificial Intelligence, Trento, Italy, November 20-23, 2018, Proceedings*, volume 11298 of *Lecture Notes in Computer Science*, pages 152–165. Springer, 2018.

[31] J. R. Searle. *Intentionality: An Essay in the Philosophy of Mind*. Cambridge University Press, 1983.

[32] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. Ontology library (wonder-web deliverable d18). *URL: http://www. loa-cnr. it/Papers D*, 18:36, 2003.

[33] R. Ferrario and A. Oltramari. Towards a computational ontology of mind. In A. C. Varzi and L. Vieu, editors, *Formal Ontology in Information Systems, Proceedigs of the Intl. Conf. FOIS 2004*, pages 287–297. IOS Press, 2004.

[34] C. Masolo, A. B. Benevides, and D. Porello. The interplay between models and observations. *Appl. Ontology*, 13(1):41–71, 2018.

[35] D. Porello, G. Guizzardi, T. P. Sales, G. C. M. Amaral, and N. Guarino. An ontological account of the action theory of economic exchanges (full paper). In B. Roelens, W. Laurier, G. Poels, and H. Weigand, editors, *Proceedings of 14th International Workshop on Value Modelling and Business Ontologies, Brussels, Belgium, January 16-17, 2020*, volume 2574 of *CEUR Workshop Proceedings*, pages 157–169. CEUR-WS.org, 2020.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200667

Foundations for an Ontology of Belief, Desire and Intention

Fumiaki TOYOSHIMA[a,1], Adrien BARTON[b,a,2] and Olivier GRENIER[c]
[a] *GRIIS, Université de Sherbrooke, Quebec, Canada*
[b] *Institut de Recherche en Informatique de Toulouse, CNRS, Toulouse, France*
[c] *CIRST, Université du Québec à Montréal, Quebec, Canada*

Abstract. Belief, desire, and intention are central notions in mentality and agency. We provide conceptual and formal foundations for an ontology of those mental entities. In this framework, beliefs and desires have a dual face: dispositional and occurrent. As distinct from beliefs and desires, intentions are dispositions to actions that emerge from a decision process in which occurrent beliefs and occurrent desires interact. We also discuss how our theory can be extended to some major philosophical accounts of desires, and cognitive biases such as wishful thinking.

Keywords. Belief, desire, intention, disposition, occurrent

1. Introduction

Belief, desire, and intention (which we will call "BDI entities") merit careful investigation because they are of paramount importance for an ontology of mental reality and agency. Inspired by Bratman's [1] philosophy, for instance, the BDI model of agency recognizes the primacy of the BDI entities in practical reasoning and rational actions [2]. It has been utilized in formal ontology of mind [3] and action [4]. Relatedly, the notion of goal and related entities (e.g. trying) have been formally explored [5], as it plays a vital role in the BDI model as well as in commonsense psychology [6]. To take another example, cognitive processes and representations have been investigated in the Mental Functioning Ontology (MF) [7] (which aims to serve as a mid-level ontology for mental functioning), and religious and spiritual beliefs are formalized along with the MF in the Web Ontology Language (OWL) [8]. Nonetheless, the BDI entities (*inter alia* desire and intention) would tend to be loosely characterized, especially so that their parent types are identified, but sometimes with no further detailed examination. Examples include belief as a "mental disposition" in the MF, the BDI entities as "mental moments" in UFO-C [9] (which is a module for social and intentional entities in the upper ontology the Unified Foundational Ontology), and the BDI entities as "mental states" in a Deontic Cognitive Event Ontology [10] (which provides an OWL support of representation and reasoning on complex cognitive information).

---

[1] Corresponding Author: Fumiaki Toyoshima, Université de Sherbrooke, 2500 Boulevard de l'Université, Sherbrooke, QC, J1K 2R1, Canada; E-mails: toyo.fumming@gmail.com; fumiakit@buffalo.edu. FT acknowledges financial support by the SPOR Canadian Data Platform (CIHR).

[2] Corresponding Author: Adrien Barton, 118 Route de Narbonne, F-31062 Toulouse Cedex 9, France; E-mail: adrien.barton@irit.fr.

In this paper we aim to pursue an ontological analysis of belief, desire, and intention. Along with previous works on belief [11], affordance [12-14], and directing actions [15], this paper is part of our project to build an ontology that covers the core categories and relations concerning agency, cognition, and actions. It will also give an impetus to develop and integrate existing BDI-based information systems and ontologies. For this purpose, we present previous formal-ontological works on dispositions (Section 2) and provide an ontological analysis of the BDI entities, actions, and their relationship along dispositional lines (Section 3). Then, we provide a core formalization of our ontology of the BDI entities and show its compatibility with some additional axioms or hypotheses (Section 4). We also discuss the relationships of our account with existing philosophical accounts of desires as well as with affordances and mental content (Section 5). Finally, we conclude the paper with some remarks on future work (Section 6).

We will use the following scenario named "(HEATER)" as a driving example, while taking for granted the notions of agent and action (see e.g. [4] for detailed discussion). An agent John is at his apartment. During a winter day, he feels cold and desires to get warm. There is a heater in his apartment. He believes that pushing a button of the heater will activate the heater. He also believes that the activated heater will heat up the apartment. He then intends to push the button of the heater and finally performs the action of pushing the button. Next to first-order logic, we will use OWL (using the Manchester Syntax [16]), which is a representation language for ontologies based on Description Logics. Terms for universals will be italicized and terms for particulars and relations will be written in bold.

2. Methodology: A dispositional approach

To describe mental mechanisms found in (HEATER) and other similar cases, we will employ an ontological distinction between continuants (e.g. objects), which persist through time; and occurrents (e.g. processes), which extend through time while having temporal parts and which typically have as participant some continuant. (Other occurrent-related terms "event" and "state" will not be used for simplicity.) We will also utilize previous formal-ontological works on dispositions [17-19], as dispositions are valuable for modeling of various entities (see [20] for a discussion about the relevance of dispositions to scientific ontologies). Note that preceding works on BDI entities (such as Bratman's [1]) do not necessarily see them as dispositional, but our dispositional approach aims for a "core basis" for an ontology of the BDI entities whose full development may require introducing other more specific categories and relations.

A disposition is a property that is linked (by a relation **realized_in**) to a realization, namely to a specific possible behavior of the bearer (such as an object) of the disposition. To be realized in a process, a disposition needs to be triggered (**has_trigger**) by some other process. Classical examples include fragility (the disposition to break when pressed with a force) and solubility (the disposition to dissolve when put in a solvent). Characteristically, dispositions may exist even if they are not realized or even triggered: for example, a glass is fragile even if it never breaks or even if it never undergoes any shock. We will also focus on "sure-fire dispositions" [17] whose realizations necessarily occur once the disposition has been triggered, as well as "single-track dispositions" [17] which have one kind of realizations and one kind of triggering processes. The term "disposition" will henceforth refer to a sure-fire and single-track disposition unless otherwise stated. (Note that our resulting theory of the BDI entities can extend to other kinds of dispositions such as "multi-track dispositions" [18,21] which have different kinds of realizations according to different kinds of triggers.)

There are two major frameworks to represent dispositions, called (ONLY) and (PARTHOOD) [18]. The (ONLY) framework (first developed by Röhl and Jansen [17]) characterizes a disposition by pointing to its classes of triggers and realizations: a disposition **d** whose class of realization is *R* and whose class of triggers is *T* would be formalized by: **d realized_in** only *R* and **d has_trigger** only *T*.

The (PARTHOOD) framework [18] considers that it is usually not possible to list the whole class of triggers and realizations of a disposition. For example, the specific shock on a glass would trigger its fragility, but so would the process that extends one millisecond earlier and after. Similarly, the specific breaking process of the glass would be a realization of its fragility, but so would the process that extends further to the glass pieces flying apart. Therefore, it introduces the class *TRmin*(**d**) of minimal triggers of a disposition *d* such that, informally speaking, their instances are the "smallest causal process" which exceeds the threshold value for causation. Formally, it is defined as the class of triggers of *d* for which no proper part is a trigger of *d*:

TRmin(**d**) EquivalentTo [(**trigger_of d** and not (**has_proper_part** o **trigger_of d**)]

where **trigger_of** is an inverse relation of **has_trigger**. It also introduces the class *Rmax*(**d**) of maximal realizations of *d*, which is the class of resulting whole causal chain of processes. Formally, it is defined as the class of realizations of *d* which are not proper parts of another realization of *d*:

Rmax(**d**) EquivalentTo [(**realization_of d** and not (**proper_part_of** o **realization_of d**)]

where **realization_of** is an inverse relation of **realized_in**. Two sure-fire single-track dispositions are then considered as identical if and only if they have the same instance of categorical basis, the same class of minimal triggers, and the same class of maximal realizations. Unlike the (ONLY) model, this (PARTHOOD) model of dispositions avoids "disposition multiplicativism" (that is, the excessive arbitrary proliferation of dispositions) [18].

Finally, a recent mereological theory of dispositions [19] specifies several kinds of parthood relations between dispositions. Among them, the **mod-part_of** relation formalizes several possible pathways, or *mod*es, of realizations of dispositions: e.g. the ferromagnetic disposition of this magnet having two mod-parts, i.e. its disposition to attract another magnet when facing an unlike pole and its disposition to repulse a magnet when facing a like pole.

3. An ontological account of belief, desire, and intention

3.1. Belief

Let us begin by discussing belief.[3] In (HEATER), John believes that pushing a button of a heater will activate the heater. The first thing to note is that John's belief exists even

[3] Note that the term "belief" is polysemous. One may sometimes use the term "John's belief" to refer to the truth-evaluable content of John's belief: e.g. "John and James both have the very same false belief that the earth is flat". In philosophy, this content is generally taken to be a proposition [22]. The controversial nature of mental contents or propositions is outside the scope of this paper (but see Section 5.5). We will instead focus

when he is not consciously thinking about it, e.g. when he is sleeping. John's belief exists in virtue of some mentally relevant feature of his cognitive (neutral) system. This consideration would suggest a dispositional characterization of belief, since dispositions are properties that are physically grounded and that do not always need to be realized, i.e. activated. The next question to be addressed is how belief (as a disposition) can be realized. A naive idea [23] is that beliefs are dispositions to perform certain kinds of actions: that is, dispositions to behave in a certain way. John's belief that pushing a button of a heater will activate the heater is realized when he performs the action of activating the heater by pushing its button, for instance.

This approach does not seem to account for the nature of belief, however. Intuitively, John can have a belief to that effect even if he is totally paralytic and he is not able to press a button of the heater. One may counter that beliefs are rather dispositions to act if further conditionalized: e.g. John's belief is a disposition to activate the heater if he is physically capable of pressing its button. This proposal would nevertheless only capture the practical (behavioral) dimension of belief (how belief is related to action), but not how belief is connected to the purely theoretical (cognitive) attitude of taking something to be the case. Imagine an omniscient spirit with no power of action at all: he would have many beliefs about the world, but no disposition to act [11].

We therefore hold that a dispositional belief is not realized by physically performing actions, but by some mental process that we call "occurrent belief": namely, the cognitive process of taking something to be the case [11]. In (HEATER), John may believe that pushing on a button of a heater will activate the heater, but this (dispositional) belief is not continuously realized (or, activated) in his mind, as when the weather is hot. In a cold day, on the other hand, he deliberates whether he should press a button of the heater and his dispositional belief **dBEL** is realized, at time t_0, in a process **oBEL** of him taking the pressing of the button of the heater to be responsible for the activation of the heater (formally: **dBEL realized_in oBEL**) – a process that will, as we will see, play a causal role in his decision process. In general, an agent may have a dispositional belief even when asleep or unconscious. Briefly, we suggest the following terms and their characterizations:

- Dispositional belief: A disposition that can be realized in an occurrent belief.
- Occurrent belief: A mental process of taking something to be the case.[4]

One may suspect that those characterizations of dispositional belief and occurrent belief are circular, as "taking something to be the case" is usually taken to be synonymous with "believing". But those characterizations can be taken as *elucidations* of the terms in question. As a matter of fact, upper-level entity terms (e.g. "continuant") are hardly definable without circularity and they can be at best elucidated together with the examples to illustrate the entities to which they can apply (cf. [26, p. 89]). Our elucidations of the term "belief" would thus serve to classify two different entities to which it refers to: a belief as a disposition and a belief as a process. This could be likened to a dispositional account of diseases provided in the Ontology for General Medical

on John's belief as an entity in John's mind. That is, even if John and James both believe that the Earth is flat, they do not have the same mental state (in a non-technical sense of the term) of belief. Note that this scope for belief will apply, *mutatis mutandis*, to our later discussion on desire and intention.

[4] Our dual account of belief may have a historical root [24]. For another alternative, one may attempt to posit a single kind of belief, for example, by using the notion of "process as a continuant-like occurrent" [25]. Note that these two different approaches to belief can apply for desire, which we will analyze below.

Science [27]: a disease (e.g. epilepsy) is a subtype of disposition that is manifested by undergoing “pathological processes” (e.g. epileptic seizures).

3.2. Desire

We move onto a discussion on desire. Intuitively, there seems to be an intimate connection between desires and behavioral dispositions because the former are closely related to motivation, which is in turn related to the latter [28]. In (HEATER), John is *prima facie* motivated to do things that he believes will result in his warmth. It is however implausible to think that desire is just a disposition towards an action because desires can be active in the agent’s mind without exerting a causal effect on its behavior [29]. John’s desire to be warm can be active on a cold day, but still not affect his behavior if it is countered by a greatest desire to save energy.

Based on this observation, we adopt a distinction similar to the one drawn by Schroeder [29] between “standing desires” and “occurrent desires”, the former being *potentially* active during its existence, and the latter being *actually* and constantly active during its existence. This dual view of desire leads reasonably to the following “desire counterparts” of dispositional and occurrent beliefs [30]:

- Dispositional desire: A disposition that can be realized in an occurrent desire.
- Occurrent desire: A mental process of wanting something to be the case.

In (HEATER), John’s dispositional desire $\mathbf{d_{DEL}}$ to get warm is realized in his occurrent desire $\mathbf{o_{DEL}}$ to get warm (formally: $\mathbf{d_{DEL}}$ **realized_in** $\mathbf{o_{DEL}}$). It should be emphasized that, with the same set of desires, an agent could act in multiple ways depending on her beliefs. For instance, John could have decided to put on a sweater, in virtue of his belief that putting on a sweater will get him warmer. To understand John’s action of pushing a button of the heater therefore requires considering not only his occurrent desire $\mathbf{o_{DEL}}$ but also its interaction with $\mathbf{o_{BEL}}$ and his other occurrent belief that the activated heater will warm him up effectively. We will detail this point later.

3.3. Intentions

3.3.1. Intentions and dispositions

We will use the word “intention” rather than “intent”, as the former is not necessarily the output of a deliberative decision process. As a matter of fact, an intention can also result from a heuristic (intuitive, instinctive) decision process [31,32]. I can have a disposition to act that is due to an intuitive decision process, but this disposition might still be blocked, although maybe with more difficulty than dispositions to act that result from a deliberative decision process.

Intentions behave more like continuants than like occurrents: John’s intention to read a book in May can be wholly present at different times in April, and wax and wane as time passes. Moreover, intentions have a dispositional character in the sense that my intention to do *A* typically leads to me doing *A*, or can be blocked. For example, I formed the intention to go for lunch but suddenly I remember that I have to write this email, so my intention is not realized. Or more radically, I decided to stand up but suddenly I’m paralyzed, so I cannot. Therefore, we categorize intentions as (mental) dispositions. In (HEATER), John’s (dispositional) intention $\mathbf{d_{INT}}$ to activate the heater is realized in his action $\mathbf{o_{ACT}}$ of pushing the button of the heater (formally: $\mathbf{d_{INT}}$ **realized_in** $\mathbf{o_{ACT}}$).

Of course, dispositions can exist without intentions: inanimate objects do have plenty of dispositions. For example, John also has a disposition **d** to push the button of the heater that is triggered by a heavy object pushing his finger down on the button, but this disposition is purely mechanical, and has nothing to do with his intention to push the button. The dispositions $\mathbf{d_{INT}}$ and **d** do not have the same kinds of triggers: $\mathbf{d_{INT}}$ is not triggered by a heavy object pushing down John's finger; hence those are two different dispositions, by Barton et al.'s [18] identity criterion for dispositions.

3.3.2. In favor of the non-reductivity of intentions to beliefs and/or desires

Our proposal presupposes that intentions are *bona fide* entities distinct from both beliefs and desires, in keeping with a vital theoretical role of intentions in commonsense psychology [6] and the "adequatist" principle of ontology building according to which "the entities in any given domain should be taken seriously on their own terms" [26, p. 46]. This non-reductive view of intentions has been nonetheless criticized. That is, some philosophical theories have identified intentions with beliefs, whereas others have identified intentions with desires, or with desire-belief compounds.

For instance, strong intention cognitivism maintains that intending to *V* consists in believing that one will *V*. It is primarily motivated by the linguistic observation that canonical sentences expressing intentions, such as "I am going to *V*" and "I will *V*" are also used to express beliefs [33]. As Levy [34] says, however, this linguistic argument is not convincing enough to establish an intimate connection between intentions and beliefs, since declarative sentences are typically used to express non-belief attitudes: e.g. "I'd like to know what time it is."

To take another example, a "belief-desire account" of intentions has been popular since the former Davidson's [35] theory of intentional actions. According to Mulder's [36] formulation of such account, an agent A intends to φ iff A desires, all things considered, φ. For example, Shihababu [37] argues that intentions are reducible to desires because desires can "motivate action when combined with an appropriate means-end belief". Alternatively, intentions could be identified with desire-belief compounds.

Indeed, there seems to be a close proximity between desires and intentions. Seen linguistically, for instance: "beliefs are like declarative sentences, which are satisfied (made true) by whether the world as it is conforms to them. But desires are like imperative sentences, which are satisfied (fulfilled) by changes in the world bringing the world into conformity with them" [29]. Intentions, like desires, are more like imperative sentences than declarative sentences: they are satisfied by changes in the world bringing it into conformity with them.

The crucial notion of practical reasoning nevertheless shows the distinction between desires and intentions [1,38]. However, as explained by the later Davidson [38], desires attach to actions less directly than intentions do. For example, John may have a desire to be warm, but this desire may be trumped by a stronger desire to spare energy. On the other hand, a (well-formed) intention to activate the heater sees as settled this issue of what action to perform so as to satisfy a desire to get warm. Mulder [36] argues that intentions need to be posited in order to capture the notion of practical reasoning that would evaluate and hierarchize the various beliefs and desires, and that issues in "a *practical judgment* about what is to be done". Our theory accounts for this non-reductive nature of intentions: occurrent desires and occurrent beliefs are parts of a decision process of practical reasoning, which may lead to the formation of an intention.

4. Formal ontological foundations of belief, desire, and intention

4.1. Core formalization

We will now propose some axioms in OWL. In the (ONLY) framework of disposition [17] presented in Section 2, we would say that a dispositional belief has as realization only occurrent beliefs, and that a dispositional desire has as realization only occurrent desires:

(Bel_O) *Dispositional belief* SubClassOf (**realized_in** only *Occurrent belief*)
(Des_O) *Dispositional desire* SubClassOf (**realized_in** only *Occurrent desire*)

Conversely, in (PARTHOOD), every maximal realization of a dispositional belief **bel** has as part some occurrent belief: intuitively, an occurrent belief is the "minimal part" that is to be found in every realization of a dispositional belief (but the realization might be larger, if, for example, the occurrent belief causes on its own other cognitive processes). That is:

(Bel_P) *Rmax*(**bel**) SubClassOf (**has_part** some *Occurrent belief*)

(note that we have to write one such axiom for every instance **bel** of *Dispositional belief*, which is a shortcoming for the OWL representation of the framework (PARTHOOD))

Similarly, we can state that every maximal realization of a dispositional desire **des** has as part some occurrent desire. That is:

(Des_P) *Rmax*(**des**) SubClassOf (**has_part** some *Occurrent desire*)

Let us now turn to decision processes and intentions. A decision process is a process that integrates some belief(s) and some desire(s) to yield an intention. We formalize it as stating that a decision process has as parts some occurrent belief(s) and occurrent desire(s):

(Dec) *Decision process* SubClassOf [(**has_part** some *Occurrent_belief*) and (**has_part** some *Occurrent_desire*)]

Every intention is the result of some decision process. This can be formalized using the **specified_output_of** relation from the Ontology for Biomedical Investigation (OBI) [39] as follows:

(Int-Dec) *Intention* SubClassOf (**specified_output_of** some *Decision_process*)

(note that the converse does not hold: a decision process may not lead to any intention, if, for example, the agent is still hesitant at the end of the decision process).

Moreover, an intention is realized in an action. In the (ONLY) framework, this would be written as:

(Int_O) *Intention* SubClassOf (**realized_in** only *Action*)

In the (PARTHOOD) framework, for every instance **i** of *Intention*, we would have:

(Int_P) *Rmax*(**i**) SubClassOf (**has_part** some *Action*)

4.2. The dynamic structure of the decision process

In the former account, we did not enter into the details of how the decision process is structured. In particular, we did not specify how beliefs and desires could interact. One could imagine, for example, that an occurrent belief (e.g. that I'm allergic to apples) would trigger a dispositional desire (not to get an allergic crisis). In some cases at least, however, an occurrent desire could be triggered by something else than an occurrent belief (I can actively desire not to get an allergic crisis without having any active belief that I have an allergy).

Also, suppose for example that at t_1, John has both a desire to eat an apple and a desire to eat a peach. He deliberates whether he will eat an apple or a peach, and this decision process has as parts an occurrent desire to eat an apple, and an occurrent desire to eat a peach. How the weighing of desire takes place is a further question that exceeds this article.

4.3. Compatibility with additional axioms or hypotheses

We provided above the most basic conceptualization and formalization of the BDI entities. Our theory is compatible with some additional axioms or hypotheses, two of which we will discuss below: (i) some occurrent beliefs triggering some dispositional desires and (ii) intentions being always triggered by occurrent beliefs.

4.3.1. Occurrent belief triggering a dispositional desire

We did not delve into the details of the structure of a decision process, in particular exactly how beliefs and desires therein interact. We leave the determination of those interactions open, as they rely on complex psychological, neurological and epistemological questions that are out of scope of this article. Still, it is a plausible claim that one or more occurrent beliefs may trigger a dispositional desire. Suppose that John deliberates at t_0 whether he should press on the heater switch. His dispositional belief $\mathbf{d_{BEL}}$ will then be realized in his occurrent belief $\mathbf{o_{BEL}}$ that if he presses the switch, the temperature will increase. He therefore needs to consider whether he wants the temperature to increase. This will trigger his dispositional desire to be warm $\mathbf{d_{DES}}$, that will be realized in an occurrent desire to be warm $\mathbf{o_{DES}}$. In such a case, we would have a dispositional desire triggered by an occurrent belief: $\mathbf{d_{DES}}$ **has_trigger** $\mathbf{o_{BEL}}$.

4.3.2. Occurrent belief triggering an intention

To explain the other claim that an occurrent belief triggers an intention, let us consider Bosse et al.'s [40] following "semi-formal" (in their terminology) explanation of the BDI model in developing a recursive BDI-based agent model for the theory of mind:

- At any point in time: If a desire is present and a "belief in reason" is present, then an intention for an action will occur. (The term "reason" therein means "the (rational) choice of an action that is reasonable to fulfill the given desire".)
- At any point in time: If an intention for an action is present and a "belief in opportunity" is present, then the action will be performed. (The term "belief in

opportunity" therein refers to "the belief that certain circumstances in the world are fulfilled such that the opportunity to do the action is there".)

Not surprisingly, their first statement fits well with our view of a decision process as a process in which occurrent beliefs and occurrent desires closely interact in some way. Being possibly motivated by their recursive considerations, by contrast, their second claim can be formalized in our dispositional account of intention in such a way that an intention (disposition) would be stimulated by an occurrent belief (trigger), resulting in some action (realization). This is compatible with our general notion of triggers of dispositions and it can be formalized as follows:

Given (ONLY): *Intention* SubClassOf (**has_trigger** only *Occurrent belief*)

Given (PARTHOOD): For every instance **i** of *Intention*, *Tmin*(**i**) SubClassOf *Occurrent belief*

To illustrate this, suppose that a local singing contest takes place every month and Mary receives a regular voice training to win the competition while intending to participate in it when her voice will be trained to a certain level. When she believes at a certain time that she sings well enough to win the contest, she will perform the action of participating in the event of the month. In our ontological framework, Mary's occurrent belief in her developed singing skills triggers her intention to participate in a singing competition.

5. Discussion

5.1. Comparison with philosophical accounts of desires

It is well worth comparing our dispositional model of the BDI entities with some philosophical accounts of desires. Let us look at three major theories of desires in contemporary philosophy [29]. Note that we remain neutral on the object of desire that is designated by '*p*' in the following (but see Section 5.5 for a brief discussion):

- **Action-based theory:** For an organism to desire *p* is for the organism to be disposed to take whatever actions it believes are likely to bring about *p*.
- **Pleasure-based theory:** For an organism to desire *p* is for the organism to be disposed to take pleasure in it seeming that *p*, and to take displeasure in it seeming that not-*p*.
- **Good-based theory:** For an organism to desire *p* is for the organism to believe that *p* is good.

Those theories embrace some kind of relationship between desires and dispositions. We will consider how each theory of desires can be formally characterized in OWL within our ontological framework for the BDI entities.

5.1.1. The action-based theory of desire

At first sight, the action-based theory of desire would dovetail with a deflationary view of intentions since it states that desires are something that disposes an agent towards actions, in contradistinction with our non-reductive approach to intentions. One possible interpretation of this account reduces intention to desires, and considers that an intention is simply a desire which motivates an agent to act in a context where the agent has an appropriate "means-end belief": e.g. John's desire to warm up in the context where he has a belief that switching on a heater is the best way to achieve warmth [37]. The following axiom would ensue:

Intention SubClassOf *Dispositional desire*

In that case, we could keep (Int_O)/(Int_P) but should reject (Des_O)/(Des_P), since dispositional desires would be realized in actions, rather than in occurrent desires.

Another possible construal is that an intention is a desire-belief compound. Thus, an intention could be described as the mereological sum of a desire and a means-end belief:

Intention SubClassOf (**has_part** some *Dispositional desire*) and (**has_part** some *Dispositional belief*)

One question would be what kind of parthood (see [19] for three kinds of disposition-parthood) is involved in the axiom above. Answering this would be important to determine which of the former axioms from Section 4.1 would be accepted.

Let us now consider a more specific action-based theory. Ashwell [28,41] accounts for an agent's desire as a "second-order disposition" to have a particular behavioral disposition, such that which behavioral disposition is chosen is determined by her beliefs as to how she can bring about something desirable. Let us illustrate it with a variant of (HEATER). In Ashwell's framework, John's desire to get warm could be seen as a second-order disposition $\mathbf{d^{2nd}_{DES}}$ to acquire multiple behavioral dispositions, such as a disposition to activate a heater and a disposition to wear a sweater.

We will consider two ways to specify Ashwell's view, depending on how to interpret the terms "first-order" and "second-order" dispositions. The first one would be that a second-order disposition is a disposition that has two modes, and is therefore composed, in the "mod-parthood" sense of the term [19] mentioned earlier, by two first-order dispositions that each have a single causal pathway. Let $\mathbf{d_{activate}}$ (resp. $\mathbf{d_{wear}}$) be John's mental disposition realized in him activating the heater (resp. wearing his sweater) to get warm. We can think of the desire $\mathbf{d^{2nd}_{DES}}$ as a multi-track disposition [18,21] to activate the heater and to wear the sweater, whose realized pathway will depend on John's belief about which is the better way of warming him up. That is: both $\mathbf{d_{activate}}$ **mod-part_of** $\mathbf{d^{2nd}_{DES}}$ and $\mathbf{d_{wear}}$ **mod-part_of** $\mathbf{d^{2nd}_{DES}}$ hold. Following Section 4.3.1, $\mathbf{d_{activate}}$ (and thus $\mathbf{d^{2nd}_{DES}}$) would be triggered by John's occurrent belief that activating the heater is the best way to get warm; whereas $\mathbf{d_{wear}}$ (and thus $\mathbf{d^{2nd}_{DES}}$ here too) would be triggered by his occurrent belief that wearing a sweater is the best way to get warm. In such a view, a desire would be realized by an action, and thus it should reject (Des_O)/(Des_P).

According to the second interpretation of Ashwell's proposal, second-order dispositions are dispositions that are realized in a process leading to the formation of a first-order disposition. This would be in line with our own formalization presented in Section 4.1, which considers a desire as a disposition whose realization is part of a process that leads to the formation of an intention, which is a disposition to act. In this view of second-order dispositions, the intention is a first-order disposition to act, whereas desires (and beliefs) are second-order dispositions to act.

5.1.2. The pleasure-based theory of desire

Next, while leaving aside the psychological nature of (dis)pleasure, we can understand the pleasure-based theory as claiming that a dispositional desire (for *p*) is a disposition with two causal pathways: one to take pleasure in some circumstances (namely, seeming that *p*), and one to take displeasure in some other circumstances (namely, seeming that not-*p*). Such a disposition with two modes is formalized, as explained above, with the notion of "mod-parthood". Therefore, we could formalize the pleasure-based theory of desire based on two classes *Disposition to take pleasure* and *Disposition to take displeasure* and the **has_mod-part** relation (which we define as the inverse of **mod-part_of**), hence the following axiom:

> *Dispositional desire* SubClassOf (**has_mod-part** some *Disposition to take pleasure*) and (**has_mod-part** some *Disposition to take displeasure*)

In this framework, dispositional desires are realized by processes of taking pleasure or taking displeasure, rather than by occurrent desires that can be part of a decision process, as in our account; therefore, (Des_O)/(Des_P) should be rejected in such a framework.

5.1.3. The good-based theory of desire

A good-based theory of desire would typically consider that a dispositional desire for *p* is a dispositional belief that *p* is good, which leads to the following axiom:

> *Dispositional desire* SubClassOf *Dispositional belief*

Similarly, *Occurrent desire* might be seen as a subclass of *Occurrent belief*.

A slightly different construal would be that a desire of *p* would be a disposition to get a belief that *p* is good. Then, a dispositional desire for *p* is realized in an occurrent desire that has as output a dispositional belief that *p* is good. This can be formalized using the OBI:**has_specified_output** relation [39] and the (ONLY) model of dispositions as follows:

> *Dispositional desire* SubClassOf (**realized_in** only (*Occurrent desire* and **has_specified_output** some *Dispositional Belief*))

5.2. Belief-forming biases

Consider how our model could account for additional mental phenomena, such as cognitive biases. Let us take the example of wishful thinking [42], which we will understand as humans' tendency to believe what they desire. Suppose that John desires his lucky number to be selected at the lottery (manifested by his occurrent desire $\mathbf{o}^{\mathbf{lucky}}_{\mathbf{DES}}$) and therefore believes that his lucky number will be selected at the lottery (hence his dispositional belief $\mathbf{d}^{\mathbf{lucky}}_{\mathbf{BEL}}$). We would not have $\mathbf{o}^{\mathbf{lucky}}_{\mathbf{DES}}$ triggering $\mathbf{d}^{\mathbf{lucky}}_{\mathbf{BEL}}$: indeed, $\mathbf{d}^{\mathbf{lucky}}_{\mathbf{BEL}}$ does not exist before $\mathbf{o}^{\mathbf{lucky}}_{\mathbf{DES}}$, and thus cannot be triggered by it. Instead, we formalize the human bias of wishful thinking as another mental disposition $\mathbf{d}_{\mathbf{WISH}}$ in Jonh's mind, a disposition to create wishful thinking beliefs on the basis of his beliefs. $\mathbf{o}^{\mathbf{lucky}}_{\mathbf{DES}}$ triggers $\mathbf{d}_{\mathbf{WISH}}$, which is then realized in a mental process of wishful thinking (**o**$_{WISH}$) that creates **d**$^{lucky}_{BEL}$ (note that we do not claim that *all* occurrent beliefs would lead to the formation of a corresponding desire).

To represent formally **d**$_{WISH}$, we might use the OBI relation **has_specified_output** as follows:

d$_{WISH}$ **has_trigger o**$^{lucky}_{DES}$
d$_{WISH}$ **realized_in** some (*Mental process* **and has_specified_output d**$^{lucky}_{BEL}$)

To generalize this, in the (ONLY) framework, we might introduce a new class *Dispositional_wishful_thinking* of cognitive biases as follows:

Dispositional_wishful_thinking SubClassOf (**has_trigger** only *Occurrent_desire*)
Dispositional_wishful_thinking SubClassOf [**realized_in** only (*Mental process* and **has_specified_output** some *Dispositional Belief*)]

5.3. Beliefs and moral motivation

Moral motivation is the motivation that is linked to one's moral (and thus normative) judgments. Rosati [43] describes it more systemically: When an agent judges that it would be morally right to perform an action, the agent is ordinarily motivated to perform the action. There is a philosophical debate over moral motivation between Humeanism and anti-Humeanism [43]. Humeanism says that the belief that acting in some way will lead to the achievement of a goal is not sufficient for motivation: a desire to achieve the goal is required as well. Anti-Humeanism counters that some beliefs, more specifically moral beliefs, can motivate an action on their own.

Let us identify here motivation with intention (although this would require further investigations). As said above, our model of the BDI entities states that intentions emerge from decision processes, which are characterized as processes in which not only beliefs but also desires interact in some way. Thus, anti-Humeanism would conflict with the combination of our axioms (Dec) and (Int-Dec), which imply together that all intentions to act result from a decision process that includes both (occurrent) beliefs and desires. Therefore, our framework is Humean. Note however that it would be easy to switch to a framework that would be agnostic concerning the issue of Humeanism by accepting, instead of (Dec), the following weaker axiom:

Decision process SubClassOf (**has_part** some *Occurrent_belief*)

5.4. Linking intentions with affordances

Gibson [44] defines affordances of the environment as "what it *offers* the animal, what it *provides* or *furnishes*, either for good or ill" (p. 119). Canonical examples include the character of stairs to be potentially climbed and the character of gaps to potentially hide agents. A theory of affordances offers a first foundation upon which agents and inanimate objects (e.g. tools) are distinguished and identified [45]. According to Heras-Escribano's [46] affordance-based approach to agency, an agent's intentional actions are characterized by the agent's possibility to choose among different affordances.

Proposed by Turvey [47], a dispositional theory of affordances has been philosophically furthered by Heras-Escribano [46]. In formal ontology, it has been theoretically investigated by some of us [12-14], in a way that could also be used as a basis for other accounts of affordances (e.g. [48]). For instance, the affordance of the stairs is their disposition to support people as they move upward (or downward) when the stairs are of a suitable proportion for their leg length. Those considerations could lead us to link intentions with affordances: for instance, Mary's intention to climb up the stairs is realized only when so is the affordance of the stairs to support her as she moves upward.

5.5. The content of the BDI entities and its aboutness

We make some brief observations about the content of the BDI entities that are to be taken seriously along with this line of inquiry. As we said in Section 3.1, the content of mental attitudes has been examined in philosophy under the name of "proposition" [22]. While the content of belief is traditionally acknowledged to be a proposition (as a "truth-bearer"), it is highly contentious whether desire is a "propositional attitude" as well [49]. When John desires to get warm, for instance, it is not obvious whether he desires a state of affairs [50] of him being warm (which is standardly taken to be propositionally structured) or he simply desires for warmth as a thing with no propositional structure. This discussion can be extended to the content of intention and also of many other mental attitudes such as love and fear [51].

It is important to remark that some existing ontologies of mental entities [3,7] share the idea that the notion of *aboutness* plays a vital role in connecting the BDI entities with their content. For instance, a previous work [11] on belief provides a preliminary formalization of the content of belief using the notion of information content entity [52] which is, by definition, about something. For another example, Biccheri et al. [53] complement a BDI approach to mental states (in their terminology) with their dual account of aboutness: a mental state is about an "intentional content" and also about an "intentional object". Aboutness still remains elusive in formal ontology, but it may be elucidated in terms of semiotics [54] which explicates meanings and representations in terms of the triad of a sign, an object and an interpreter.

6. Conclusion and future work

We examined beliefs, desires, and intentions conceptually and formally. Beliefs and desires are Janus-faced: dispositional beliefs (resp. desires) can be realized in occurrent beliefs (resp. desires). As distinct from beliefs and desires (or their compounds), intentions are dispositions to actions that emerge from a decision process in which occurrent beliefs and occurrent desires interact. This account can be linked to an action-based theory of desire such as Ashwell's [28,41], in which desires are interpreted as second-order dispositions to action, mediated by intentions (whereas pleasure-based or good-based accounts of desires would require some changes in the formalization). Our account can also mesh with some BDI model of agency such as Bosse et al.'s [40], and can represent some cognitive biases such as wishful thinking.

There are several ways in which we will be able to investigate agency, cognition, and actions. First, our discussion on desires needs to extend to undesirability: e.g. severe adverse effects from a new drug would be undesirable to me. This work will contribute to e.g. Grenier's [30] ontological analysis of risk as a disposition whose realizations would be undesirable for an agent. Second, it will require exploration exactly how our account of intentions can formalize e.g. what Searle [55] calls "intention in action": an intention that is not formed in advanced of the action and causes it by representing its condition of satisfaction "on the fly". Third, the notion of agency is underpinned not only by affordances but also by image schemas: mental patterns that are extracted from the sensory and motile experiences [45]. A previous affordance-based ontological approach [14] to image schemas will help to consider the relationship between the BDI entities and image schemas. Such considerations are also linked to the notion of instrumental desire: e.g. Mary's desire to raise her feet because she has a desire to climb up the stairs. Fourth, cognitive and neuroscientific BDI-related studies can be integrated with our proposal e.g. by specifying the "categorical bases" [18] of dispositional belief and desire: for instance, John's dispositional belief has as its categorical basis some neural structure of his brain, just as fragility of this glass has as its categorical basis some molecular structure of the glass. Fifth and finally, it will be worthwhile to connect our ontology of the BDI entities with action-directing informational entities such as recipes [15]. This line of inquiry could yield e.g. further development of an ontology of drug prescriptions [56] in representing the processes of drug-taking that drug prescriptions can direct.

References

[1] Bratman ME. Intention, plans, and practical reason. Cambridge: Harvard University Press; 1987. 208 p.
[2] Wooldridge MJ. Reasoning about rational agents. Cambridge: MIT Press; 2000. 241 p.
[3] Ferrario R, Oltramari A. Towards a computational ontology of mind. In: Proceedings of FOIS2004. Amsterdam: IOS Press; c2004. p. 287-97.
[4] Trypuz R. Formal ontology of action: a unifying approach [dissertation]. University of Trento; 2007. 273 p. Available from: https://www.researchgate.net/publication/233990262
[5] Hobbs J, Gordon A. Goals in a formal theory of commonsense psychology. In: Proceedings of FOIS2010. Amsterdam: IOS Press; c2010. p. 59-72
[6] Kashima Y, McKintyre A, Clifford P. The category of the mind: folk psychology of belief, desire, and intention. Asian J. Soc. Psychol. 1998 Dec;1(3):289-313.
[7] Hastings J, Smith B, Ceusters W, Jensen M, Mulligan, K. Representing mental functioning: ontologies for mental health and disease. ICBO2012. c2012. p. 1-5.
[8] Schulz S, Jansen L. Towards an ontology of religious and spiritual belief. In: Proceedings of FOIS2018. Amsterdam: IOS Press; c2018. p. 235-60.
[9] Guizzardi G, Falbo RD, Guizzardi R. Grounding software domain ontologies in the Unified Foundational Ontology (UFO): The case of the ODE software process ontology. In: Proceedings of CIbSE2008. CIbSE; c2008. p. 127-40.
[10] Vacura M. Modeling artificial agents' actions in context - a deontic cognitive event ontology. Appl. Ontol. 2020; pre-pressed:1-35.
[11] Barton A, Duncan W, Toyoshima F, Ethier JF. First steps towards an ontology of belief. In: Proceedings of JOWO2018. CEUR Workshop Proceedings, vol. 2205; c2018. p. 1-5.
[12] Toyoshima F. Modeling affordances with dispositions. In: Proceedings of JOWO2018. CEUR Workshop Proceedings, vol. 2205; c2018. p. 1-6.
[13] Toyoshima F, Barton A. A formal representation of affordances as reciprocal dispositions. In: Proceedings of TriCoLore2018. CEUR Workshop Proceedings, vol. 2347; c2018. p. 1-14.
[14] Toyoshima F, Barton A. Linking image schemas with affordances: an ontological approach. In: Proceedings of JOWO2019. CEUR Workshop Proceedings, vol. 2518; c2019. p. 1-10.
[15] Barton A, Vieu L, Ethier JF. Directing actions. In: Proceedings of JOWO2019. CEUR Workshop Proceedings, vol. 2518; c2019. p. 1-10.
[16] Horridge M. Patel-Schneider PF. OWL 2 web ontology language Manchester syntax. 2nd ed. W3C Working Group Note 11 December 2012.
[17] Röhl J, Jansen. L. Representing dispositions. J. Biomed. Semant. 2011 Aug;2(Suppl 4):S4.
[18] Barton A, Grenier O, Jansen L, Ethier JF. The identity of dispositions. In: Proceedings of FOIS2018. Amsterdam: IOS Press; c2018. p. 113-25.
[19] Barton A, Jansen L, Ethier JF. A taxonomy of disposition-parthood. In: Proceedings of JOWO2017. CEUR Workshop Proceedings, vol. 2050; c2017. p. 1-10.
[20] Toyoshima F. Natural necessity: an introductory guide for ontologists. Appl. Ontol. 2020 Feb;15(1):61-89.

[21] Barton A, Jansen L. A modelling pattern for multi-track dispositions for life-science ontologies. In: Proceedings of ODLS2016. CEUR Workshop Proceedings, vol. 2692; c2016. p. H1-2.
[22] McGrath M, Frank D. Propositions. In: Zalta EN, editor. The Stanford encyclopedia of philosophy. Spring 2018 ed. Available from: https://plato.stanford.edu/archives/spr2018/entries/propositions/
[23] Schwitzgebel, E. Belief. In: Zalta EN, editor. The Stanford encyclopedia of philosophy. Fall 2019 ed. Available from: https://plato.stanford.edu/archives/fall2019/entries/belief/
[24] de Haas FAJ. Potentiality in Aristotle's psychology and ethics. In Engelhard K, Quante M, editors. Handbook of potentiality. The Netherlands: Springer; 2018. p. 71-91.
[25] Galton A, Mizoguchi R. The water falls but the waterfall does not fall: new perspectives on objects, processes and events. Appl. Ontol. 2009;4(2):71-107.
[26] Arp R, Smith B, Spear AD. Building ontologies with Basic Formal Ontology. MIT Press; 2015. 248 p.
[27] Scheuermann RH, Ceusters W, Smith B. Toward an ontological treatment of disease and diagnosis. In: Proceedings of the 2009 AMIA Summit on Translational Bioinformatics; c2009. p. 116-20.
[28] Ashwell L. The metaphysics of desire and dispositions. Philos. Compass. 2014 Jul;9(7):469-77.
[29] Schroeder T. Desire. In: Zalta EN, editor. The Stanford encyclopedia of philosophy. Summer 2017 ed. Available from: https://plato.stanford.edu/archives/sum2017/entries/desire/
[30] Grenier O. Ontologies du risque [master's thesis]. University of Sherbrooke; 2019. 150 p. Available from: http://hdl.handle.net/11143/15991
[31] Kahneman D, Tversky A. Prospect theory: an analysis of decision under risk. Econometrica. 1979 Mar;47(2):263-91.
[32] Gigerenzer G, Gaissmaier W. Heuristic decision making. Annu. Rev. Psychol. 2011 Jan;62:451-82.
[33] Setiya K. Intention. In: Zalta EN, editor. The Stanford encyclopedia of philosophy. Fall 2018 ed. Available from: https://plato.stanford.edu/archives/fall2018/entries/intention/
[34] Levy Y. Why cognitivism? Can. J. Philos. 2018 Jun;48(2):223-44.
[35] Davidson D. Actions, reasons, and causes. J. Philos. 1963 Nov;60(23):685-700.
[36] Mulder JM. Why intentions? Ratio. 2018 Oct;31(S1):51-64.
[37] Sinhababu N. The desire-belief account of intention explains everything. Noûs. 2012 Jun;47(4):680-96.
[38] Davidson D. How is weakness of the will possible? In: Feinberg J, editor. Moral concepts. London: Oxford University Press; 1969. p. 93-113.
[39] Bandrowski A. et al. The ontology for biomedical investigations. PLoS One. 2016 Apr;11(4):e0154556.
[40] Bosse T, Memon ZA, Treur J. A recursive BDI-agent model for theory of mind and its applications. Appl. Artif. Intell. 2011 Jan;25(1):1-44.
[41] Ashwell L. Conflicts of desire: dispositions and the metaphysics of mind. In: Jacobs JD, editor. Causal Powers. Oxford: Oxford University Press; 2017. p. 167-76.
[42] Reber AS, Reber ES, Allen R. The penguin dictionary of psychology. 4th ed. Penguin Books; 2009. p. 928.
[43] Rosati CS. Moral motivation. In: Zalta EN, editor. The Stanford encyclopedia of philosophy. Winter 2016 ed. Available from: https://plato.stanford.edu/archives/win2016/entries/moral-motivation/
[44] Gibson JJ. The ecological approach to visual perception. Houghton Mifflin; 1979. 332 p.
[45] Kutz O, Troquard N, Hedblom MM, Porello D. The mouse and the ball: towards a cognitively-based and ontologically-grounded logic of agency. In: Proceedings of FOIS2018. Amsterdam: IOS Press; c2018. p. 141-8.
[46] Heras-Escribano M. The philosophy of affordances. Palgrave Macmillan; 2019. 232 p.
[47] Turvey MT. Affordances and prospective control: an outline of the ontology. Ecol. Psychol. 1992 Sep;4(3):173-87.
[48] Chemero A. An outline of a theory of affordances. Ecol. Psychol. 2003 Jan; 15(2):181-95.
[49] Thagard P. Desires are not propositional attitudes. Dialogue. 2006;45(1):151-56.
[50] Armstrong DM. A world of states of affairs. Cambridge: Cambridge University Press; 1997. 300 p.
[51] Grzankowski A, Montague M, editors. Non-propositional intentionality. Oxford: Oxford University Press; 2018. 320 p.
[52] Ceusters W, Smith B. Aboutness: towards foundations for the Information Artifact Ontology. In: Proceedings of ICBO2015. CEUR Workshop Proceedings, vol. 1515; c2015. p. 1-5.
[53] Biccheri L, Ferrario R, Porello D. Needs and intentionality: an ontological analysis and an application to public services. In: Proceedings of FOIS2020. Amsterdam: IOS Press; c2020. To appear.
[54] Peirce CS. The essential Peirce. vol. 2. Peirce edition project. Bloomington I.N.: Indiana University Press; 1998. 624 p.
[55] Searle J. Intentionality: an essay in the philosophy of mind. New York: Cambridge University Press; 1983. 278 p.
[56] Ethier JF, Barton A, Taseen R. An ontological analysis of drug prescriptions. Appl. Ontol. 2018 Nov;13(4):273-94.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200668

Embodied Functional Relations: A Formal Account Combining Abstract Logical Theory with Grounding in Simulation

Mihai POMARLAN and John A. BATEMAN
Bremen University, Bremen, Germany

Abstract. Functional relations such as containment or support have proven difficult to formalize. Although previous efforts have attempted this using hybrids of several theories, from mereology to temporal logic, we find that such purely symbolic approaches do not account for the embodied nature of functional relations, i.e. that they are used by embodied agents to describe fragments of a physical world. We propose a formalism that combines descriptions of a high level of abstraction with generative models that can be used to instantiate or recognize arrangements of objects and trajectories conforming to qualitative descriptions. The formalism gives an account of how a qualitative description of a scene or arrangement of objects can be converted into a quantitative description amenable to simulation, and how simulation results can be qualitatively interpreted. We use this to describe functional relations between objects in terms of spatial arrangements, expectations on behavior, and counterfactual expectations for when one of the participants is absent. Our method is able to tackle important questions facing an agent operating in the world, such as what would happen if an arrangement of objects is created and why. This gives the agent a deeper understanding of functional relations, including what role background objects, not explicitly asserted to participate in a functional relation such as containment, play in enabling or hindering the relation from holding.

Keywords. image schemas, embodiment, simulation, linguistic semantics, ontological analysis, formal ontology

1. Introduction: background and motivations

It is a fact universally acknowledged that an agent acting in a world must be in need of understanding how that world works. In humans, such an understanding is grouped under the label of commonsense, which includes aspects of intuitive physics and social behavior, and is acquired through one's own or observed experience, sometimes explicitly taught, but always sinking to an intuitive level that is hard to make explicit again. This presents a challenge for the creation of artificial agents that would be able to operate in the physical world in shared environments with humans. Commonsense turns out to be very difficult to capture formally, and finding the right balance of expressive power versus tractability is an unresolved problem, even despite some recent successes of machine learning in other AI domains. Perennial questions such as the utility of formal ontologies for capturing commonsense everyday knowledge also remain open.

Viewed abstractly, state of the art robot programming consists of following a script which may contain branches for some failure situations. However, problems occur as soon as events outside of the provided script occur. Consider, as an illustrative example, the situation where a robot is tasked with making popcorn. A pot containing oil and some corn kernels lies on a hot stove. Having grabbed a lid, the robot attempts to place it on the pot, but drops it on the way. Unaware of the mistake, the robot carries on with the next step, waiting for three minutes while the popcorn cooks. Anyone having encountered popcorn before can already tell what is wrong with this scene. A failure handling routine to detect and regrab a dropped lid would address only a symptom, not the root of the problem, *which is that the robot cannot leverage knowledge about the causal structure of the world into mechanisms to detect and repair failures as such failures arise*.

Our paper aims at addressing one of the parts needed to computationally implement such an ability: the need to represent practical world knowledge such that it is *transferable between situations*. To this end, one needs formal theories of functional relations and causal laws at a level that abstracts away from 'irrelevant' particulars, i.e. some qualitative descriptions of categories of scenes and causal laws. However, knowledge must also be grounded in procedures that can instantiate or recognize the instantiation of some general, qualitative pattern into a particular scene or arrangement of objects.

To bridge between these levels of abstraction, we draw on a formalisation of the notion of *image schemas* developed within cognitive linguistics (e.g., Johnson, Talmy [1,2]) and ontological formalizations pursued by Hedblom and others [3]. Image schemas are a plausible inventory of preconceptual building blocks of cognition, and can be used to describe functional relations between objects, i.e. relations whose applicability depends on the ability of objects to support certain behaviors, such as one object 'containing' another [4,5]. Such relations offer a useful level of description for characterizing expected or required participant behaviors and we formalize them by augmenting our ontological accounts with logical theories qualitatively describing spatial arrangements, expectations on behavior, as well as counterfactual expectations should required participants be absent. Formalizing appropriate levels of abstraction between generalized schemas and actual characterisation of physical locations and movements in the world (or in a simulation) allows us to define and address some core competency questions that an agent must engage with when acting in a changeable world: "what would happen if?", "why did something happen?", and "how can some state of affairs be brought about/avoided?" Being able to provide appropriate responses to such questions is a significant indication of an agent's understanding of the world in which it finds itself.

2. Background: Image Schemas

An image schema is defined by Johnson [1, p. xiv] as: "a recurring dynamic pattern of our perceptual interactions and motor programs that gives coherence and structure to our experience". Image schemas have attracted much attention in cognitive linguistics, and are thought to be involved in mental processes including metaphor construction and concept invention. Several authors have proposed lists of image schemas, with significant overlap among them; it thus seems likely that the number of image schemas in a complete inventory should not be very large. Nevertheless, until recently, formalizations of image schemas as theories amenable to computational implementation have been scarce.

This has now been addressed by Hedblom, Kutz, Neuhaus [3], and Hedblom, Kutz, Mossakowski and Neuhaus [6], who propose a first-order logic axiomatization of image schemas. This formalization has been used to explore both concept blending [7] and concept invention [8]. There remains, however, considerable work to be done towards developing a fully satisfying formal treatment. Firstly, we believe there are several requirements that current formalizations do not meet. At least some form of non-monotonic reasoning is needed: e.g., a table 'supports' a cup in a different way than a rack would. This raises questions concerning how best to distribute the various sources of knowledge that would enable flexible descriptions. Tractability must also be a prime concern. The difficulty of finding a formalism that can both answer commonsense reasoning problems and be well-behaved computationally is discussed by Davis [9].

Secondly, a purely logical model of image schemas appears to miss crucial properties that led originally to their very proposal. As defined by Johnson, image schemas capture patterns of *embodied* experience but, without proper grounding, logical models remain empty symbols. Geometric or physical concerns must enter into descriptions of image schemas but are awkward to capture using first-order logical machinery. E.g., in the well-known egg cracking problem (Morgenstern [10]), despite relatively complex axiomatization, only a handful of material resilience levels are described. Simply put, a purely logical approach uses the wrong tools for the job precisely because the critical contributions of embodiment are not available.

Simulation may then be a more appropriate tool for capturing intuitions about image schemas, but exhibits its own problems when used for commonsense reasoning; surveys addressing this topic are given by Davis and Marcus [11,12,13]. Simulation alone cannot answer basic questions such as what would be relevant to simulate. Reasoning is required, which we consider best provided by a logical component of a schematic description of the world *in addition* to any treatment in terms of simulations.

As a result, we propose the following requirements for a theory of image schemas; these requirements are meant to cover logical, as well as grounding, aspects of the theory:

- Non-monotonicity: allow defaults and exceptions when describing how an image schema would be instantiated in an arrangement of objects
- Well-foundedness: theories for schemas expressed in terms of simpler schemas, according to some principles of decomposability
- Tractability: crucial for agents acting in the physical world; implies the need for some sort of approximation or compromise in inferential power
- Correspondence to generative models: an image schema must be associated with procedures by which an agent can create instances of the schema
- Correspondence to perception procedures by which instantiations of image schemas can be recognized in an arrangement of objects.

We now detail our approach to formalizing image schema theories in a manner that is in line with these requirements and illustrate how we can then use this formalization to provide deepened understandings of the consequences of physical situations.

3. A Multi-stratal Ontological Treatment of Functional Relations

In this section we provide an overview of a pipeline for converting qualitative, underspecified descriptions of object arrangements into fully quantitatively specified scenes that are amenable to simulation (section 3.1), and then converting the quantitative simulation results back into qualitative descriptions of behavior (section 3.2). We argue that such a pipeline establishes a powerful tool for constructing theories of functional relations using Containment and Support as examples; a more complete inventory of the functional relations to be covered is specified in the GUM-Space ontology [5]. A purely logical theory might suffice for 'typical' combinations of objects, but a one-size fits all formalized approach is untenable when faced with the extreme variation and contingency of the real world [14]. Such an account would not allow us to interrogate whether, e.g., containment relations can actually hold between objects of widely varying shapes and sizes.

Simulation offers an additional reasoning mechanism well-suited for such geometric or physical aspects [15]. Although we take to heart the arguments from Davis and Marcus [12] against simulation, we find their perspective overstated. Human beings make inferences about functional relations even in the presence of great uncertainty. People expect their clothes to be in their luggage after a plane trip, even if they do not know how the luggage moved and cannot simulate the clothes inside. It is also true that simulators depend on having a physical model and that model may be inaccurate, unstable, or fail to cover interesting physics (although in such technical regards, simulators are only getting better). But embodied understanding of functional relations is not an either-or between purely logical approaches and simulation; both have their place. An inference rule such as "things in locked containers tend to stay there" would, for example, justify a traveller's conclusion that despite its unknown trajectory, the luggage still contains the clothes.

Of course, such a rule has many exceptions – e.g., if the container has holes bigger than the contents. Attempting to formalize the entire complex of possible configurations and their consequences as abstract rules is consequently both infeasible and unlikely to cover new situations. In contrast, simulators encode knowledge about the physical world in a very compact form. As an alternative reconciliation of the either-or case, therefore, Bateman [14] discusses the need for allowing flexible selections of formalizations that more appropriately and systematically distribute explanatory work across hybrid formalizations. This would allow logical theories to be used to specify simulated 'introspection' concerning physical arrangements, whose results may then be interpreted back into propositions that can be reasoned with at the symbolic level. We detail the arrangement and interpretation of such mental simulation experiments in the next two subsections.

3.1. "Scene generation": From Qualitative Description to Fully Specified Arrangements of Objects

We approach the general simulation specification task using image schemas, formalizing these across several levels of abstraction. This allows us to ontologically characterize image schemas not just in terms of interdependencies of logical theories sharing the same formalism [3], but also in terms of the nature of the formalisms needed at each particular level of abstraction. We then work towards full specifications, by which we mean a quantitative description in which shapes, coordinates, and physical properties of the objects involved in a scene are given. The input for this process of refinement is a qualitative description expressed in terms of functional or spatial relations between objects; this is of necessity (and also usefully) comparatively underspecified; there are always several ways to instantiate a qualitative description. Our process of refinement then operates by relating information at each of the following levels of abstraction maintained.

Functional relations are spatial relations between entities that also constrain behavior. An inventory of functional relations is given by subconcepts of *FunctionalSpatialModality* from GUM-Space [5]. New to our account here is an explicit formalization of the Expectations a functional relation gives rise to, and counterfactual Expectations about what would happen if some participant in the relation would be removed. We draw a distinction between expectation in a colloquial sense, and Expectation as a qualitative description of behavior conditional on an entity, as used in our formalization – that is, a colloquial expectation would be that popcorn is contained in the pot it cooks in; this comes from cultural norms concerning how a well-performed cooking task unfolds. In contrast, the formal Expectation that the containee remains in a container is part of our formalization of the Containment relation, and is inferred as soon as a Containment relation is asserted without requiring additional consultation of norms, tasks or contexts.

A theory template for a functional relation is a set of defeasible Horn clauses where the terms are predicates parameterized by variables; a variable may appear as an argument for several predicates. To produce a theory for a functional relation holding between particular objects, all variables must be consistently replaced by identifiers referring to objects in some environment or entities from an ontology of spatial relations [5], resulting in a propositional defeasible logic theory. We select defeasible logic to account for exceptional ways in which a functional relation may be brought about, and here we will only consider inferences on the propositional theories resulting from instantiating templates. As a consequence, our system can reason about the consequences of statements such as “the popcorn is in the pot” but, because of the exclusion of logical quantification, does not consider statements such as “there exists something which contains the popcorn”. This limitation is imposed to enable a clear separation of concerns between our system and more complex reasoners it may form a part of: our hybrid reasoning is a way to check to what extent a collection of propositions describing relations between specific objects is physically feasible, and to extract information, on physical grounds, about which other objects contribute to a relation, as described in our competency questions section; we note here that inference for propositional defeasible logic lies in P-time [16]. A task planner would be interested additionally in existentially quantified statements, e.g. whether there is some set of objects which can be arranged to obey a functional specification, and may then use our system to check candidate object sets.

A fragment of an example theory template is shown for Support in Listing 1, where $\Rightarrow$ denotes defeasible implication. The various predicates appearing on the righthand side correspond to lower layers of schemas defined in subsequent paragraphs. Capital single letters are variables that must be replaced when producing an instance of a theory, and “constants”, i.e. parametrizations of predicates by entities from an ontology valid for all instances, are given in quotation marks. A Default Expectation describes what should happen when all participants are allowed to physically interact. A counterfactual Expectation describes what should happen if one of the participants does not physically influence others. The descriptions of observed behavior, such as SpecificDirectionalDown, will be presented in section 3.2.

Listing 1: Fragment of the theory template for Support

```
Support(X,Y) ⇒ Location(X,Y,'on')
Support(X,Y) ⇒ Expectation(Default(), SpecificDirectionalStayLevel(X,Y))
Support(X,Y) ⇒ Expectation(Disabled(Y), SpecificDirectionalDown(X,Y))
Support(X,Y) ⇒ ¬Support(Y,X)
```

Spatial relations are schematic relations that constrain the placement of objects in terms of geometric primitive relations, such as alignments, between their primitive features. Theories for spatial relations are also instantiated from templates, and a theory of a spatial relation holding between a collection of objects is a propositional defeasible logic theory. An example theory template for Locations with spatial modality 'on' (cf. [5]) is given in Listing 2.

Listing 2: Fragment of the theory template for Locations with spatial modality 'on'

```
Location(X,Y,'on') ⇒
    SurfaceContainment(
        ObjectRelativeBottomSurface(X),
        WorldRelativeTopSurface(Y))
Location(X,Y,'on') ⇒
    AxisAlignment(
        ObjectRelativeUpright(X),
        WorldRelativeUpright(Y))
Location(X,Y,'on') ⇒
    ¬Location(Y,X,'on')
```

Instances of theories for spatial relations operating at a lower level of abstraction often appear because a functional relation implies a spatial relation; e.g., the theory for $Support(cup, table)$ would imply $Location(cup, table, \text{'}on\text{'})$. We require that the parameters that can be accessed to create an instance for a spatial relation theory are constrained by the entities mentioned at the more abstract level of the functional relation theory, and the spatial relation must not depend on entities not mentioned at the more abstract level – that is, the theory for $Location(cup, table, \text{'}on\text{'})$ must not reference some other object apart from the cup and table. We impose this limit because otherwise we would effectively have existential quantification, which we wish to avoid because of the separation of concerns mentioned above, and to avoid combinatorial explosion.

Geometric primitive relations describe constraints on how geometric parts of objects may be arranged. They are not formalized as logical theories but rather implemented as numeric procedures to generate and filter a set of candidate placements using a probability distribution on spatial configurations. This approach is standard in robotics for representing regions; detailed presentations may be found, for example, in work describing the Cognitive Robotics Abstract Machine [17,18] or Action Related Places [19]. One feature of this approach is the ability to combine several constraints on object relative placement under a uniform representation – thus, probability distributions corresponding to different constraints, e.g., AxisAlignment and SurfaceContainment, can be combined into a single distribution, corresponding to the conjunction of constraints.

As they occupy lower abstraction levels than spatial relations, geometric primitive relations are typically invoked because they are implied by the theory of some spatial relation; e.g., $Location(cup, table, \text{'}on\text{'})$ implies

$$AxisAlignment(ObjectRelativeUpright(X), WorldRelativeUpright(Y))$$

This means that the entities participating in a geometric primitive relation must be a subset of the entities participating in the invoking spatial relation or their parts. The parts are obtained by invoking the next lower level of abstraction of the geometric primitives themselves, which are described next.

Geometric primitives are specified in terms of procedures to convert a shape description of an object into a representation of one of its parts or features. Examples include the centroid of an object, its outer surface, or its length axis. Primitives may be object- or world-relative, depending on which coordinate system they use. They can be determined by geometry (e.g. a PCA analysis can identify a longest axis) or asserted by convention (e.g. human objects are often designed with a particular direction intended as upright).

3.2. "Behavior interpretation": from quantitative simulation results back to qualitative descriptions

The behaviors we model currently as illustrative examples are represented by primitive movements, which are movements that describe the motion of some trajector object relative to another object, the relatum; these are covered by relational spatial modalities within the GUM-Space ontology [5]. In particular, the primitive movements we have considered so far are the two *GeneralDirectional* modalities 'Nearing' and 'Approaching', *SpecificDirectional* movements constrained to the vertical direction, and two further movement descriptions defined for modeling convenience: *RelativeMovement* and *RelativeStillness*.

We formalize primitive movements in terms of recognition procedures that take trajectory data as input – i.e., the relative pose of the trajector to the relatum at different time steps – and compute a cumulative cost over the duration of the input trajectory. The cost measures to what extent the actually observed trajectory deviates from the specification. This is not the same as deviating from some ideal trajectory, however. There is, for example, no ideal trajectory for RelativeApproach; instead, displacements that move the trajector away from the relatum are counted towards increasing the trajectory cost. If and only if the cost exceeds a threshold value is the observed trajectory deemed not to respect the primitive movement. The threshold is currently based on a fraction of the sum of the lengths – i.e., the longest axes – of the participating objects. This fraction may differ for different primitive movements, but currently we set it to a tenth of the sum of lengths of the relatum and trajector. This might be finetuned by a number of methods.

In the section following we proceed to the competency questions relevant for the new levels of formalization introduced and show how they can be computationally implemented within our system, building on the levels of representation defined.

4. Competency questions enabled by a multi-layered schematic approach to physics reasoning

To begin, we summarize our competency questions thus: "what would happen if?", "why did something happen?", and "how can some state of affairs be brought about/avoided?". These questions seem very natural, but they hide several sources of complexity.

One important set of concerns involves just how far into the future do we push a "what if?" question and how far into the past do we push a "why?" For the purposes of formalization, we must be explicit about our horizons. Why-questions also pose the problem of defining what counts as a cause. What-to-do questions are hard to solve in general, because planning is complex; it is more plausible that what humans do is learn routines which are appropriate to some class of situations, and to some degree adaptable. Finally, the level of abstraction at which these questions should be answered needs specification. For example, one can always analyse a cause in finer detail, assuming the data is there but, very often, we do not seem to care in our activities about the motion of many small component parts, and instead prefer high level descriptions. Fuller specifications of the competency questions at issue will now be listed, in each case showing how answering them is implemented.

"What if" questions. These questions are understood here as taking a qualitative description of an arrangement of objects, e.g. a pot contains cooking popcorn, and outputting a qualitative description of how the arrangement would naturally behave, e.g. the popcorn distances itself relative to the pot. Conversion from a qualitative description to fully specified arrangements of objects, including coordinates and initial velocities, proceeds down along the hierarchy of levels described in section 3.1 above.

As an example of a schematically described scene, let us consider:

Containment (container=pot, containee=popcorn).

The Containment schema is a functional relation, and so has a theory constraining both the spatial placement of the entities and expectations on their behavior as suggested above. The placement of the pot and popcorn is simply the spatial relation schema:

Location (relatum=pot, locatum=popcorn, spatial_modality='in')

which in turn further implies the following geometric primitive relation constraint:

VolumeContainment(big_volume=InteriorCavity(pot), small_volume=popcorn).

The VolumeContainment primitive relation guides sampling for positioning pot and popcorn such that the relevant geometric parts (an interior cavity in the case of the pot, and the popcorn itself) obey the volume containment constraint. A fully specified scene can then be simulated, and the trajectories of objects analyzed to check correspondence to the relevant movement schemas (cf. section 3.2).

An issue however appears at this second step: what schemas should be tested against the trajectory data? And since schemas have multiple participants, how does one know for which participants to do the test? This is related to the critique from Marcus and Davis [12] that a simulation, on its own, does not offer guidance about how it should be interpreted, or what objects or movements are significant. Indeed, the bare facts of an activity may have many interpretations, and people appear to select such interpretations based on an interpretive framework constructed from contextual expectations. Dropping a cup means one thing in an argument, and another when bringing a drink.

To model such interpretive frameworks, we take the approach that a "what if" question must itself specify the schemas to test for. In other words, what we understand as a "what if" question is to check which, if any, of some qualitative expectations on the behaviors of some objects will hold, assuming the objects are arranged to satisfy some qualitative spatial constraints. The expectations to check are then built into the qualitative description of the scene in terms of functional relations. In the pot and popcorn example in the context of cooking, the Containment relation has, among others, the expectation:

Expectation (condition=Default(), event=RelativeStillness(a=pot, b=popcorn)).

This guides the interpretation of the simulation by pinpointing just those objects and movements that are relevant, and establishes a criterion to judge said movement. In this case, some of the popcorn particles will actually escape the popcorn interior, thereby violating the RelativeStillness expectation of Containment.

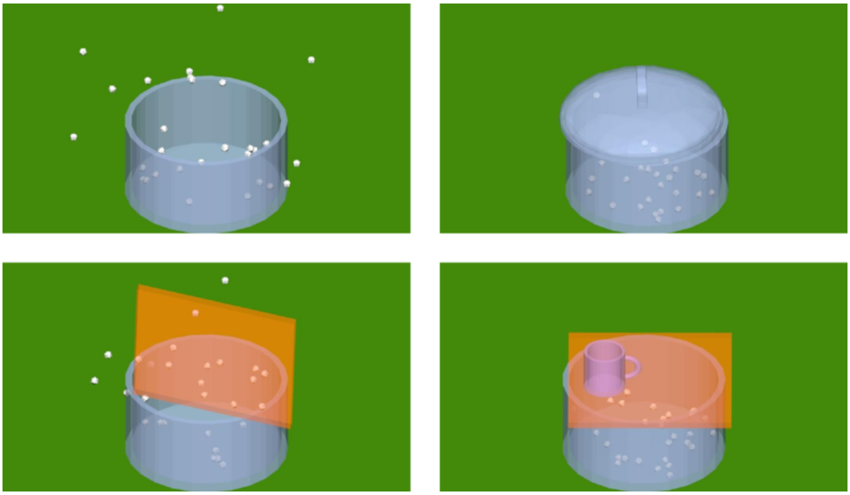


Figure 1. "What if" example scenarios: can various arrangements of objects contain popcorn?

Further examples of such "what if" questions follow for illustration. These are scenes involving pots, popcorn (which starts with some initial random velocity), lids, balsa boards (very lightweight), and cups. Screenshots of some frames from simulations relative to specific questions illustrative of the observed behaviors are shown in figure 1.

Question	Scene specification	Gloss of Result
"what if we tried to contain popcorn in a pot?"	Containment(pot, popcorn)	containment fail: popcorn flies out
"what if we tried to contain popcorn in a pot with a lid on top of it?"	Containment(pot, popcorn), Support(pot, lid)	ok
"what if we tried to contain popcorn in a pot with a light balsa board on top of it?"	Containment(pot, popcorn), Support(pot, balsa)	containment fail: popcorn flies out; support fail: balsa board does not stay level relative to pot
"what if we tried to contain popcorn in a pot with a light balsa board on top of it, and a cup on top of the balsa board?"	Containment(pot, popcorn), Support(pot, balsa), Support(balsa, cup)	ok

"Why" questions. These questions are understood here as attributing blame/credit to objects in a scene for the observed non/compliance of observed behavior to qualitative expectations placed on the scene by functional relations.

The approach we took to operationalize causality testing is interventionist [20,21]: an object can only be credited for a behavior if, by removing the object's influence from the scene, the behavior is no longer observed. Removing an object's influence means to stop it from interacting physically with other objects; we do not remove the object because its presence may be necessary for qualitative behavioral specifications, i.e., movement schemas, but we can readily prevent physical interactions. We refer to an object without physical influence as a phantom. Note that phantoms pose no problem for the simulator in terms of the physical consistency of the worlds created – they are simply ignored when performing updates of the physical state.

As an example of using phantom objects, let us consider two scenes, one in which we have a pot with cooking popcorn inside and covered by a lid, and another scene in which the pot is covered by a lid and contains a meditating pixie. In the former case, the popcorn particles have just popped and fly out. In the latter case, the pixie is simply content to levitate in place. The behavioral specification we are interested in for deriving potential causes is the prevention of a relative distancing between the pot and the popcorn or pixie. Obviously, both popcorn and pixie stay inside the pot if both the pot and lid are physically active, which by virtue of their properties as solid objects prevent objects from passing through. If, however, either pot or lid are phantoms, the popcorn will escape. In contrast, making the lid a phantom will still not result in the pixie leaving the interior region of the pot. Our system would then say that both the pot and the lid contribute towards keeping the popcorn near the pot, but the lid is not the cause of the pixie remaining in the pot.

Thus, in our framework, a "why" question must specify some schematic behaviors of interest, and a list of objects which may be responsible for those behaviors. Several scenes are simulated, one default scene in which all objects participate physically, and counterfactual scenes in which, in turn, one of the objects is turned into a phantom. Further illustrative examples of "why" questions follow. Having observed compliance (or not) to some functional specification of a scene in the previous section, we ask why the observed behaviors happened.

Question	Scene specification	Gloss of Result
"when a lid is on the pot, why does the popcorn stay inside the pot?"	Containment(pot, popcorn), Support(pot, lid)	both lid and pot are needed for containment
"a balsa board is on the pot; why does the balsa board fly off a pot with popcorn in it?"	Containment(pot, popcorn), Support(pot, balsa)	the popcorn is to blame, not the pot
"a balsa board is on the pot, and a cup is on the balsa board; why does popcorn stay in the pot?"	Containment(pot, popcorn), Support(pot, balsa), Support(balsa, cup)	pot, balsa board, and cup are all necessary for containment

Note that in the scenarios given, the containment relation is assured by interactions between objects which at the qualitative level are not explicitly asserted to contribute to the containment. This shows why adding a physical layer adds to the understanding of a situation beyond what a purely symbolic and qualitative approach is capable of. Some illustrative screenshots from simulations of 'counterfactual' scenarios are shown in figure 2. 'Phantom' objects are distinguished by black colors and higher transparency.

"What to do" questions. While such questions suggest planning, planning is expensive and in practice does not seem to be used often by humans. Instead, human beings learn simple rules of action, e.g. "to prevent popcorn from flying out of a pot, cover the pot". Answering a "what to do" question becomes a process of checking which such action rule might apply in a given situation, and what that might entail in terms of changes to an arrangement of objects. The focus of this competency question therefore is not on deliberation, but rather on the representational structure needed to know what are good responses to some class of situations. How these responses are acquired is a separate issue, for which we suspect pragmatic considerations are paramount. An agent might learn from instruction, or from observation, or even by simulated or actual experiments performed out of 'curiosity'.

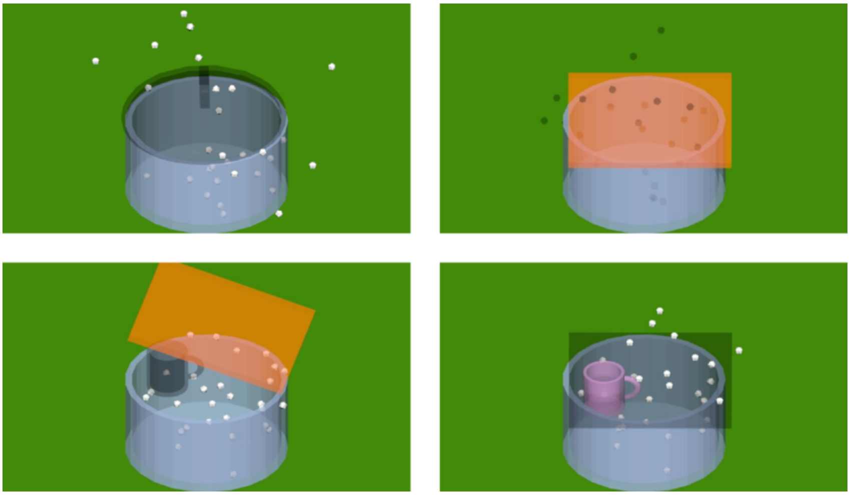


Figure 2. "Why" example scenarios: to check which objects contribute to an observed behavior, what happens when some objects have physical interactions disabled (marked by transparent black texture)?

To see how such a question could be answered, consider the example from the previous section of containing popcorn in a pot. Assume the following action rule: if an item of type pot fails to contain some other item X, then place another item Y on the pot. The $\mapsto$ symbol means a transformation from a scene qualitatively described by the left side to a scene qualitatively described by the right.

$$\neg Containment(X, pot) \mapsto Containment(X, pot) \wedge Support(Y, pot)$$

The rule antecedent, $\neg Containment(X, pot)$, can be asserted from prior knowledge or observed from real or simulated scenes. Deciding what object to add to a scene requires having a list of candidates to try out in simulation, e.g., small, manipulable items known to exist in the kitchen. A candidate is successful if the expectations of the functional relations are all met. Suppose then that possible candidates for Y are a cup, a plate, or a balsa board. Of these, only the plate achieves the intended result; the cup falls in the pot without stopping the popcorn; the balsa board is pushed off by the escaping popcorn.

Moreover, and as we have seen, combining items (e.g. the balsa board and the cup) can help the pot achieve containment too. Searching for such combinations of scene modifications might be done in an iterative deepening fashion, where modifications involving more objects are searched only if it is not possible to fix a scene with fewer items.

5. Related work

We refer to a recent survey by Davis and Marcus [22] for an overview of research into commonsense reasoning. By and large, commonsense reasoning has been pursued by way of attempting to construct either large repositories of facts – i.e., knowledge graphs – or ontologies [23], or rich logical theories often involving mixed formalisms to cover aspects such as time, geometry, topology [24]. Such logical approaches have been criticized, often by their own proponents [25,9], on the grounds of requiring intractable or undecidable formalisms, or, as in [14], on the grounds of over-commitments causing them to need complex formalisms in the first place. Naturally, critiques of logical approaches to commonsense inference have also been made on the same grounds as critiques against purely symbolic approaches to AI in general [26,27].

A strong case can be made, however, that machine learning approaches fare no better. Although deep learning can construct agents that master a specific game, such agents do not have a conceptual understanding that would allow them to transfer their competence to even slightly modified versions of it (Kansky [28]). Kansky suggests "schema networks", a hybrid between a classical propositional logic and learning, as an approach to remedy this, but it is not yet clear if they would scale beyond the worlds of very simple Atari games. More complex sensorimotor concepts have been modelled as control policies or state estimation routines for partially observable Markov processes [29], but the way these policies are learned depends strongly on a curriculum, suggesting that knowledge of the concepts needs to be already formalized somewhere else, and in particular the dependencies of complex sensorimotor concepts on simpler ones must be explicitly known by whoever sets up the training protocol.

Spatial reasoning is also a very relevant area, comprising topics such as parthood [30] or region connectivity [31], quantitative reasoning about how a qualitative arrangement might be instantiated [32], and linguistically-motivated ontological modelling of spatial relations such as the GUM-Space ontology [5] or the theory of sense clusters [33]. Spatial calculi are also applied in Hedblom et al.'s latest characterizations of image schemas [34]. We have made considerable use throughout of previous work on image schemas, originally proposed by Johnson [1] as remarked above. We view our work as a continuation of the formalization attempts of Hedblom and others [3,6] in which we combine a logical formalism with geometric and simulation techniques to provide grounding for image schemas into generative models and recognition procedures. This then extends previous accounts towards interaction with embodied simulation as well.

An ontological characterization of causal/causal-like relations between individual occurrents has been provided by Galton [35]. On the practical side however, we have chosen to follow the treatment of causation offered by Pearl [20], i.e. an interventionist understanding of causation [21]: X can be a cause of Y if some change to X specifically results in a change to Y. As a result, in this work we analyze causal relations by tracking how particular modifications to a simulated scene alter the observed qualitative behavior.

6. Conclusions

An understanding of the physical world an agent is embodied in requires a hybrid formalism: one able to operate at a high level of abstraction, and hence generality, but also account for physical and geometric aspects of the world. A difficulty in creating such a hybrid is the tension between the need for underspecification when one aims for generally applicable knowledge, and the requirement for precisely quantified descriptions usable by tools for modelling physical interactions, such as simulation.

We resolve this tension by taking inspiration from image schemas, which are intended to be strongly related to embodied interactions as well as amenable to logical formalization. We propose a multi-layer formal approach, where each layer is characterized by a different level of abstraction and modelling task. The most abstract level is that of functional relations qualitatively describing expectations on object behavior and formalized in terms of spatial relations and primitive movements. These are described, at lower levels of abstraction, via generative models to instantiate and recognize arrangements of objects that satisfy a qualitative description.

We use our formal theories of functional relations to answer questions about object arrangements, such as whether particular arrangements can enact a functional relation and why (not), and show that our approach allows a deeper understanding of such functional relations, including how background objects, not explicitly participating in the relation, contribute to it. We have also sketched how our approach could be used to describe response rules for an agent – what to do in particular situations in order to achieve some qualitative goal – but we leave further developments in this direction for future work.

Acknowledgments

The research reported in this paper was supported by the German Research Foundation (DFG), as part of the Collaborative Research Center (Sonderforschungsbereich) 1320 "EASE - Everyday Activity Science and Engineering", University of Bremen (http://www.ease-crc.org/). The research was conducted primarily within subproject P01: 'Embodied Semantics for the Language of Action and Change'.

References

[1] Johnson M. The body in the mind: the bodily basis of meaning, imagination, and reason. University of Chicago Press Chicago; 1987.

[2] Talmy L. The fundamental system of spatial schemas in language. In: Hampe B, editor. From perception to meaning: image schemas in cognitive linguistics. Berlin: Mouton de Gruyter; 2006. p. 37–47.

[3] Hedblom MM, Kutz O, Neuhaus F. Choosing the Right Path: Image Schema Theory as a Foundation for Concept Invention. Journal of Artificial General Intelligence. 2015;6(1):21–54.

[4] Coventry KR, Garrod SC. Saying, seeing and acting. The psychological semantics of spatial prepositions. Essays in Cognitive Psychology series. Hove, UK: Psychology Press; 2004.

[5] Bateman JA, Hois J, Ross R, Tenbrink T. A Linguistic Ontology of Space for Natural Language Processing. Artif Intell. 2010 Sep;174(14):1027–1071.

[6] Hedblom MM, Kutz O, Mossakowski T, Neuhaus F. Between Contact and Support: Introducing a logic for image schemas and directed movement. In: 16th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2017); 2017. p. 256–268.

[7] Hedblom MM, Kutz O, Neuhaus F. On the Cognitive and Logical Role of Image Schemas in Computational Conceptual Blending. In: Lieto A, Radicioni DP, Cruciani M, editors. AIC 2014 Artificial Intelligence and Cognition. vol. 1315 of CEUR Workshop Proceedings. CEUR-WS.org; 2014. p. 110–121.

[8] Hedblom MM, Kutz O, Neuhaus F. Image Schemas and Concept Invention. In: Confalonieri, R and Pease A, Schorlemmer M, Besold TR, Kutz O, Maclean E, Kaliakatsos-Papakostas M, editors. Concept Invention. Springer; 2018. p. 99–131.

[9] Davis E. Qualitative Spatial Reasoning in Interpreting Text and Narrative. Spatial Cognition & Computation. 2013;13(4):264–294. Available from: https://doi.org/10.1080/13875868.2013.824976.

[10] Morgenstern L. Beyond Toy Problems: A Logical Formalization of the Egg-Cracking Domain. In: Fourth International Symposium on Logical Formalizations of Commonsense Reasoning, http://www-formal.stanford.edu/leora/cs98/egg.a.ps; 1998. .

[11] Ludwin-Peery E, Bramley N, Davis E, Gureckis TM. Limits on the Use of Simulation in Physical Reasoning. In: Proceedings of the 41th Annual Meeting of the Cognitive Science Society, CogSci 2019: Creativity + Cognition + Computation, Montreal, Canada, July 24-27, 2019; 2019. p. 707–713.

[12] Davis E, Marcus G. The scope and limits of simulation in automated reasoning. Artificial Intelligence. 2016;233:60–72.

[13] Davis E, Marcus G. The Scope and Limits of Simulation in Cognitive Models. CoRR. 2015;abs/1506.04956. Available from: http://arxiv.org/abs/1506.04956.

[14] Bateman JA. Space, Language and Ontology: A Response to Davis. Spatial Cognition & Computation. 2013;13(4):295–314.

[15] Bateman J, Pomarlan M, Kazhoyan G. Embodied contextualization: Towards a multistratal ontological treatment. Journal of Applied Ontology. 2019;14(1):1–35.

[16] Maher MJ. Propositional Defeasible Logic Has Linear Complexity. Theory Pract Log Program. 2001 Nov;1(6):691–711. Available from: https://doi.org/10.1017/S1471068401001168.

[17] Beetz M, Mösenlechner L, Tenorth M. CRAM – A Cognitive Robot Abstract Machine for Everyday Manipulation in Human Environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. Taipei, Taiwan; 2010. p. 1012–1017.

[18] Mösenlechner L, Beetz M. Fast Temporal Projection Using Accurate Physics-Based Geometric Reasoning. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Karlsruhe, Germany; 2013. p. 1821–1827.

[19] Stulp F, Fedrizzi A, Mösenlechner L, Beetz M. Learning and Reasoning with Action-Related Places for Robust Mobile Manipulation. Journal of Artificial Intelligence Research (JAIR). 2012;43:1–42.

[20] Pearl J. Causality. 2nd ed. Cambridge University Press; 2009.

[21] Woodward J. Making Things Happen: A Theory of Causal Explanation. Oxford University Press; 2004.

[22] Davis E, Marcus G. Commonsense reasoning and commonsense knowledge in artificial intelligence. Commun ACM. 2015;58(9):92–103. Available from: https://doi.org/10.1145/2701413.

[23] Cycorp. OpenCyc 0.7.0; 2004. Available from: http://www.opencyc.org.

[24] Davis E, Marcus G, Frazier-Logue N. Commonsense reasoning about containers using radically incomplete information. Artificial Intelligence. 2017;248:46 – 84.

[25] Davis E. Naive Physics Perplex. AI Magazine. 1998 Dec;19(4):51.

[26] Churchland PM. On the nature of theories: a neurocomputational perspective; 1990. .

[27] Smith BC. The owl and the electric encyclopedia. Artificial Intelligence. 1991;47:251–288.

[28] Kansky K, Silver T, Mély DA, Eldawy M, Lázaro-Gredilla M, Lou X, et al. Schema Networks: Zero-shot Transfer with a Generative Causal Model of Intuitive Physics. In: Precup D, Teh YW, editors. Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. vol. 70 of Proceedings of Machine Learning Research. PMLR; 2017. p. 1809–1818.

[29] Hay N, Stark M, Schlegel A, Wendelken C, Park D, Purdy E, et al. Behavior Is Everything: Towards Representing Concepts with Sensorimotor Contingencies. In: McIlraith SA, Weinberger KQ, editors. Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018. AAAI Press; 2018. p. 1861–1870.

[30] Casati R, Varzi AC. Parts and places: the structures of spatial representation. Cambridge, MA and London: MIT Press (Bradford Books); 1999.

[31] Randell DA, Cui Z, Cohn AG. A Spatial Logic Based on Regions and Connection. In: Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning. KR'92. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1992. p. 165–176.

[32] Bhatt M, Lee JH, Schultz C. CLP(QS): A Declarative Spatial Reasoning Framework. In: Egenhofer M, Giudice N, Moratz R, Worboys M, editors. Spatial Information Theory. Berlin, Heidelberg: Springer; 2011. p. 210–230.

[33] Bennett B, Cialone C. Corpus Guided Sense Cluster Analysis: a methodology for ontology development (with examples from the spatial domain). In: Garbacz P, Kutz O, editors. Formal Ontology in Information Systems - Proceedings of the Eighth International Conference, FOIS 2014, September, 22-25, 2014, Rio de Janeiro, Brazil. vol. 267 of Frontiers in Artificial Intelligence and Applications. IOS Press; 2014. p. 213–226. Available from: https://doi.org/10.3233/978-1-61499-438-1-213.

[34] Hedblom MM, Kutz O, Peñaloza R, Guizzardi G. Image Schema Combinations and Complex Events. KI – Künstliche Intelligenz. 2019;33:279–291.

[35] Galton A. States, Processes and Events, and the Ontology of Causal Relations. In: Formal Ontology in Information Systems - Proceedings of the Seventh International Conference, FOIS 2012, Gray, Austria, July 24-27, 2012; 2012. p. 279–292.

IV. Parts and Wholes

This page intentionally left blank

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200670

A Mereology for Connected Structures

Michael GRÜNINGER [a], Carmen CHUI [a], Yi RU [a], Jona THAI [a]
[a] *Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario, Canada*

Abstract. Classical mereology is based on the assumption that any two underlapping elements have a sum, yet there are many domains (such as manufacturing assemblies, molecular structure, gene sequences, and convex time intervals) in which this assumption is not valid. In such domains, mereological sums must be connected objects. However, there has been little work in providing an axiomatization of such a mereology. Based on the observation that the underlying structures in these domains are represented by graphs, we propose a new mereotopology that axiomatizes the connected induced subgraph containment ordering for a graph, and then identify an axiomatization of the mereology that is a module of the mereotopology.

Keywords. mereology, mereotopology, lattices, graphs

1. Introduction

From its inception, research in mereology has been dominated by two presuppositions. One has focused on what has come to be known as classical mereology, in which underlapping elements have sums and overlapping elements have products. The other presupposition, known as mereological monism, is based on the idea that there is a single parthood relation that applies to all domains, whether they be spatial regions, temporal intervals, physical objects, or activities.

More recently, mereological pluralism has been posited in [1, 2], in which there are multiple distinct parthood relations for different classes of objects. Furthermore, there are a wide variety of domains (such as manufacturing assemblies, molecular structure, gene sequences, and time intervals) in which we need mereologies for connected substructures, not arbitrary substructures. The problem is that existing mereologies (such as classical extensional mereology) are too strong to represent connected substructures, that is, they allow models in which disconnected elements have mere sums. We therefore want to address the following challenge:

What is the mereology for connected substructures of a structure?

In designing an ontology, our objective is twofold – first, to prove that the models of the ontology are actually the intended models, and second, to demonstrate that the intended models do indeed formalize the ontological commitments. Our strategy is to first specify a class of mathematical structures and show that the ontology axiomatizes this class of structures (that is, there is a one-to-one correspondence between the class of models of the ontology and the class of mathematical structures). We then specify a representation theorem for this class of mathematical structures to demonstrate that it formalizes the ontological commitments. The primary benefit of this strategy is that it makes explicit the modular organization of the subtheories of the ontology, thereby highlighting how other ontologies are reused.

We therefore begin by presenting a series of motivating scenarios from diverse domains in which the underlying structures are graphs, and we seek the mereologies for connected induced subgraphs of these graphs. After showing that existing mereologies are inadequate for capturing these scenarios, we propose new mereotopologies that axiomatize the connected induced subgraph containment ordering for a graph. We then identify the axiomatization of the mereology that is a subtheory of the mereotopology.

2. Motivating Scenarios

2.1. Assemblies and Components

A three leg table as shown in Figure 1(i) has a topological structure as seen in Figure 1(ii). Table top a is connected to all the legs b, c, and d, while all the legs are disconnected from each other. Connected components (e.g., a and b) can have sums that correspond to subassemblies, while disconnected components (e.g., b and c) do not constitute subassemblies, and hence do not have sums. The complete set of subassemblies for the table is shown in Figure 1(iii). Similarly, for a picture frame, the bars a, b, c, and d topologically form a cyclic graph (see Figure 2). Bars a, c and bars b, d are not directly connected, so there is no sum for each of these two pairs. The mereologies for these two examples are shown in Figure 1(iv) and Figure 2(iv), and it is clear that these are not classical mereologies.

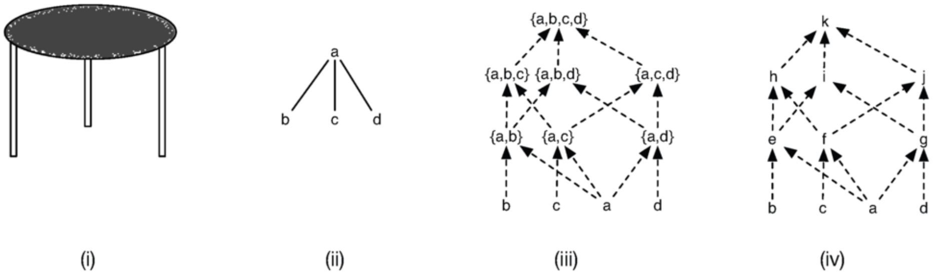


Figure 1. Mereology on the components of a table.

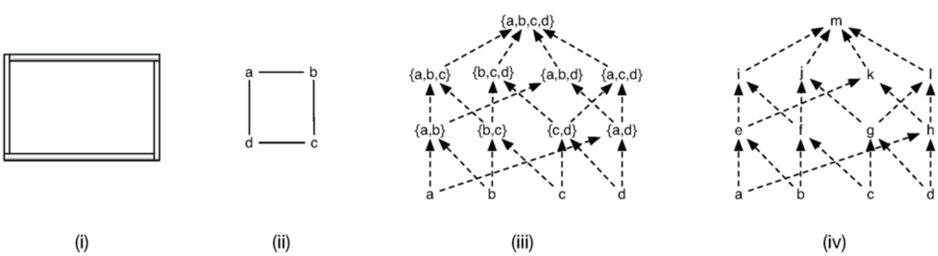


Figure 2. Mereology on the components of a picture frame.

2.2. Convex Time Intervals

Relations over temporal intervals have been foundational for qualitative temporal reasoning and representation. An early proposal for the axiomatization of an ontology of time intervals was the work of van Benthem [3], in which there is one primitive ordering relation and one primitive mereological relation over intervals. Notably, the intervals in the models of this ontology are convex – every interval between two other subintervals is also a subinterval[1]. In Figure 3(i), there is no interval that is the sum of intervals **a** and **c**; although interval **f** is the least upper bound of **a** and **c**, it contains the interval **b** which is disjoint from **a** and **c**. The mereology in Figure 3(ii) is not classical.

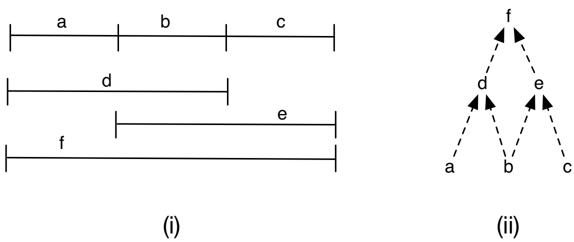


Figure 3. Mereology on convex time intervals.

2.3. Molecular Structure Ontology (MoSt)

MoSt [4] combines conventional graph theory and ontological approaches to describe the shape of molecules. This ontology allows us to consider molecules from the shape perspective by identifying basic functional groups of the ring and chain types, and to use the axioms of the ontology to combine these functional groups together.

A skeleton in MoSt is the composition of one or more functional groups that are attached together. Skeletons can be composed of other skeletons – they allow us to partition the structure of molecules into various pieces, along with also combining pieces together. Because we allow various *decompositions* of molecules, we also must permit the notion that multiple skeletons can be formed from the combination of primitive functional groups with other groups or other atoms. We introduce a parthood relation called $part(x,y)$ that outlines how two skeletons x and y are part of each other *if and only if* all elements found inside one skeleton are also found in the other skeleton:

$$(\forall x \forall y\, (part(x,y) \equiv (skeleton(x) \wedge skeleton(y) \wedge \forall z\, ((mol(z,x) \supset mol(z,y))))))$$

This is exemplified with Figure 4, where s_3 is composed of s_1 and the skeleton for g_3. We can state that "s_1 is part of s_3." We are again faced with the question: *What mereology corresponds to this definition of parthood?*

[1] van Benthem recognizes the need to axiomatize convex intervals, but he does not provide an explicit axiomatization of the mereology alone.

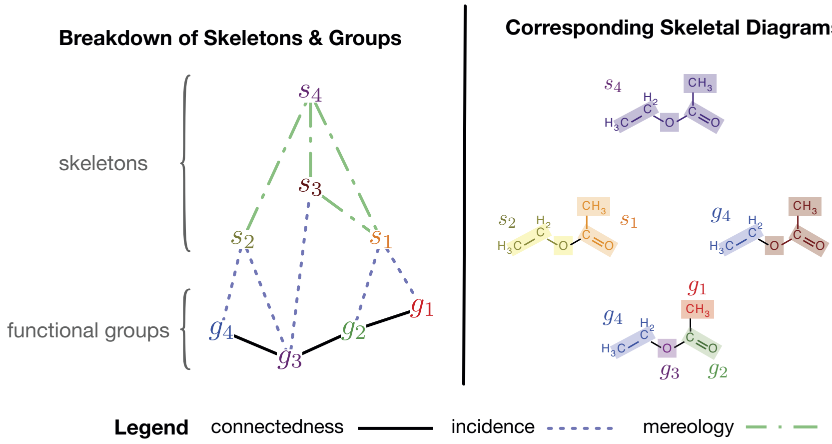


Figure 4. Composition of the skeleton for ethyl acetate. g_1, g_2, g_3, and g_4 signify the primitive functional groups, and s_1, s_2, s_3, and s_4 signify the skeletons, respectively. The primitive functional groups are connected via the $mol(x,y)$ relation using the dark, bolded black lines. Skeletons that contain functional groups are outlined in the dotted blue lines in the figure; for example, s_2 consists of g_3 and g_4. Green dash-dotted lines show parthood between skeletons.

2.4. Gene Sequences

A gene is a sequence of nucleotides that encodes the synthesis of proteins. A reading frame is a way of dividing such a sequence of nucleotides into a set of consecutive, non-overlapping triplets (shown in Figure 5(i)). We can therefore specify a mereology on gene sequences; for example, given the sequence in Figure 5(ii), the containment ordering in Figure 5(iii) is isomorphic to the mereology in Figure 5(iv).

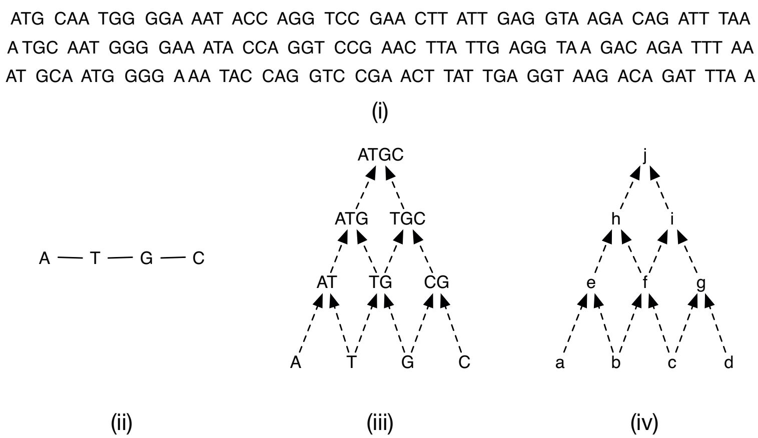


Figure 5. Mereology on nucleotides in a gene sequence.

3. Relationship to Existing Mereotopologies

Classical mereology is a formal theory of the part-whole relation [5]; in particular, it is a theory which captures how parts can be combined to form wholes and how wholes can be decomposed into parts. Pietruszczak [6] goes so far as to say that mereology arose as a theory of sums. As posited by Fine [2] and Sider [7], the prevalent intuition is that "a whole is a mere sum, or aggregate or fusion, formed from its parts without regard for how they might fit together or be structured within a more comprehensive whole." All of this is captured by $T_{cm\_mereology}$[2], in which any pair of overlapping elements have a product and any pair of underlapping elements has a sum (although the precise definition of *sum* varies across different axiomatizations [5]).

Nevertheless, in domains such as manufacturing assembly, molecules, gene sequences, and convex time intervals, not all underlapping elements have sums. Thus, no mereology that extends $T_{cm\_mereology}$ can be used to represent the scenarios in Section 2. It is clear that the mereologies we need in the motivating scenarios are *not* classical, since sums do not exist for all pairs of elements in the mereologies of Figure 1, 3, and 5.

Simons [8] addresses this problem in his discussion of integral wholes. He correctly notes that one commonality among all of the above scenarios is that the objects we are considering must be *connected*. The assembled table is distinct from the set of tabletop and legs scattered on the floor. A skeleton within a molecule must be connected – it never consists of two disconnected functional groups within a molecule. Why not leave the mereology to be classical and capture the notion of connectedness using mereotopology? In fact, Whitehead proposed a nonclassical mereology (i.e., one in which not all pairs of elements have sums) based on the notion of self-connected objects [8]. This approach has been criticized in [9] from the perspective of attempting to define connection with respect to parthood, but it was not considered as an independent mereotopological axiom.

In this paper, we are not proposing a mereology for *all* objects; for example, a mereology of space needs to cover all spatial regions, and not be restricted to connected regions. Instead, we are proposing a mereology that is satisfied by different specific classes of entities, such as assembled physical objects, molecules, gene sequences, and convex time intervals. Of course, there do exist objects for which sums are not connected; for example, a bikini is an object whose parts are disconnected, and the United States is a geographical entity whose parts are not connected. Indeed, this is the primary reason for adopting mereological pluralism – not all classes of objects have the same parthood relation satisfying equivalent axioms. For example, the parthood relation for convex time intervals is distinct from the parthood relation for arbitrary time intervals; the latter can be represented using a classical mereology, while the former cannot.

4. Semantic Requirements: Connected Induced Subgraph Structures

If we take a closer look at the motivating scenarios, we can get a sense of what requirements we need to impose on the models of the mereology that we need. We begin with the observation that each of the motivating scenarios involves subgraphs of a simple graph.

Definition 1 *A graph is a pair of sets* $\mathbb{G} = \langle V, \mathbf{E} \rangle$ *such that*

[2] http://colore.oor.net/mereology/cm_mereology.clif

1. $\mathbf{E} \subseteq V \times V$*;*
2. $\mathbf{E} = \mathbf{E}^{-}$*;*

$\mathbb{G}$ *is simple iff* $I \cap \mathbf{E} = \emptyset$. $\mathbb{G}$ *is a graph with loops iff* $I \subseteq \mathbf{E}$.

In the case of assembled products, the underlying graph is the connection relation between non-decomposable components; subassemblies correspond to subgraphs of the graph that corresponds to the entire object. Molecules within the MoSt Ontology are specified by a graph consisting of functional groups, and skeletons correspond to subgraphs of the graph of the entire molecule. With convex time intervals and gene sequences, we have the special class of path graphs, and each sequence forms another path that is a subgraph of the entire graph.

Definition 2 *Let* $\mathbb{H}_1 = \langle V_1, \mathbf{E_1} \rangle$ *and* $\mathbb{H}_2 = \langle V_2, \mathbf{E_2} \rangle$ *be simple graphs.*
$\mathbb{H}_2$ *is a subgraph of* $\mathbb{H}_1$ *(denoted by* $\mathbb{H}_2 \subseteq \mathbb{H}_1$*) iff* $V_2 \subseteq V_1$ *and* $\mathbf{E_2} \subseteq \mathbf{E_1}$.
$\mathbb{H}_2$ *is an induced subgraph of* $\mathbb{H}_1$ *(denoted by* $\mathbb{H}_2 \preceq \mathbb{H}_1$*) iff*

- $\mathbb{H}_2$ *is a subgraph of* $\mathbb{H}_1$*;*
- *For any* $\mathbf{x}, \mathbf{y} \in V_2$*, if* $(\mathbf{x}, \mathbf{y}) \in \mathbf{E_1}$ *then* $(\mathbf{x}, \mathbf{y}) \in \mathbf{E_2}$.

The subgraph of $\mathbb{H}_1$ *induced by a subset* $V \subseteq V_1$ *is denoted by* $\mathbb{H}_1[V]$.

From the motivating scenarios, we can see that we are not interested in arbitrary subgraphs of a graph $\mathbb{H}$, but rather in subgraphs that are *connected* (as noted in [8]). For example, the subassemblies of an assembled product are always considered to be connected. Also, each gene forms a convex interval within the entire graph that represents the genetic sequence. Within graphs, we can formalize *connectedness* as follows:

Definition 3 *Let* $\mathbb{H} = \langle V, \mathbf{E} \rangle$ *be a simple graph.*
$\mathbb{H}$ *is a path iff* $V = \mathbf{x_1}, ..., \mathbf{x_n}$ *and* $\mathbf{E} = \{(\mathbf{x_0}, \mathbf{x_1}), ..., (\mathbf{x_{n-1}}, \mathbf{x_n})\}$.
$\mathbb{H}$ *is connected iff for any two vertices* $\mathbf{x}, \mathbf{y} \in V$*, there exists an induced subgraph that is a path containing* $\mathbf{x}, \mathbf{y}$.

The motivating scenarios therefore lead us to focus on the set of connected induced subgraphs of the simple graph $\mathbb{H}$. It is easy to see that the induced subgraph relation is a partial ordering on the set of all connected induced subgraphs of $\mathbb{H}$, in which elements of V are atoms (since they cannot have any nontrivial subgraphs). For example, in Figure 1, the induced connected subgraph corresponding to the subassembly $\{\mathbf{a}, \mathbf{c}\}$ is contained in the connected induced subgraphs $\{\mathbf{a}, \mathbf{b}, \mathbf{c}\}, \{\mathbf{a}, \mathbf{c}, \mathbf{d}\}$. We can also specify a topology on the connected induced subgraphs of $\mathbb{H}$. For example, in Figure 2, the subassemblies $\{\mathbf{a}, \mathbf{b}\}$ and $\{\mathbf{c}, \mathbf{d}\}$ are disjoint as subgraphs, but are connected because $(\mathbf{b}, \mathbf{c}), (\mathbf{a}, \mathbf{d}) \in \mathbf{E}$.

Definition 4 *Let* $\mathbb{H} = \langle V, \mathbf{E} \rangle$ *be a simple graph, and suppose*

- $\mathscr{C}(\mathbb{H}) = \{\mathbb{J} : \mathbb{J} \preceq \mathbb{H}, \ \mathbb{J} \text{ connected}\}$*;*
- $\mathscr{E}(\mathbb{H}) \subseteq \mathscr{C}(\mathbb{H}) \times \mathscr{C}(\mathbb{H})$ *such that* $(\mathbb{H}_1, \mathbb{H}_2) \in \mathscr{E}(\mathbb{H})$ *iff* $(V_1 \times V_2) \cap \mathbf{E} \neq \emptyset$.

The connected induced subgraph structure of $\mathbb{H}$ *is* $\mathbb{C}^{\mathbb{H}} = \langle \mathscr{C}(\mathbb{H}), \preceq, \mathscr{E}(\mathbb{H}) \rangle$.

We can now specify the fundamental semantic requirement for the mereology we seek:

The models of the mereology must be representable by the following class of structures: $\mathfrak{C} = \{\mathbb{C}^{\mathbb{H}} : \mathbb{H} \in \mathfrak{M}^{simple\_graph}\}$.

The next challenge is to find an axiomatization that satisfies this requirement.

5. Using Mereotopologies for Connected Substructures

We can use theories from the $\mathbb{H}^{combined\_mereotopology}$ Hierarchy[3] of the Common Logic Ontology REpository (COLORE)[4] to characterize the mereologies that we are seeking. In this section, we seek a deeper understanding of the properties of the connected induced subgraph containment ordering $\mathbb{C}^{\mathbb{H}}$ for a simple graph $\mathbb{H}$. Using these properties, we define the class of intended models for the mereotopologies of connected subgraphs that can be represented by such a containment ordering. Finally, we identify a theory in the $\mathbb{H}^{combined\_mereotopology}$ Hierarchy of COLORE that axiomatizes this class of structures.

5.1. Properties of $\mathbb{C}^{\mathbb{H}}$

The crucial insight is that the set of connected induced subgraphs of a simple graph $\mathbb{H}$ can be characterized by the relationships among the subgraphs. It is easy to see how $\mathbb{C}^{\mathbb{H}}$ can be constructed from $\mathbb{H}$ – we simply extract all connected induced subgraphs, and the containment and connection relations are already determined by their relationship to $\mathbb{H}$.

Definition 5 *For each* $\mathbb{K} \in \mathscr{C}(\mathbb{H})$*, the set of subgraphs of the graph* $\mathbb{H}$ *that overlap a given subgraph is denoted by* $\mathscr{O}(\mathbb{K}) = \{\mathbb{J} : \mathbb{K} \cap \mathbb{J} \neq \emptyset,\ \mathbb{J} \in \mathscr{C}(\mathbb{H})\}$

For each $\mathbb{K} \in \mathscr{C}(\mathbb{H})$*, the set of subgraphs of the graph* $\mathbb{H}$ *that are connected a given subgraph is denoted by* $\mathscr{N}(\mathbb{K}) = \{\mathbb{J} : (\mathbf{x},\mathbf{y}) \in E, \mathbf{x} \in V_J, \mathbf{y} \in V_K,\ \mathbb{J} \in \mathscr{C}(\mathbb{H})\}$

The basic operation for constructing connected induced subgraphs is the following:

Definition 6 *Let* $\mathbb{H}_1 = \langle V_1, \mathbf{E_1} \rangle$, $\mathbb{H}_2 = \langle V_2, \mathbf{E_2} \rangle$, $\mathbb{H}_3 = \langle V_3, \mathbf{E_3} \rangle$ *be connected simple graphs.*

$\mathbb{H}_1$ *is the* <u>*sum*</u> *of* $\mathbb{H}_2$ *and* $\mathbb{H}_2$ *(denoted by* $\mathbb{H}_1 = \mathbb{H}_2 + \mathbb{H}_3$*) iff*

- $\mathbb{H}_2 \preceq \mathbb{H}_1$;
- $\mathbb{H}_3 \preceq \mathbb{H}_1$;
- $\mathbb{H}_1 = \mathbb{H}_1[V_2 \cup V_3]$.

The key to the characterization theorem is to identify the properties that the set of subgraphs must satisfy so that we can reconstruct $\mathbb{H}$[5].

Theorem 1 *Suppose* $\mathbb{H}$ *is a simple graph.*

Let $\mathscr{C}$ *be a set of connected induced subgraphs of* $\mathbb{H}$*, and suppose* $\mathscr{E} \subseteq \mathscr{C} \times \mathscr{C}$ *such that* $(\mathbb{H}_1, \mathbb{H}_2) \in \mathscr{E}$ *iff* $(V_1 \times V_2) \cap \mathbf{E} \neq \emptyset$.

$\langle \mathscr{C}, \preceq, \mathscr{E} \rangle$ *is the connected induced subgraph structure of* $\mathbb{H}$ *iff the following conditions are satisfied:*

[3] http://colore.oor.net/combined_mereotopology

[4] http://colore.oor.net/

[5] The full version of this paper, containing proofs for all results, can be found at http://stl.mie.utoronto.ca/publications/full_cisco.pdf

1. *Any two connected induced subgraphs of* $\mathbb{H}$ *are contained in another connected induced subgraph: If* $\mathbb{H}_1, \mathbb{H}_2 \in \mathscr{C}$*, then there exists* $\mathbb{H}_3 \in \mathscr{C}$ *such that*

$$\mathbb{H}_1 \preceq \mathbb{H}_3, \mathbb{H}_2 \preceq \mathbb{H}_3$$

2. *The containment ordering* $\preceq$ *is preserved by the combination of a graph and its subgraphs: If* $\mathbb{H}_1, \mathbb{H}_2 \in \mathscr{C}$*, then*

$$\mathbb{H}_1 \preceq \mathbb{H}_2 \Leftrightarrow \mathbb{H}_2 = \mathbb{H}_1 + \mathbb{H}_2$$

3. *For any two subgraphs of* $\mathbb{H}$ *that are connected to each other, there exists a subgraph of* $\mathbb{H}$ *that is their sum:* $(\mathbb{H}_1, \mathbb{H}_2) \in \mathscr{E}$ *iff there exists* $\mathbb{H}_3 \in \mathscr{E}$ *such that*

$$\mathbb{H}_3 = \mathbb{H}_1 + \mathbb{H}_2$$

4. *The composition of two subgraphs is equal to the mereological and topological sums of the subgraphs: If* $\mathbb{H}_1, \mathbb{H}_2, \mathbb{H}_3 \in \mathscr{C}$*, then*

$$\mathbb{H}_3 = \mathbb{H}_1 + \mathbb{H}_2 \Leftrightarrow \mathscr{O}(\mathbb{H}_3) = \mathscr{O}(\mathbb{H}_1) \cup \mathscr{O}(\mathbb{H}_2)$$
$$\mathbb{H}_3 = \mathbb{H}_1 + \mathbb{H}_2 \Rightarrow \mathscr{N}(\mathbb{H}_3) = \mathscr{N}(\mathbb{H}_1) \cup \mathscr{N}(\mathbb{H}_2)$$

5. *Every connected induced subgraph of* $\mathbb{H}$ *can be decomposed into the sum of a trivial subgraph and a connected induced subgraph: If* $\mathbb{H}_1 \in \mathscr{C}$*, then there exists* $\mathbb{H}_2, \mathbb{H}_3 \in \mathscr{C}$ *such that* $\mathbb{H}_2 \cong K_1$ *and*

$$\mathbb{H}_1 = \mathbb{H}_2 + \mathbb{H}_3$$

A careful inspection of the structures in Figures 1(iii), 2(iii), 3(ii), and 5(iii) reveals that each of them do indeed satisfy the conditions in Theorem 1, so that they are the connected induced subgraph structures for their respective graphs. In other words, we have the right set of structures to use as the basis for a representation theorem.

5.2. *Mereographs for Connected Induced Subgraphs*

What is the mereotopology that axiomatizes the connected induced subgraph structure for a simple graph $\mathbb{H}$?

5.2.1. *Mereographs and the Mereotopology MT*

We follow the work of [10] for the approach to mereotopology in which both parthood and connection are primitive relations. The mereology of the parthood relation is represented by the class of partial orderings $\mathfrak{M}^{partial\_ordering}$, and the connection relation is represented by the class of graphs with loops $\mathfrak{M}^{graph\_loops}$.

Definition 7 *Suppose* $\mathbb{P} \in \mathfrak{M}^{partial\_ordering}$ *such that* $\mathbb{P} = \langle V, \preceq \rangle$.
The upper set for $\mathbf{x}$ *in* $\mathbb{P}$*, denoted by* $U^{\mathbb{P}}(\mathbf{x})$*, is*

$$U^{\mathbb{P}}(\mathbf{x}) = \{\mathbf{y} : \mathbf{x} \leq \mathbf{y}\} \qquad U^{\mathbb{P}}(X) = \bigcup_{\mathbf{x} \in X} U(\mathbf{x})$$

$\mathscr{L}_{\mathbb{P}} = \langle V, E \rangle$ *is the lower bound graph for* $\mathbb{P}$*:* $(\mathbf{x}, \mathbf{y}) \in E \quad L^{\mathbb{P}}[\mathbf{x}] \cap L^{\mathbb{P}}[\mathbf{y}] \neq \emptyset$

Definition 8 *Suppose* $\mathbb{G} \in \mathfrak{M}^{graph\_loops}$*, such that* $\mathbb{G} = \langle V, \mathbf{E} \rangle$*.*
The neighbourhood of $\mathbf{x}$ *in* $\mathbb{G}$*, denoted by* $N^{\mathbb{G}}(\mathbf{x})$*, is*

$$N^{\mathbb{G}}(\mathbf{x}) = \{\mathbf{y} : (\mathbf{x}, \mathbf{y}) \in \mathbf{E}\} \qquad N^{\mathbb{G}}(X) = \bigcup_{\mathbf{x} \in X} N^{\mathbb{G}}(\mathbf{x})$$

A new class of mathematical structures was introduced by [10] to characterize the models of mereotopological theories.

Definition 9 $\mathbb{P} \oplus \mathbb{G} = \langle V, \mathbf{E}, \leq \rangle$ *is a mereograph iff*

1. $\mathbb{P} = \langle V, \leq \rangle$ *such that* $\mathbb{P} \in \mathfrak{M}^{partial\_ordering}$*;*
2. $\mathbb{G} = \langle V, \mathbf{E} \rangle$ *such that* $\mathbb{G} \in \mathfrak{M}^{graph\_loops}$*;*
3. $U^{\mathbb{P}}(N^{\mathbb{G}}(\mathbf{x})) \subseteq N^{\mathbb{G}}(\mathbf{x})$*, for each* $\mathbf{x} \in V$*.*

$\mathfrak{M}^{mereograph}$ *denotes the class of mereographs.*

In other words, a mereograph is the amalgamation of partial orderings and graphs with loops, where Condition (3) constrains how these two structures are related to each other: the neighbourhood of a vertex in the graph is closed under upper sets in the partial ordering. Consequently, *mereographs* are the right class of structures that we need:

Lemma 1 *If* $\mathbb{H}$ *is a simple graph, then* $\mathbb{C}^{\mathbb{H}} = \langle \mathscr{C}(\mathbb{H}), \preceq, \mathscr{E} \rangle$ *is a mereograph.*

The next question is to determine exactly what *class* of mereographs we need.

5.2.2. Connected Induced Subgraph Mereographs

Our goal is to specify the conditions that a mereograph must satisfy if it is to be representable by $\mathbb{C}^{\mathbb{H}}$. The approach we take is to "translate" the properties of $\mathbb{C}^{\mathbb{H}}$ (proven in Theorem 1) into properties of mereographs.

Definition 10 *Suppose*
$\Sigma(\mathbf{x}, \mathbf{y}) = \{\mathbf{z} : N^{\mathscr{L}(\mathbb{P})}[\mathbf{z}] = N^{\mathscr{L}(\mathbb{P})}[\mathbf{x}] \cup N^{\mathscr{L}(\mathbb{P})}[\mathbf{y}]\}$
$\Sigma^{-1}(\mathbf{x}) = \{(\mathbf{y}, \mathbf{z}) : \Sigma(\mathbf{y}, \mathbf{z}) = \mathbf{x}\}$
$\Pi(\mathbf{x}, \mathbf{y}) = \{\mathbf{z} : N^{\mathbb{G}}[\mathbf{z}] = N^{\mathbb{G}}[\mathbf{x}] \cup N^{\mathbb{G}}[\mathbf{y}]\}$

A mereograph $\mathbb{P} \oplus \mathbb{G} = \langle V, \mathbf{E}, \leq \rangle$ *is a connected induced subgraph mereograph iff*

1. $U^{\mathbb{P}}[\mathbf{x}] \cap U^{\mathbb{P}}[\mathbf{y}] \neq \emptyset$*;*
2. $U^{\mathbb{P}}[\mathbf{x}] \subseteq U^{\mathbb{P}}[\mathbf{y}]$ *iff* $\mathscr{L}^{\mathbb{P}}(\mathbf{x}) \subseteq \mathscr{L}^{\mathbb{P}}(\mathbf{y})$*;*
3. $\mathbf{y} \in N^{\mathbb{G}}[\mathbf{x}]$ *iff* $\Sigma(\mathbf{x}, \mathbf{y}) \neq \emptyset$ *for any* $\mathbf{x}, \mathbf{y} \in V$*;*
4. $\Sigma(\mathbf{x}, \mathbf{y}) \subseteq \Pi(\mathbf{x}, \mathbf{y})$*, for any* $\mathbf{x}, \mathbf{y} \in V$*;*
5. $\Sigma^{-1}(\mathbf{x}) \subseteq (\mathscr{A}(\mathbb{P}) \times V)$*, where* $\mathscr{A}(\mathbb{P})$ *is the set of atoms in* $\mathbb{P}$*.*

$\mathfrak{M}^{cisco\_mt}$ *denotes the class of connected induced subgraph mereographs*[6]*.*

[6]The name "cisco" is an acronym for "**c**onnected **i**nduced **s**ubgraph **c**ontainment **o**rdering."

5.2.3. Representation Theorems for Connected Induced Subgraphs

Now that we have defined the class of connected induced subgraph mereographs, we can demonstrate that they are the correct set of structures by proving that they are indeed representable by the connected induced subgraph structures of simple graphs.

Theorem 2 *There is a bijection* $\varphi : \mathfrak{M}^{cisco\_mt} \rightarrow \mathfrak{C}$ *and an isomorphism*

$$\mu : \mathbb{P} \oplus \mathbb{G} \rightarrow \varphi(\mathbb{P} \oplus \mathbb{G})$$

This constitutes the *representation theorem* for connected induced subgraph mereographs. As such, it also provides a validation of this class of mereographs as the right class of structures to be the intended models of the axiomatization of the mereotopology, which we now consider.

5.3. Connected Sums of Subgraphs

The next step is to provide an axiomatization of the class of connected induced subgraph mereographs. Before moving to the axiomatization of the mereotopology for connected induced subgraphs, we briefly prove some properties of sums and the relationship to the theory $T_{em\_mereology}$ (Strong Supplementation) that we will need later in the paper.

5.3.1. Strong Supplementation and Properties of Sums

Classical mereology $T_{cm\_mereology}$[7] entails the existence of the sum of any two underlapping elements. Although we are seeking a logical theory that is weaker than $T_{cm\_mereology}$, we still need to adopt a definition for mereological sum. In classical mereology, there are actually different axiomatizations for the mereological sum of two elements [11], but we adopt the following:

Definition 11 T_{sumdef} *is the definitional extension of* $T_{m\_mereology}$ *with the sentence* [8]*:*

$$(\forall x \forall y \forall z\, (sum(x,y,z) \equiv (\forall u\, (overlaps(u,z) \equiv (overlaps(u,x) \vee overlaps(u,y)))))) \quad (1)$$

However, in the absence of the $T_{cm\_mereology}$, we cannot guarantee basic properties of sums, such as functionality or its relationship to parthood. In fact, there is a close relationship between the Strong Supplementation Principle (axiomatized by $T_{em\_mereology}$[9]), the required properties of mereological sums, and weaker supplementation principles.

Proposition 1 *Let* $T_{sumpart}$ *be the extension of* T_{sumdef} *with the sentence* [10]*:*

$$(\forall x \forall y \forall z\, (sum(x,y,z) \supset part(x,z))) \quad (2)$$

$T_{em\_mereology}$ *is logically equivalent to* $T_{mm\_mereology}$[11] $\cup T_{sumpart}$.

Proof: http://colore.oor.net/mereology/theorems/sumpart/ □

[7] http://colore.oor.net/mereology/cm_mereology.clif
[8] http://colore.oor.net/mereology/definitions/sum.clif
[9] http://colore.oor.net/mereology/em_mereology.clif
[10] http://colore.oor.net/mereology/sumpart.clif
[11] Weak Supplementation Principle: http://colore.oor.net/mereology/mm_mereology.clif

Proposition 2 *Let T_{sumfun} be the extension of T_{sumdef} with the sentence* [12]*:*

$$(\forall x \forall y \forall z \forall u\,(sum(x,y,z) \wedge sum(x,y,u) \supset (z=u))) \tag{3}$$

$T_{em\_mereology}$ is logically equivalent to $T_{ppp\_mm\_mereology}$[13] *$\cup T_{sumfun}$.*

Proof: http://colore.oor.net/mereology/theorems/sumfun/ □

Finally, we can show that the Strong Supplementation Principles is equivalent to the relationship between parthood and sums which corresponds to Condition (2) in Theorem 1:

Proposition 3 *$T_{em\_mereology}$ is logically equivalent*[14] *to the extension of T_{sumdef} by:*

$$(\forall x \forall y\,(part(x,y) \equiv sum(x,y,y))) \tag{4}$$

Proof: http://colore.oor.net/mereology/theorems/emsum/ □

This is interesting in the light of Fine's notion of *operationalism*, in which the parthood relation is defined in terms of composition operations like sums, rather than defining sums in terms of the parthood relation.

5.3.2. Mereotopology of Connected Sums

The mereotopology we have been pursuing can be obtained by a rather straightforward axiomatization[15] of the conditions in the definition of $\mathfrak{M}^{cisco\_mt}$.

Definition 12 *T_{cisco_mt} is the extension*[16] *of $T_{mt} \cup T_{em\_mereology} \cup T_{ub\_mereology} \cup T_{sumdef}$*

$$(\forall x \forall y\,(C(x,y) \equiv (\exists z\,(sum(x,y,z))))) \tag{5}$$

$$(\forall x \forall y \forall z\,(sum(x,y,z) \supset (\forall u\,(C(u,z) \equiv (C(u,x) \vee C(u,y)))))) \tag{6}$$

$$(\forall x \exists y \exists z\,(atom(y) \wedge sum(y,z,x))) \tag{7}$$

$T_{ub\_mereology}$ corresponds to Condition (1) in Definition 10, and by Proposition 3, $T_{em\_mereology}$ corresponds to Condition (2) in Definition 10. The remaining axioms in T_{cisco_mt} correspond to Conditions (3) to (5), respectively, in Definition 10.

Formalizing these correspondences gives us the following result, which is the verification of T_{cisco_mt}, and shows that we have the correct set of axioms:

Theorem 3 *There exists a bijection $\varphi : Mod(T_{cisco\_mt}) \rightarrow \mathfrak{M}^{cisco\_mt}$ such that*

1. $\langle \mathbf{x},\mathbf{y} \rangle \in \mathbf{C}^{\mathscr{M}}$ *iff* $\mathbf{y} \in N^{\mathbb{G}}[\mathbf{x}]$;
2. $\langle \mathbf{x},\mathbf{y} \rangle \in \mathbf{part}^{\mathscr{M}}$ *iff* $\mathbf{x} \in L^{\mathbb{P}}[\mathbf{y}]$.
3. $\langle \mathbf{x},\mathbf{y},\mathbf{z} \rangle \in \mathbf{sum}^{\mathscr{M}}$ *iff* $\Sigma(\mathbf{x},\mathbf{y}) = \{\mathbf{z}\}$;

---

[12] http://colore.oor.net/mereology/sumfun.clif

[13] Proper Parts Principle: http://colore.oor.net/mereology/ppp_mm_mereology.clif

[14] In [11], Varzi shows that $T_{em\_mereology}$ entails the sentence in Proposition 3, but does not establish the equivalence.

[15] Axiom (5) is equivalent to Simons' [8] combination of axiom (TID8) from Tiles and (WD5) from Whitehead.

[16] http://colore.oor.net/combined_mereotopology/cisco_mt.clif

6. Weak Mereotopology of Connected Substructures

If we revisit the motivating scenario for molecular structure, then it appears that T_{cisco_mt} is too strong. In Figure 4, the functional groups $\mathbf{g_2}, \mathbf{g_3}$ are attached, yet there is no skeleton that is the sum of these groups, and hence we do not have a connected induced subgraph mereograph. On the other hand, the remaining conditions in Definition 10 are satisfied. We therefore consider a *generalization* of the class of a connected induced subgraph mereographs that can be used to capture mereotopologies such as the one in Figure 4. In particular, Condition (3) in Definition 10 is weakened to allow the existence of connected elements that do not have a sum:

Definition 13 *A mereograph $\mathbb{P} \oplus \mathbb{G} = \langle V, \mathbf{E}, \leq \rangle$ is a* *self-connected induced subgraph mereograph iff*

1. $U^{\mathbb{P}}[\mathbf{x}] \cap U^{\mathbb{P}}[\mathbf{y}] \neq \emptyset$;
2. $U^{\mathbb{P}}[\mathbf{x}] \subseteq U^{\mathbb{P}}[\mathbf{y}]$ *iff* $\mathscr{L}^{\mathbb{P}}(\mathbf{x}) \subseteq \mathscr{L}^{\mathbb{P}}(\mathbf{y})$;
3. $\Sigma(\mathbf{x}, \mathbf{y}) \neq \emptyset$ *implies* $\mathbf{y} \in N^{\mathbb{G}}[\mathbf{x}]$, *for any* $\mathbf{x}, \mathbf{y} \in V$;
4. $\Sigma(\mathbf{x}, \mathbf{y}) \subseteq \Pi(\mathbf{x}, \mathbf{y})$, *for any* $\mathbf{x}, \mathbf{y} \in V$;
5. $\Sigma^{-1}(\mathbf{x}) \subseteq (\mathscr{A}(\mathbb{P}) \times V)$.

$\mathfrak{M}^{weak\_cisco\_mt}$ *denotes the class of self-connected induced subgraph mereographs.*

Since this is a generalization of the class of connected induced subgraph mereographs, we can also obtain a generalization of the representation theorem:

Theorem 4 *There is a bijection* $\varphi : \mathfrak{M}^{weak\_cisco\_mt} \to \mathfrak{C}$ *and a monomorphism*

$$\mu : \mathbb{P} \oplus \mathbb{G} \to \varphi(\mathbb{P} \oplus \mathbb{G})$$

that fixes $\mathscr{A}(\mathbb{P})$.

The key difference between Theorem 2 and Theorem 4 is that, because self-connected induced subgraph mereographs allow connected elements that do not have sums, they need not be isomorphic to a connected induced subgraph structure $\mathbb{C}^{\mathbb{H}}$. However, there still needs to be a one-to-one correspondence between the vertices of the graph $\mathbb{H}$ and the atoms of the partial ordering in the mereograph. Recalling the structure in Figure 4, we can see that it is monomorphic to the connected induced subgraph structure for the path graph P_4.

The axiomatization of the mereotopology that we need for motivating scenarios such as molecular structure in MoSt therefore only requires that sums are connected. The following result shows that this set of axioms does indeed provide the correct axiomatization of the class of self-connected induced subgraph mereographs:

Theorem 5 *Let $T_{weak\_cisco\_mt}$ be the extension of $T_{mt} \cup T_{em\_mereology} \cup T_{ub\_mereology} \cup T_{sumdef}$ with the following sentences*[17]*:*

$$(\forall x \forall y \forall z\, (sum(x,y,z) \supset (\forall u\, (C(u,z) \equiv (C(u,x) \vee C(u,y)))))) \tag{8}$$

$$(\forall x \exists y \exists z\, (atom(y) \wedge sum(y,z,x))) \tag{9}$$

$$(\forall x \forall y\, (sum(x,y,z) \supset C(x,y))) \tag{10}$$

[17] http://colore.oor.net/combined_mereotopology/weak_cisco_mt.clif

There exists a bijection $\varphi : Mod(T_{weak\_cisco\_mt}) \rightarrow \mathfrak{M}^{weak\_cisco\_mt}$ *such that*

1. $(\mathbf{x},\mathbf{y}) \in \mathbf{C}^{\mathscr{M}}$ *iff* $\mathbf{y} \in N^{\mathbb{G}}[\mathbf{x}]$;
2. $(\mathbf{x},\mathbf{y}) \in \mathbf{part}^{\mathscr{M}}$ *iff* $\mathbf{x} \in L^{\mathbb{P}}[\mathbf{y}]$.

$T_{weak\_cisco\_mt}$ is closely related to the notion of self-connected objects:

Proposition 4 *If* T_{scdef} *is the extension of* $T_{emt} \cup T_{sumdef}$ *with*

$$(\forall x\,(SC(x) \equiv (\forall y \forall z\,(sum(y,z,x) \supset C(y,z))))) \tag{11}$$

then $T_{weak\_cisco\_mt} \cup T_{scdef} \models (\forall x)\,SC(x)$.

7. Mereology of Connected Subgraphs of a Graph

Up to this point, we have characterized the mereotopology that corresponds to the connected induced subgraph containment ordering for a graph. However, the axiomatization in T_{cisco_mt} uses the combined signature of mereology and topology. In this section, we identify an axiomatization of the connected induced subgraph containment ordering using only the signature of mereology, and show that such an axiomatization forms a module of T_{cisco_mt}. This allows us to speak of the mereology of connected subgraphs of a graph.

7.1. Subgraph Containment Lattices

Remarkably, the problem of characterizing the class of posets which are isomorphic to the connected induced subgraph containment ordering of a graph has been posed and solved within the mathematics community [12–15]. If we re-examine the mereologies for the motivating scenarios, we see that they satisfy the following definition:

Definition 14 *A partial ordering* $\mathbb{P} = \langle V, \leq \rangle$ *is* <u>*properly semimodular*</u> *iff*

1. $\mathbb{P}$ *is atom-height, that is, the cardinality of all maximal chains in* $\mathbb{P}$ *is equal to the cardinality of the set of atoms in* $\mathbb{P}$;
2. *for each* $\mathbf{x} \in V$, $\langle U^{\mathbb{P}}[\mathbf{x}], \leq \rangle$ *is an upper semimodular lattice:*
   (a) *any two elements* $\mathbf{y},\mathbf{z}$ *have a least upper bound and a greatest lower bound in* $U^{\mathbb{P}}[\mathbf{x}]$;
   (b) *if* $\mathbf{z}$ *covers the greatest lower bound of* $\mathbf{z}$ *and* $\mathbf{y}$, *then the least upper bound of* $\mathbf{z}$ *and* $\mathbf{y}$ *covers* $\mathbf{y}$.

$\mathfrak{M}^{proper\_semimodular}$ *denotes the class of properly semimodular partial orderings.*

The central theorem shows that the class of properly semimodular partial orderings is equivalent to the connected induced subgraph ordering for a simple graph $\mathbb{H}$:

Theorem 6 *Let* $\mathbb{H} = \langle V, \mathbf{E} \rangle$ *be a simple graph, and let* $\mathbb{P} = \langle V, \leq \rangle$ *be a partial ordering.*

$$\mathbb{P} \cong \langle \mathscr{C}(\mathbb{H}), \preceq \rangle \quad \textit{iff} \quad \mathbb{P} \in \mathfrak{M}^{proper\_semimodular}$$

Moreover, this suggests that we can axiomatize the class of connected induced subgraph orderings by axiomatizing the class of properly semimodular partial orderings.

7.2. Introducing T_{cisco}

Using Theorem 6, we can specify a logical theory within the $\mathbb{H}^{mereology}$ Hierarchy of COLORE that is logically synonymous with the class of properly upper semimodular partial orderings.

Theorem 7 *Let T_{cisco} be the extension of $T_{em\_mereology}$ with the sentences*[18]*:*

$$(\forall u \forall x\,(ppart(u,x) \supset (\exists y\,(atom(y) \wedge part(y,x))))) \tag{12}$$

$$(\forall x \forall y\,(covers(x,y) \supset (\exists z\,(atom(z) \wedge ppart(z,x) \wedge \neg part(z,y))))) \tag{13}$$

$$\begin{aligned}(\forall x \forall y \forall z \forall u\,((covers(x,y) \wedge atom(z) \wedge ppart(z,x) \wedge \neg part(z,y)\\ \wedge atom(u) \wedge ppart(u,x) \wedge \neg part(u,y)) \supset (z = u)))\end{aligned} \tag{14}$$

$$\begin{aligned}&(\forall x \forall a \forall b\,((part(x,a) \wedge part(x,b))\\ &\supset (\exists z\,(part(x,z) \wedge (\forall u\,(part(z,u) \equiv (part(a,u) \wedge part(b,u))))))))\end{aligned} \tag{15}$$

$$\begin{aligned}&(\forall x \forall a \forall b\,((part(x,a) \wedge part(x,b))\\ &\supset (\exists z\,(part(x,z) \wedge (\forall u\,((part(u,z) \equiv (part(u,a) \wedge part(u,b)))))))))\end{aligned} \tag{16}$$

$$\begin{aligned}&(\forall p \forall x \forall y\,((atom(p) \wedge part(x,y) \wedge \neg part(p,y))\\ &\supset (\exists z\,(part(x,z) \wedge part(p,z) \wedge part(y,z) \wedge covers(z,y)))))\end{aligned} \tag{17}$$

There exists a bijection $\varphi : Mod(T_{cisco}) \rightarrow \mathfrak{M}^{proper\_semimodular}$ such that

$$(\mathbf{x},\mathbf{y}) \in \mathbf{part}^{\mathscr{M}} \quad \textit{iff} \quad \mathbf{x} \in L^{\mathbb{P}}[\mathbf{y}]$$

Axioms 12, 13, and 14 guarantee that the mereology is atom-height (condition (1) of Definition 14). Axioms 15, 16, and 17 guarantee that the upper set of each element in the mereology is an upper semimodular lattice (Condition (2) of Definition 14).

Theorem 8 *For any $\mathbb{P} \in \mathfrak{M}^{cisco}$ there exists a unique $\mathbb{G} \in \mathfrak{M}^{graph\_loops}$ such that*

$$\mathbb{P} \oplus \mathbb{G} \in \mathfrak{M}^{cisco\_mt}$$

This result shows that T_{cisco_mt} is a conservative extension of T_{cisco}. Consequently, T_{cisco} is indeed the mereology we seek – a new nonclassical mereology that applies to the classes of objects seen in the motivating scenarios of Section 2.

8. Summary

We began this paper with the observation that classical mereology is not appropriate for certain classes of objects, such as assemblies, convex time intervals, molecules, gene sequences, because sums do not exist for every pair of such elements. This launched the quest for exactly what mereology corresponds to the parthood relation for such objects.

[18] http://colore.oor.net/mereology/cisco.clif

A key insight is that the underlying structure that specifies an object in each of the motivating scenarios is a graph, and all parts of the object correspond to connected induced subgraphs of that graph. We therefore introduced the parthood and connection structure on the set of connected induced subgraphs of a graph and used this as the basis of the representation theorem for a new mereotopology, T_{cisco_mt}. Finally, we specified the axiomatization of the nonclassical mereology T_{cisco}, which is the mereology that is conservatively extended by T_{cisco_mt}.

In the mereotopology T_{cisco_mt}, the sum of two elements exists iff they are connected. We also introduced a weaker mereotopology $T_{weak\_cisco\_mt}$ in which not all connected elements have sums, although elements for which sums do exist must be connected. The characterization of the mereology that is the module of $T_{weak\_cisco\_mt}$ remains an open question.

An additional area for future work is to explore the extensions of T_{cisco} that correspond to special classes of graphs. For example, in the cases of convex time intervals and gene sequences, the underlying graph is a path graph, and the resulting mereology corresponds to a special class of lattices.

References

[1] Ru Y, Grüninger M. Parts Unknown: Mereologies for Solid Physical Objects. In: Proceedings of the Joint Ontology Workshops 2017. vol. 2050. CEUR-WS.org; 2017. p. 1–9.
[2] Fine K. Towards a Theory of Part. Journal of Philosophy. 2010;107:559–589.
[3] van Benthem J. Logic of Time. Springer Verlag; 1983.
[4] Chui C, Grüninger M. A Molecular Structure Ontology for Medicinal Chemistry. In: Ferrario R, Kuhn W, editors. Formal Ontology in Information Systems - Proceedings of the 9th International Conference, FOIS 2016, Annecy, France, July 6-9, 2016. IOS Press; 2016. p. 285–298.
[5] Hovda P. What is Classical Mereology? J Philos Logic. 2009;38:55–82.
[6] Pietruszczak A. A General Concept of Being a Part of a Whole. Notre Dame Journal of Formal Logic. 2014;55:359–381.
[7] Sider T. Nothing over and above. Grazer Philosophische Studien. 1989;91:191–216.
[8] Simons P. Parts: A Study in Ontology. Oxford University Press, USA; 1987.
[9] Casati R, Varzi AC. Parts and Places: The Structures of Spatial Representation. MIT CogNet; 1999.
[10] Grüninger M, Aameri B. A New Perspective on the Mereotopology of RCC8. In: Proceeding of the 13th International Conference on Spatial Information Theory (COSIT 2017). vol. 86. Schloss Dagstuhl - Leibniz-Zentrum für Informatik; 2017. p. 2:1–2:13.
[11] Varzi C. Formal Theories of Parthood. In: Calosi C, Graziani P, editors. Mereology and the Sciences: Parts and Wholes in the Contemporary Scientific Context; 2014. p. 359–378.
[12] Niemenen J. The Lattice of Connected Subgraphs of a Connected Graph. Annales Societatis Mathematicae Polonae. 1979;21(1):187–193.
[13] Klavžar S, Petkovšek M, Steelman JH. E3281: The Lattice of Connected Subgraphs of a Graph. The American Mathematical Monthly. 1990;97(10):924–925.
[14] Kézdy A, Seif S. When is a Poset Isomorphic to the Poset of Connected Induced Subgraphs of a Graph? Southwest J Pure Appl Math. 1996;1:42–50.
[15] Smith J. The Poset of Graphs Ordered by Induced Containment. J Comb Theory Ser A. 2019;168:348–373.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200671

Pluralities, Collectives, and Composites

Claudio MASOLO [a], Laure VIEU [b,a], Roberta FERRARIO [a], Stefano BORGO [a], Daniele PORELLO [a]

[a] *Laboratory for Applied Ontology (ISTC-CNR)*
[b] *IRIT, CNRS, Toulouse University*

Abstract. Forests, cars and orchestras are very different ontological entities, and yet very similar in some aspects. The relationships they have with the elements they are composed of is often assumed to be reducible to standard ontological relations, like parthood and constitution, but how this could be done is still debated. This paper sheds light on the issue starting from a linguistic and philosophical analysis aimed at understanding notions like plurality, collective and composite, and proposing a formal approach to characterise them. We conclude the presentation with a discussion and analysis of social groups within this framework.

Keywords. mereology, parthood, constitution, membership, plural, plurality, collective, composition, composite, social group, agency, functionality.

1. Introduction

In most domains, applied ontology is bound to deal with groups or collections—roughly, entities such as orchestras, herds, forests or decks of cards that can be said to have members and that we here call *collectives*. However, only few formal ontology theories attempt to deeply characterise collectives [1,2,3]. Focusing on objects (aka endurants or continuants), we propose a new account of collectives contrasting these objects with those that we call *composites* (aka assemblies [4] or complexes [5]), i.e., objects such as animals, trees, mountains or cars that have another sort of internal structure.

The linguistic literature on plurals and group (or collection) nouns is quite large in contrast. To account for their semantics, formal approaches have taken two main paths: one based on mereology following Link [6,7] and the other based on plural quantification following Boolos [8]. The latter develops plural logics intended to avoid the commitment to the existence of referents for plurals and group nouns on top of the referents of their members [9]. In this paper we aim to develop a view within the traditional first-order framework, especially to make the proposal readily available to existing ontological systems. Moreover, we favour ontological precision over parsimony when both are inconciliable and so are prepared to adopt multiplicativism [10] when necessary. We will therefore rather take inspiration from mereological approaches.

A main issue in the literature on plurals is how to account for predication over plural nouns (see among many others [6,9,11,12]). Consider for instance plural noun phrases like ‘the students’ or ‘Alice and Bob’. These refer to several things at once, i.e., to what is often called *pluralities*. Sometimes, one may reduce predication on plurals to independent claims for each element in the plurality: ‘Alice and Bob are students’ boils down to

'Alice is a student' and 'Bob is a student', this is called *distributivity*. However, distributivity is blocked in expressions involving collective predicates like 'the students collaborate in the project', 'the cards are scattered on the floor' and 'all the workers met in the cafeteria'. Formally speaking, such examples of non-distributivity motivate the introduction of pluralities in the domain of quantification (or the use of plural quantifiers).

Plurals and singular group nouns are often accounted for without distinction in this literature, especially in the plural logics approach. Indeed, collective predicates apply to group nouns as well, viz. 'the deck (of cards) is scattered on the floor' and 'the committee met in the cafeteria'. Yet, some work in the mereology-based literature on group nouns [13] emphasised the need for distinguishing between the referents of group nouns and of plurals, that is, between collectives and pluralities. In contrast to pluralities, collectives manifest some sort of unity. It is not enough to have a plurality of musicians to have an orchestra: they need to play together or be bound to do so. Not any plurality of trees form a forest, they need to satisfy spatial constraints (and perhaps manifest ecological interrelationships). In addition, group nouns (and so kinds of collectives) usually convey the kind that characterises the collective's *members*: an orchestra is a collective of musicians, a forest a collective of trees, and one can go on with pairs like army-soldiers, herd-animals, deck-cards. Collectives thus display a homogeneity among members that plurals given in extension (with the conjunction 'and') as in 'Alice and her cat' do not possess. We will then distinguish three sorts of objects: *pluralities*, *collectives*, and *composites*, focusing on their internal structure and on the relationships between them.

Barker [13], expanding Link's proposal [6], showed how collectives are related to pluralities by some sort of *constitution*. For instance, if Alice and Bob are the members of a duet, the duet (a collective) is constituted by the plurality 'Alice and Bob'. Considering how constitution relates them throws further light on the differences between pluralities and collectives. The same plurality can constitute several collectives, even at the same time: an orchestra and a soccer team may have exactly the same members. Moreover, a plurality is directly bound to each of its members, and so it cannot change members without this changing the plurality itself. One can even introduce pluralities of objects existing at disjoint periods, as with 'Bach and Mozart'. In contrast, a collective can change members and so may be constituted at different times by different pluralities (of objects all existing at those times). Finally, while we may have collectives of collectives (e.g., federations of sport clubs), pluralities of pluralities are usually rejected (but see [14]), one reason why the literature on plurals embraced mereology rather than set theory. We also have pluralities of collectives: two orchestras are neither simply a plurality of musicians nor a larger single orchestra, just as 'two pairs of shoes' and 'four shoes' mean different things, a pair of shoes having its own unity.

The literature on constitution [15] is more concerned with composites than with collectives, and focused on what distinguishes composites from amounts of matter, often calling for notions such as form or function. Differences between composites and collectives are rarely addressed. The formal semantics literature above is also largely focused on comparing plurals and group nouns with mass nouns such as 'water', 'sand' or 'furniture' and little is said on how plurals or group nouns compare with count nouns referring to composites. We thus review here additional intuitions taken from various works that we take as starting point for developing our proposal.

Each composite has parts, that we will call *components*. Just as above with collectives, a composite differs from the plurality of its components. The classical literature on constitution argues at length that different pluralities of components may form a composite (say, my car) at different times, since some of its components (say, a tire) can change across time. Moreover, my car can be dismantled. The plurality of its components, scattered, does survive, while there would be no car there. So, as for collectives, we hold that there is a sense in which the composite is constituted by the plurality of its components.

Regarding the difference between composites and collectives, we know that membership is not transitive [16,17,18], while componenthood is (albeit with some caveats [16,19]). Moreover, as we saw, the members of a collective are associated with a kind, thus in principle one can unambiguously count such members, let them be the musicians in an orchestra, the trees in a forest or the cards in a deck. On the contrary, trying to count the components of a composite may lead to several answers as it might be unclear what to count: the number of components of a lawn mower depends on the choice of a decomposition method (functional, structural, topological) and of the adopted level of granularity. A formal model able to capture the distinctions between pluralities, collectives and composites could prove helpful in representing information in such diverse scenarios as industrial plants, museums, galleries of art, systems of mechanical artefacts etc.

These concepts may also play an important role in the domain of social reality. The literature on metaphysics of social groups primarily addressed two questions: what kind of relationship holds between a social group and its members? What kind of entities (sums of individuals, sets of individuals, roles, etc.) is this relation connecting with social groups? Concerning the first question, there are in general two types of approaches, the former supporting some sort of mereological relation (for instance [20,21]) and the latter some sort of constitution relation ([22,23,24,25,26,27]). In the '80s Ruben [28,29] noted that there are three main properties of group membership that a metaphysical theory should represent: the persistence of groups through changes of their members; the existence of co-extensional but numerically distinct groups; and the non-transitivity of the relation between groups and members (properties already examined above). Concerning the second question, the view of groups as artefacts or organisms has been used as metaphor when addressing particular cases. But, as far as we know, the distinction of kinds of groups in terms of collectives and composites has been overlooked so far.

We propose a formal representation that captures the results of an analysis of the notions of plurality, collective and composite, and to test and apply it to the case of social groups and organisations. Although we illustrate our work on this specific application domain, we aim at generality, considering all sorts of objects, be they artefacts or natural objects, be they agentive or not. This will allow us to focus on the intrinsic structure of collectives and composites, and to refrain from calling for extrinsic properties such as function and agency that have been very often emphasised when characterising the relationships between a component and a composite or between a member and a collective. Although some argue in favour of a notion of function encompassing both artefacts and biological entities [30], we would not defend a function-based approach to composites as it would exclude from the start entities like mountains composed of peaks and valleys. Similarly, although many collectives are actually group agents, founding collectives on agency would exclude from the start forests and decks of cards.

The paper is organised as follows. Sect. 2 presents Link's proposal [6] as expanded by Barker [13], where collectives are related to pluralities by some sort of constitution, a starting point for our account. This general idea is sketched in Sect. 3 and is formally characterised in Sect. 4. Sect. 5 illustrates our account in the domain of social reality.

2. Pluralities and collectives in Link's and Barker's proposals

In 1983 Link proposed a formal account ([6], reprinted in [7]) of collective predication, as in 'The children gather around their teachers', in analogy to predication involving mass nouns, like 'The water gathers in big pools.' Link's formal theory is rich and articulated. For what concerns us in this paper, we need only a fragment of his approach as presented below (symbols and terminology are our own), and as enhanced by Barker [13].

Link assumes the existence of two sets of entities, the set O of (material) *objects* and the set A of *amounts of matter* (with $A \subseteq O$), and two order relations, $\leq$ holding among objects and $\circledleq$ holding among amounts of matter, with $(O, \leq)$ and $(A, \circledleq)$ being join-semilattices. In a mereological perspective, join-semilattices can be characterised by imposing on $\leq$ and $\circledleq$ the axioms of classical extensional mereology with unrestricted sum operators, see [31]. In Link's approach, $\leq$ (but not necessarily $\circledleq$) is atomic and $\leq$-atoms are taken as referents of singular nouns, while non-$\leq$-atomic objects, that Link calls *plural objects*—our pluralities, are taken as referents of plural nouns. Link assumes that all amounts of matter are $\leq$-atoms.

Link adds a constitution function h relating any object to the amount of matter making it up, where h restricted to amounts of matter is identity, and with h preserving order structures, i.e, mapping $\leq$ onto $\circledleq$. Such a coupled double mereological structure enables Link to account for a significant number of linguistic phenomena involving plurals and mass nouns, clarifying their similarities and their differences.

Link makes explicit that collectives are to be distinguished from the pluralities of their members, collectives being $\leq$-atoms, and Barker [13] further enhances his proposal by adding a second constitution function f over objects to relate collectives (called *groups* by Barker) to the pluralities making them up.

3. Our approach in a nutshell

Our account of pluralities, collectives, and composites formalised as a first-order theory in Sect. 4 builds on Link's and Barker's work, exploiting an atomic classical mereology to characterise pluralities. We will rephrase the constitution function f between collectives and pluralities as a temporalised relation of constitution $\lhd$, since, as seen in Sect. 2, several arguments classically involved in constitution studies and based on change across time are used to distinguish these sorts of entities.

Since we are also interested here in social entities which do not have a clear (material) substrate, a second move is concerned with avoiding the commitment to a substrate when characterising the notions of plurality and collective. Note that this move does not prevent the integration of our proposal with foundational ontologies exploiting substrates (for instance, DOLCE [32] uses a constitution relation in the very same spirit of Link's function h). We leave such integration as future work but we point out in the following a possible enhancement of the axiomatisation based on the existence of a substrate; see the discussion on axiom (a11) below.

Finally, we introduce a second temporalised relation of constitution $\prec$ that links the composites to the pluralities of their components. As for collectives and pluralities, our characterisation of composites does not rely on the existence of a (material) substrate.

The strategy adopted, fully described in Sect. 4, is to account for the diversity of parthood relations on the basis of a diversity of constitution relations (namely $\lhd$ and $\prec$), exploiting a single mereological relation $\leq$. Importantly, $\leq$ is not meant to be a general parhood relation further specialised into a variety of parthood relations, as often suggested [16]. Rather, the atomic mereology built on $\leq$ is only aimed to model pluralities as finite sums of objects without assuming any unity or temporal constraint—in a spirit close to Lesniewski's efforts to eschew sets. The constitution relations, on the other hand, need to be temporalised to account for collective and composite changes across time. As required by Link, it is necessary that $\leq$ be atomic. Any entity considered as being singular, i.e., as having a unity, in particular any composite or collective, will be a $\leq$-atom, allowing it to be part of pluralities. Since $\leq$ is not a general parthood relation, being a $\leq$-atom only entails not being a plurality, and doesn't entail having no "parts" in a general sense of part.

4. The formal account

We consider two kinds of entities, namely *objects* (OB) like forests, cars, persons, etc. and *times* (TM), the class of instants. The framework can be extended to the case of events and temporal intervals, but for simplicity we focus here on objects and instants. To represent the *presence* (existence) of objects in time we introduce the binary predicate ε, with the formula $\varepsilon_t x$ standing for "the object x is present at time t", (a1).[1] We assume that any object is present at some time (a2).

a1 $\varepsilon_t x \rightarrow \mathrm{OB}x \wedge \mathrm{TM}t$

a2 $\mathrm{OB}x \rightarrow \exists t(\varepsilon_t x)$

To describe pluralities we consider an atemporal mereological relation $\leq$ defined on objects (a3). Following standard practice in mereology, we require that the whole exists whenever at least one of its parts exists (a4). We assume in (a5) that $\leq$ satisfies the axioms of *classical atomic*[2] *extensional mereology* [31] over objects, closed under unrestricted binary sum (+) and, for objects that partially overlap, under binary difference (−), see our implementation referred to in footnote 10 for the details.

a3 $x \leq y \rightarrow \mathrm{OB}x \wedge \mathrm{OB}y$

a4 $x \leq y \wedge \varepsilon_t x \rightarrow \varepsilon_t y$

a5 *the axioms of closed atomic extensional mereology hold on* $\leq$, $+$ *and* $-$ *over* OB

Definitions (d1) and (d2) standardly state that an *atom* is an object without proper parts, and that an *atomic part* is a part which is also an atom. We take *pluralities* to be non-atomic objects, that is, sums of two or more atomic objects.[3] Pluralities may be present also when just some of their atoms are. The notion of *wholly present* is introduced in (d3) with the formula $\varepsilon^{\mathrm{w}}_t x$ standing for "the object x is wholly present at time t", i.e., all the parts of x are present at t. Note that some pluralities are never wholly present like the sum of temporally disjoint atoms, as with 'Bach and Mozart'.

d1 $\alpha x \triangleq \neg\exists y(y \leq x \wedge y \neq x)$ (*atom*)

[1] To improve the reading of formulas, times are noted as subscripts.

[2] We adopt the atomicity axiom $\forall x(\mathrm{OB}x \rightarrow \exists y(y \leq_\alpha x))$, using (d2). With strong supplementation $(\mathrm{OB}x \wedge \mathrm{OB}y \wedge \neg(x \leq y) \rightarrow \exists z(z \leq x \wedge \neg\exists v(v \leq z \wedge v \leq y)))$ two entities having the same atomic parts are identical.

[3] As explained above, atoms correspond to singular objects as opposed to pluralities; this doesn't mean that an atom cannot have components or members, notions that are not captured by $\leq$.

d2 $x \leq_\alpha y \triangleq x \leq y \wedge \alpha x$ (*atomic part*)
d3 $\varepsilon^{\mathtt{w}}_t x \triangleq \forall y (y \leq x \rightarrow \varepsilon_t y)$ (*x is wholly present, wholly exists, at t*)

In the following we pursue the idea of grounding the distinction between collectives and composites using two *temporally qualified constitution* primitives (a6):

– $x \lhd_t y$ is meant to hold between a collective and a (wholly present) plurality;
– $x \prec_t y$ is meant to hold between a composite and a (wholly present) plurality.

The plurality constituting the object may change from time to time, however, if an object is $\lhd$- or $\prec$-constituted at some time, then it is so for all its life, see (a7) and (a8).

a6 $(x \prec_t y \vee x \lhd_t y) \rightarrow \varepsilon_t x \wedge \varepsilon^{\mathtt{w}}_t y \wedge \alpha x \wedge \neg \alpha y$
a7 $\varepsilon_t x \wedge \varepsilon_{t'} x \rightarrow (\exists y (x \lhd_t y) \leftrightarrow \exists y' (x \lhd_{t'} y'))$
a8 $\varepsilon_t x \wedge \varepsilon_{t'} x \rightarrow (\exists y (x \prec_t y) \leftrightarrow \exists y' (x \prec_{t'} y'))$

We do not enforce unicity on these relations: the same plurality can constitute different collectives as well as different composites, and it can constitute both collectives and composites. For instance, consider a deck of cards, a collective, which has been arranged into a tower, a composite. Each card is at the same time a member of the deck and a component of the tower, so the plurality of cards $\lhd$- and $\prec$-constitutes the deck and the tower, respectively. Furthermore, composites can be members of collectives and collectives can be components of composites.

Example 1 (Modeling a forest as collective and a larch as composite). To model a forest w as a collective of larches (say, just l_1 and l_2 for simplicity), we can write the formula: $w \lhd_t (l_1 + l_2)$. To model each larch as a composite of crown and trunk (say, cr_i and tr_i), we can write: $l_1 \prec_t (cr_1 + tr_1) \wedge l_2 \prec_t (cr_2 + tr_2)$, see Fig. 1 (solid edges represent $\leq$ and all bottom objects are $\leq$-atomic; we omit many sums for clarity).

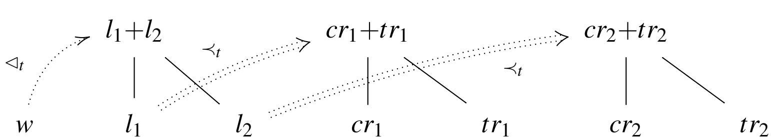


Figure 1. Modeling a forest as a collective and a larch tree as a composite.

Example 2 (Modeling a larch as composite with the crown component having foliage as a collective component). To model a larch l as a composite of trunk tr, and crown cr we write $l \prec_t (tr + cr)$, to model the crown cr as a composite with two branches b_1, b_2 and foliage f we write $cr \prec_t (b_1 + b_2 + f)$, and for foliage f being a collective of leaves, say, lv_1 and lv_2, we write $f \lhd_t (lv_1 + lv_2)$, see Fig. 2.

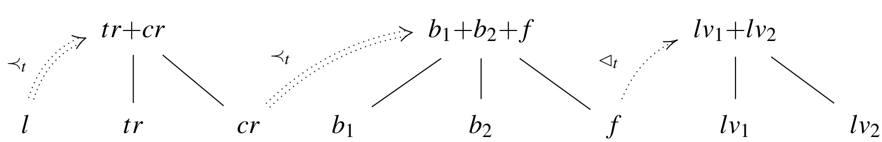


Figure 2. Modeling a larch as a composite with a component involving a collective.

The notions of *being collective* and *being member of* are defined in terms of $\lhd$, see (d4) and (d6), while the notions of *being composite* and *being component of* are defined in terms of $\prec$, see (d5) and (d7). From the definitions, members and components must be atomic objects, in line with Link's proposal where all singular objects are atoms. Note that collectives and composites are not assumed to be disjoint, i.e., it is possible for an object to be both $\lhd$- and $\prec$-constituted, a hypothesis explored at the end of this section.

d4 $\mathrm{CL}x \triangleq \forall t(\varepsilon_t x \to \exists y(x \lhd_t y))$ (*being a collective*)

d5 $\mathrm{CM}x \triangleq \forall t(\varepsilon_t x \to \exists y(x \prec_t y))$ (*being a composite*)

d6 $x\, \mathtt{memb}_t\, y \triangleq \exists z(y \lhd_t z \land x \leq_\alpha z)$ (*being member of*)

d7 $x\, \mathtt{comp}_t\, y \triangleq \exists z(y \prec_t z \land x \leq_\alpha z)$ (*being component of*)

The adoption of the above definitions makes it crucially important to distinguish $\lhd$ from $\prec$. Up to this point, they have the same characteristics, cf. (a6)-(a8). To formally differentiate the two constitution primitives, a first possibility is to identify a difference in the way pluralities are structured to constitute collectives vs. composites. As seen in Sect. 1, the literature, e.g. [4,5], considers that collectives have a uniform structure, while assemblies or functional complexes have a heterogeneous one. Some authors [33,34] further elaborate this idea by claiming that in a collective all the members play the same role, while components usually play a variety of different roles in a functional complex.

The notion of functional complex seems more restrictive than our notion of composite. For example, it seems not quite natural to consider the functional aspects of composites like mountains or molecules as fundamental, and even awkward to ask what roles could be involved in their structures. Second, this move requires us to formally characterise the notions of structure and role. Following Fine [35], a structure could be represented by means of a relation holding among the members or components. Collectives and composites could then be reduced to *variable embodiments*, i.e., entities that at any time t at which they are present are constituted by a *rigid embodiment*, i.e., a sort of compound of the objects, say, $a_1, \ldots, a_n$, and the relation R connecting them at t, shortly written $[a_1, \ldots, a_n/R]$. For instance, suppose that a car changes its engine from t_1 to t_2 and, for simplicity, that cars have just two components, namely an engine and a frame. According to Fine, the rigid embodiment $[e_1, f/R]$ that constitutes the car at t_1 is different from the rigid embodiment $[e_2, f/R]$ that constitutes the car at t_2, where e_1 and e_2 are two different engines and f is the frame. Note that both the rigid embodiments $[e_1, f/R]$ and $[e_2, f/R]$ and the pluralities $e_1{+}f$ and $e_2{+}f$ are compositionally static, but $e_1{+}f$ is wholly present whenever all its atoms are present, while $[e_1, f/R]$ requires the holding (at t_1) of $R(e_1, f)$ in addition to the existence of e_1 and f.

The introduction of variable embodiments has some drawbacks. First, it requires an additional kind of entity, namely, the rigid embodiments. The nature of rigid embodiments seems quite close to that of states in [36], but Fine, more recently, prefers to liken their ontological status to that of qua-entities [37]. Second, one could assume that variable embodiments are always constituted by rigid embodiments grounded on the same relation R. Even though this assumption seems in line with Rector and colleagues' approach, where composites are always composed by a determinate number of parts [33], it has been considered too restrictive by Jansen and Schulz [3], it does not apply to collectives that can lose or acquire members, and it is not endorsed by Fine himself. A variable embodiment can then be constituted, at different times, by rigid embodiments that are grounded on different relations (possibly with different arities). Therefore, by assuming that a given type of collectives or composites is associated with one structure, this structure cannot in general coincide with a single relation R, it should consist of the variety of relations, each one grounding a rigid embodiment, constituting the overall variable embodiment at some point in time.[4] One then should look for (meta-)criteria to characterise which kind of relations can be associated to a variable embodiment of a given type. Fine does not address this point (see [39] for additional criticisms).

As shown by Uzquiano [40], the introduction of pluralities avoids the commitment to rigid embodiments and it may alleviate the previous problem because the same property can apply to pluralities collecting different numbers of objects. In our framework—in line with the *structural-constitution view* introduced by Harris for group agents [25]—given finite sets $\mathscr{T}_{\mathrm{CL}}$ and $\mathscr{T}_{\mathrm{CM}}$ of unary predicates that represent types of, respectively, collectives and composites, one can introduce sufficient (and necessary) conditions for the existence of collectives or composites of a given type. More precisely, for each $\mathtt{P} \in \mathscr{T}_{\mathrm{CL}}$ an axiom with form (f1) can be introduced to ensure that for each wholly present plurality x satisfying $\mathtt{F}$ there exists a collective of type $\mathtt{P}$ constituted by x that during its whole life is constituted by pluralities satisfying the property $\mathtt{F}$, which is assumed to be flexible enough to allow changes in the number and in the configuration of the members. For instance, for forests, $\mathtt{F}$ should ensure that all the trees are spatially interconnected in a possibly quite general way (e.g., trees' neighbour distance is below a threshold and the plurality is maximal with respect to the distance criterion). Analogously, for each $\mathtt{Q} \in \mathscr{T}_{\mathrm{CM}}$ an axiom of the form (f2) can be added.[5] The approach can be strengthened by adding necessary conditions following (f3) and (f4).

f1 $\mathtt{F}_t x \wedge \neg \alpha x \wedge \varepsilon^{\mathtt{w}}_t x \rightarrow \exists y(\mathtt{P}y \wedge y \lhd_t x \wedge \forall t'(\varepsilon_{t'} y \rightarrow \exists x'(y \lhd_{t'} x' \wedge \mathtt{F}_{t'} x')))$

f2 $\mathtt{G}_t x \wedge \neg \alpha x \wedge \varepsilon^{\mathtt{w}}_t x \rightarrow \exists y(\mathtt{Q}y \wedge y \prec_t x \wedge \forall t'(\varepsilon_{t'} y \rightarrow \exists x'(y \prec_{t'} x' \wedge \mathtt{G}_{t'} x')))$

f3 $\mathtt{P}y \rightarrow \forall t(\varepsilon_t y \rightarrow \exists x(y \lhd_t x \wedge \mathtt{F}_t x))$

f4 $\mathtt{Q}y \rightarrow \forall t(\varepsilon_t y \rightarrow \exists x(y \prec_t x \wedge \mathtt{G}_t x))$

Axioms with forms (f1)-(f4) still do not distinguish between $\lhd$ and $\prec$: all the types in $\mathscr{T}_{\mathrm{CL}}$ and $\mathscr{T}_{\mathrm{CM}}$ have associated properties representing sufficient (and necessary) conditions. Are there differences between the nature of the properties $\mathtt{F}$ associated with the types of collectives and the nature of the properties $\mathtt{G}$ associated with the types of composites? Wilson [41] proposes to separate *compositional* constitution from *ampliative* constitution on the basis of the structure they rely upon. A structure is *intrinsic* when it holds only in virtue of how the constituents are interlinked, e.g., the bonds between the H and O atoms in a H_2O molecule. A structure is *extrinsic* when it holds because of some relations in which also non-constituent entities intervene, e.g., artefacts are usually defined also referring to intended capabilities and uses. Wilson's distinction seems however orthogonal to the one between $\lhd$ and $\prec$. Collectives like forests and composites like molecules seem both grounded on intrinsic structures while collectives like organisations and composites like artefacts on extrinsic structures. It seems then difficult to differentiate the nature of the "structures" of collectives vs. composites. Jansen and Schulz [3] re-

[4]Unless one considers polyadic or multigrade relations, see [38].

[5]Note that, if two distinct collectives of type $\mathtt{P}$ constituted by the same plurality exist (something consistent with our framework), an axiom with form (f1) guarantees the existence of just one collective. This could be the case, for instance, with two distinct societies of the same legal type $\mathtt{P}$ having exactly the same members and so the same constituting plurality instantiating $\mathtt{F}$ once. The rigid embodiment approach [35] could make the difference in case the members fill different arguments (roles) in some fine-grained enough relation R.

port that the BioTop Ontology[6] assumes that the components of composites are spatially self-connected, while the members of collectives are spatially scattered. However, they recognise that this constraint appears too strict even within the field of bio-medicine.

A promising alternative is to focus on the intuitive uniformity among the members of a collective; indeed, as pointed in Sect. 1, giving a collective one tends to specify the type of their members. This is not the case of composites that can include quite heterogeneous components. Following this intuition, Rector and colleagues [33] as well as Galton [42] assume that all the members of a collective are of a specified type. In our framework, via an axiom with form (f5), one can associate to each $\mathtt{P} \in \mathscr{T}_{\text{CL}}$ a type $\mathtt{T}$ applying to members. Galton does not require the type to be unique, but requires that the types characterising the members of a collective are closed under subsumption. Axioms of the form (f5) can then be introduced only for the minimal type (w.r.t. subsumption).[7]

f5 $\mathtt{P}y \wedge x\,\mathtt{memb}_t\,y \rightarrow \mathtt{T}x$

Unfortunately, this strategy presents problems similar to those met discussing structures of collectives and composites. Indeed, nothing prevents axioms with form (f5) from being applied to composites. For instance, all the components of a car are of type *being an artefact*. One can object that there is an important difference between types like *being a tree* and *being an artefact*: the first seems to capture quite closely the very nature of the objects one aims to classify. The latter, at least informally, seems to be characterised by much weaker conditions. In an ontological hierarchy, one expects to find *being a tree* at a lower level than *being an artefact*. However, it remains unclear whether such a difference on types could be drawn only on the basis of the level of ontological generality or whether this difference should be characterised in an alternative way.

Therefore, axioms with form (f1)-(f5) are still quite weak to clearly separate collectives from composites. We now introduce two further intuitions to differentiate $\lhd$ and $\prec$. First, we observe that collectives decompose into members in a unique way. The members of a forest are trees, the members of a crowd are persons. In contrast, in composite objects one often has a choice of possible decompositions. For instance, (the body of) a person may be decomposed into organs or, alternatively, into body parts like arms, legs, trunk, head and the like. In other words, to determine the components of a composite—but not the members of a collective—one needs to make some decomposition criterion explicit. This idea is also in line with the approach followed in [3], where components are always relative to a certain "partition"—a "(mechanical or cognitive) act or process of dividing something into parts" [3, p.5]—of a composite.[8]

We enforce the unique decomposition of collectives by means of (a9). Composites are not constrained by an anologous axiom, i.e., it is possible to have $x \prec_t y \wedge x \prec_t z \wedge y \neq z$. This also means that composites behave differently from variable embodiments: at every time at which a variable embodiment exists it has a unique rigid embodiment, thus synchronic "decompositions" into different components are banned in Fine's theory.

a9 $x \lhd_t y \wedge x \lhd_t z \rightarrow y = z$

[6] http://purl.org/biotop/

[7] Galton assumes that if members of a collective are both of type $\mathtt{T}$ and $\mathtt{T}'$, they also are of type $\mathtt{T}''$, where $\mathtt{T}''$ is subsumed by $\mathtt{T}$ and $\mathtt{T}'$, cf. axiom COLOF4 [42, p.16], to guarantee the existence of a unique minimal type.

[8] In [3] the components of a partition form a collective, while we here assume that they form a plurality.

The second intuition concerns recursive decomposition. Roughly speaking, it is possible to recursively decompose composites but not collectives. This is what the transitivity of relationships is about. As seen in Sect. 1, the members of the members of a collective are not members of the collective itself: memb is not transitive. For instance, the members of the countries that are members of UN, e.g., the federal states of USA are not members of UN. On the other hand, the components of a component of a composite are themselves components of the composite; consider the case of the branches of the crown of a tree in Example 2 above, the screws of the engine of a car, or the handle of the door of a house (although many would claim the handle is not a functional part of the house [19]). Similarly, the members of a collective which is a component of a composite are also components, see the case of the leaves of the foliage of a tree in Example 2. Rector and colleagues [33] also contrast a non-transitive *granular* parthood holding on collectives and a transitive *determinate* parthood holding on composites.

We formalise the recursive decomposition of composites in (a10). Importantly, this does not have a correspondent for relation $\lhd$. We need to ensure that alternative decompositions wholly "cover" the same composite. However, in the case of decompositions recursively ending up in atoms of different types, as in the previous example of body parts vs. organs, it is necessary to rely on some relationship between them, for instance using a common (material or spatial) substrate. Leaving such an extension for future work, (a11) simply excludes that one of the decompositions is a proper part of the other.

a10 $x \prec_t y \wedge a \leq_\alpha y \wedge (a \prec_t z \vee a \lhd_t z) \rightarrow x \prec_t ((y - a) + z)$

a11 $x \prec_t y \wedge x \prec_t z \rightarrow \neg(y \leq z \wedge y \neq z)$

Since $x \prec_t y \wedge y \prec_t z \rightarrow x \prec_t z$ is trivially true (likewise with $\lhd$) because the antecedent never holds (y cannot be both a plurality and an atom), it is more relevant to investigate the transitivity and other properties of comp and memb. The irreflexivity of memb and comp cannot yet be proven because nothing prevents an object from being an atomic part of the plurality that constitutes it, i.e., one of its own constituents. These unintended models are directly ruled out by (a12)—where the overlap relation $\between$ is defined in (d8)—see (t1).

d8 $x \between y \triangleq \exists z (z \leq x \wedge z \leq y)$ (*overlap*)

a12 $(x \lhd_t y \vee x \prec_t y) \rightarrow \neg(x \between y)$

t1 $\neg(x \, \texttt{memb}_t \, x) \wedge \neg(x \, \texttt{comp}_t \, x)$ (directly from (d6) and (a12))

The transitivity of comp (t2) and the "mixed" transitivity of memb and comp (t3) follow from (d7) and (a10), therefore comp is also asymmetric (t4). Vice versa, as desired, memb is not transitive because there are no axioms that, given $\exists a (y \lhd_t a \wedge x \leq_\alpha a)$ and $\exists b (z \lhd_t b \wedge y \leq_\alpha b)$, ensure a link between a and b (by (a9), b is the unique plurality that constitutes z). Even though the putative constraint (p1)[9] would guarantee the anti-transitivity of memb (p2), some examples bring evidence against it. For instance, consider an organisation O that among its members accepts both persons and organisations. In this case, John could be a member of the University of Oxford, and both John and the University of Oxford be members of O against (p2). Since the asymmetry of memb cannot yet be proven but is desirable, we impose it via (a13). Finally, a form of weak supplementation holds for both memb and comp, see (t5) and (t6).

[9] We label with (px) constraints that we want to discuss but are not included in the theory.

t2 $x \,\texttt{comp}_t\, y \wedge y \,\texttt{comp}_t\, z \rightarrow x \,\texttt{comp}_t\, z$
Proof. From the hypothesis and (d7), $\exists a(y \prec_t a \wedge x \leq_\alpha a)$ and $\exists b(z \prec_t b \wedge y \leq_\alpha b)$, i.e., $\exists ab(z \prec_t b \wedge y \leq_\alpha b \wedge y \prec_t a)$. By (a10), the fact that $x \leq_\alpha a$, and the transitivity of $\leq$ it follows that $z \prec_t ((b-y)+a) \wedge x \leq_\alpha ((b-y)+a)$, i.e., $x \,\texttt{comp}_t\, z$. □

t3 $x \,\texttt{memb}_t\, y \wedge y \,\texttt{comp}_t\, z \rightarrow x \,\texttt{comp}_t\, z$ (see the proof of (t2))

t4 $x \,\texttt{comp}_t\, y \rightarrow \neg(y \,\texttt{comp}_t\, x)$ (directly from (t1) and (t2))

p1 $x \lhd_t y \wedge a \leq_\alpha y \wedge a \lhd_t z \rightarrow \neg(y \between z)$

p2 $x \,\texttt{memb}_t\, y \wedge y \,\texttt{memb}_t\, z \rightarrow \neg(x \,\texttt{memb}_t\, z)$

a13 $x \,\texttt{memb}_t\, y \rightarrow \neg(y \,\texttt{memb}_t\, x)$

t5 $x \,\texttt{memb}_t\, y \rightarrow \exists z(\neg(z \between x) \wedge z \,\texttt{memb}_t\, y)$ (directly from (d1), (d2), (d6), (a5), (a6))

t6 $x \,\texttt{comp}_t\, y \rightarrow \exists z(\neg(z \between x) \wedge z \,\texttt{comp}_t\, y)$ (directly from (d1), (d2), (d7), (a5), (a6))

We can introduce *subcollective* and *subcomposite* relations via (d9) and (d10). Both `subcl` and `subcm` are trivially reflexive (at a given time). However, given the fact that the same plurality can constitute different objects `subcm` and `subcl` are not antisymmetric, and while `subcl` is transitive (t7), the fact that a composite can be decomposed in different ways rules out the transitivity of `subcm`.

d9 $x \,\texttt{subcl}_t\, y \triangleq \exists ab(x \lhd_t a \wedge y \lhd_t b \wedge a \leq b)$

d10 $x \,\texttt{subcm}_t\, y \triangleq \exists ab(x \prec_t a \wedge y \prec_t b \wedge a \leq b)$

t7 $x \,\texttt{subcl}_t\, y \wedge y \,\texttt{subcl}_t\, z \rightarrow x \,\texttt{subcl}_t\, z$
Proof. From the hypothesis and (d9) we have $\exists ab(x \lhd_t a \wedge y \lhd_t b \wedge a \leq b)$ and $\exists cd(y \lhd_t c \wedge z \lhd_t d \wedge c \leq b)$. By (a9), we have $b = c$, and by the transitivity of $\leq$, $a \leq d$. Thus $\exists ad(x \lhd_t a \wedge z \lhd_t d \wedge a \leq d)$. □

Note that $x \prec_t (y+z)$, $y \prec_t (a+b)$, and $z \prec_t (c+d)$ imply $x \prec_t (a+b+c+d)$ by (a10). The other direction does not hold: $x \prec_t (a+b+c+d)$ does not imply the existence of objects y and z that are both subcomposites and components of x. One could try to resume Fine's approach [43] to model the difference between these two situations. In [43], Fine introduced a *summation operator* Σ such that $\Sigma(\Sigma(x_1,x_2,\dots),\Sigma(y_1,y_2,\dots))$ is different from $\Sigma(x_1,x_2,\dots,y_1,y_2,\dots)$. However, Σ is not temporally qualified, therefore even though $x \prec_t \Sigma(\Sigma(a,b),\Sigma(c,d))$, $x \prec_t \Sigma(a,b,c,d)$ and the two sums are different, this cannot be used to provide the needed y and z (with $y \prec_t \Sigma(a,b)$ and $z \prec_t \Sigma(c,d)$) since in our framework y and z can change their components through time. Our framework is thus richer and requires more expressive operators.

As said, collectives and composites are not disjoint. This possibility is particularly interesting for social groups and organisations that can be described not only in terms of their members but also in terms of the variety of decompositions into departments or committees with specific roles within the organisation or group. In fact, approaches in which collectives and composites are disjoint and which insist on the homogeneity of collectives to conclude that organisations with differentiated roles such as a string trio (three musicians playing a violin, a viola and a cello) can be considered as a composite are unable to account for the fact that they have members, i.e., that members can be unambiguously counted (quite obvious for a trio), that some homogeneity is present since members are of a same type (musicians for a trio), and that no component of a member (e.g., Lea's foot) is relevant for the organisation. Sect. 5 discusses additional examples.

Additional constrains are needed for composites that are also collectives. First, we ensure that alternative decompositions "cover" the whole composite-collective, completing (a11) with (a14), which guarantees that all the members also are components. Second, we make sure with (a15) that the composite and the collective views match up to the member's level, i.e., that all the components "larger" than a member are also subcollectives. By (t2), components of members, e.g., Lea's foot, still are components of composite-collectives like the string trio. So, to avoid considering such components, we finally introduce compcl (*component of collective*) as a (non-transitive) sub-relation of comp dedicated to compositive-collectives that stops at member's level, see (d11).

a14 $\mathtt{CL}x \wedge \mathtt{CM}x \wedge y\,\mathtt{memb}_t\,x \rightarrow y\,\mathtt{comp}_t\,x$
a15 $\mathtt{CL}x \wedge \mathtt{CM}x \wedge y\,\mathtt{comp}_t\,x \wedge \exists z((z\,\mathtt{comp}_t\,y \vee z\,\mathtt{memb}_t\,y) \wedge z\,\mathtt{memb}_t\,x) \rightarrow y\,\mathtt{subcl}_t\,x$
d11 $x\,\mathtt{compcl}_t\,y \triangleq \mathtt{CL}y \wedge \mathtt{CM}y \wedge x\,\mathtt{comp}_t\,y \wedge (x\,\mathtt{memb}_t\,y \vee x\,\mathtt{subcl}_t\,y)$

The proposed theory has been implemented and tested for consistency by means of theorem provers.[10] More precisely, (a1)-(a15) is a consistent theory, and its extension with sample axioms of forms (f1)-(f5) has been proved consistent as well.

5. An application: social groups

We now analyse how social groups can be represented as collectives and/or composites in our account. Adopting this framework, we assume an anti-reductionist position on groups: both collectives and composites are constituted by and distinct from pluralities. So our approach stands on the side of the constitution-based ones [22,23,24,25,26,27]. This literature usually assumes the existence of properties keeping group members together. In most cases, such property is taken to be the structure, which some authors specify as functional structure. We take this to be similar to what we proposed with axioms (f1)-(f4) for collectives and composites. Moreover, some authors require all members to belong to a same type, e.g, *being a person*, *a social entity* or *an agent*. This is in line with (f5). Finally, as in all these theories based on constitution, we can account for groups surviving the change of their members and for different groups sharing their members.

Now, in this framework, should social groups be considered as collectives or as composites? Our position is that all social groups are collectives, but that only some of them are also composites. This choice is motivated by the fact that collectives but not composites have an associated member-type (f5), that (a9) leaves no ambiguity on what is a member of a collective, but that some groups, like composites, can have heterogeneous components and can be decomposed in alternative ways. We will thus talk of groups that are *pure collectives* and of those that are *composite-collectives*. Accounting for the double nature of the latter groups is made possible in our framework where collectives and composites are not disjoint, and where compcl is the relation of choice for them.

Fine [27] highlights that the structure of social groups can be more or less layered and articulated. In our framework the variety of the layered structures of social groups can be captured by chaining $\lhd$- and $\prec$-constitutions. For example, we can have collectives that have as members other collectives, e.g., the European Union, which has member States (that, arguably, are in their turn collectives), collectives with both collectives and individuals (i.e., non-collective objects) as members, e.g., scientific associations which can have both individuals and institutions as members (in this case, the members or the components of the institutions are not necessarily members of the association), compos-

[10] See https://github.com/diporello/plurals/blob/master/ontology_of_plurals.p

ites that have as components only collectives, only composites, collectives and individuals (composites or not), up to the most complex cases, like big companies, which can have as components individuals, e.g. the President, collective-composites, e.g. an executive board composed of subgroups like an education and a communication committee, and pure collectives, i.e., all the other members. It is important to stress here that while collectives directly provide the members, composites allow us to finely specify their structural decomposition. Suppose, for example, to have the following collectives: c is the Milan football team (with 11 football players), c_1 the defense, c_2 the attack, c_3 is a musical trio (composed by 3 footballers), and c_4 a bridge club (composed by the remaining 8 footballers). Because the members of c_{1-4} are among the ones of c, by (d9), c_{1-4} are all subcollectives (but not members) of c. Our composite-collectives allow, via (a14) and (a15), to include among the components only some subcollectives, e.g., the Milan football team c can be represented as a composite-collective with (in addition to the 11 footballers) c_1 and c_2, but not c_3 and c_4, as components. Composite social groups can then range from the Italian Parliament—which has as members the parliamentarians, but can be decomposed in different ways (e.g., in Chamber and Senate, as well as in parliamentarian groups)—to the aforesaid Milan club that indeed, differently from pure collectives, can be decomposed into defense and attack but also in alternative or more fine-grained ways, e.g., identifying goal-keepers, defenders, midfielders and strikers.[11]

Though most examples seem to rely on a notion of structure that is tightly connected with functionality or with power-responsibility relations—a fact that deeply influenced the social ontology literature, our approach allows to keep these two dimensions separate. Structure can in our framework rely on very different rationales. This seems to be a desirable feature for social groups, as it allows to account for the various ways in which a complex organisation can be synchronically organised. For instance, we could decide to decompose a composite social group, like a multinational company whose offices occupy four floors of a building, on the basis of the floor in which offices are located as well as on the basis of the organisational charter defining roles and responsibilities, etc.

Another relevant distinction in the case of social groups is that between *mere groups* and *group agents*; in [44] we have argued that what distinguishes them is that the latter have an established decision procedure, which allows them to act as a whole, while the former do not. Though collective agency is often discussed in connection with structure, in our approach the two are orthogonal dimensions. In fact, we could have non-agentive as well as agentive pure collectives (or groups and group agents) and both agentive and non agentive composite-collectives (or composite groups and composite group agents). An example of non-agentive pure collective is the collective of Starbucks bartenders (neither one of them taken individually nor they together can act on behalf of Starbucks), while an agentive pure collective is the Supreme Court of the United States. Parliaments and sport teams are agentive composite-collectives, while an example of non agentive composite-collective is a choir of an orchestra, which can be decomposed in sopranos, baritones, tenors and so on, but whose decisions (for instance which songs to sing) are taken by someone who's not a component of the choir. Something worth pointing out is that, in all cases, including social groups that have both members and subcollectives as components, the fact that the group is agentive or not is not a direct consequence of

[11] By (a14) and (a15), the most fine-grained `compcl`-decomposition of a social group represented as a composite-collective is always given in terms of the members of the collective.

its components being agentive or not, but it rather depends on the group having its own decisional procedure, allowing it to act as a whole.

6. Conclusions

This paper presents the core of a formal theory for representing the notions of plurality, collective and composite. The proposed theory is the result of an analysis of various formal approaches in the literature and of their drawbacks. The focus has been on the intrinsic structure of collectives and composites brought to the light by the use of two different relations of constitution. In particular, we were interested in developing a framework applicable to social reality, where a theory of this kind is definitely needed. However, the scope of the theory is more general, spanning natural objects, artefacts, and agentive/non-agentive entities. Our aims for the future are on the one hand to enrich the theory with functionalities and roles and to further develop its application to social reality. On the other hand, we plan its integration into some foundational ontology, mapping $\leq$ (with appropriate temporal constraints on the relata), `comp` and `memb` into a general temporalised parthood on objects, and introducing in the picture the amounts of matter constituting material objects, as Link did.

Acknowledgements We are grateful to Adrien Barton, Kit Fine, and four anonymous reviewers for helpful comments.

References

[1] Bottazzi E, Catenacci C, Gangemi A, Lehmann J. From collective intentionality to intentional collectives: An ontological perspective. Cognitive Systems Research. 2006;7(2-3):192–208.
[2] Galton A, Wood Z. Extensional and intensional collectives and the de re/de dicto distinction. Applied Ontology. 2016;11(3):205–226.
[3] Jansen L, Schulz S; Springer. Grains, components and mixtures in biomedical ontologies. Journal of Biomedical Semantics. 2011;2(Suppl 4):S2.
[4] Winston M, Chaffin R, Herrmann D. A taxonomy of part-whole relations. Cognitive Science. 1987;11(4):417–444.
[5] Gerstl P, Pribbenow S. Midwinters, end games, and body parts: a classification of part-whole relations. International Journal of Human-Computer Studies. 1995;43(5-6):865–889.
[6] Link G. The logical analysis of plurals and mass terms: A lattice theoretical approach. In: Bäuerle R, Schwarze C, von Stechow A, editors. Meaning, Use and the Interpretation of Language. de Gruyter; 1983. p. 302–323.
[7] Link G. Algebraic Semantics in Language and Philosophy. Lecture Notes. CSLI; 1998.
[8] Boolos G. To Be Is To Be a Value of a Variable (or to Be Some Values of Some Variables). The Journal of Philosophy. 1984;81(8):430–449.
[9] Linnebo Ø. Plural Quantification. In: Zalta E, editor. The Stanford Encyclopedia of Philosophy. Summer 2017 ed. Stanford University; 2017.
[10] Vieu L, Borgo S, Masolo C. Artefacts and Roles: Modelling Strategies in a Multiplicative Ontology. In: Eschenbach C, Gruninger M, editors. Proceedings of Fifth International Conference on Formal Ontology and Information Systems (FOIS 2008). IOS Press; 2008. p. 121–134.
[11] Moltmann F. Parts and Wholes in Semantics. Oxford University Press; 1997.
[12] Winter Y. Atom predicates and set predicates: Towards a general theory of plural quantification. In: Semantics and Linguistic Theory. vol. 8. Cornell University; 1998. p. 249–266.
[13] Barker C. Group terms in English: Representing groups as atoms. Journal of Semantics. 1992;9(1):69–93.

[14] Linnebo Ø, Nicolas D. Superplurals in English. Analysis. 2008;68(3):609–681.
[15] Wasserman R. Material Constitution. In: Zalta E, editor. The Stanford Encyclopedia of Philosophy. Fall 2018 ed. Stanford University; 2018.
[16] Varzi A. A note on the Transitivity of Parthood. Applied Ontology. 2006;1(2):141–146.
[17] Vieu L, Aurnague M. Part-of Relations, Functionality and Dependence. In: Aurnague M, Hickmann M, Vieu L, editors. The Categorization of Spatial Entities in Language and Cognition. John Benjamins; 2007. p. 307–336.
[18] Keet M, Artale A. Representing and reasoning over a taxonomy of part–whole relations. Applied Ontology. 2008;3(1-2):91–110.
[19] Vieu L. On the transitivity of functional parthood. Applied Ontology. 2006;1(2):147–155.
[20] Hawley K. Social Mereology. Journal of the American Philosophical Association. 2017;3(4):395–411.
[21] Strohmaier D. Group Membership and Parthood. Journal of Social Ontology. 2019;4(2):121–135.
[22] Sheehy P. The Reality of Social Groups. Routledge; 2006.
[23] Epstein B. The Ant Trap: Rebuilding the Foundations of the Social Sciences. Oxford University Press; 2015.
[24] Thomasson A. The Ontology of Social Groups. Synthese. 2019;196(12):4829–4845.
[25] Harris K. How Individuals Constitute Group Agents. Canadian Journal of Philosophy. 2020;50(3):350–364.
[26] Ritchie K. Social Structures and the Ontology of Social Groups. Philosophy and Phenomenological Research. 2020;100(2):402–424.
[27] Fine K. The Identity of Social Groups. Metaphysics. To appear.
[28] Ruben DH. Social Wholes and Parts. Mind. 1983;XCII(366):219–238.
[29] Ruben DH. The Metaphysics of the Social World. Routledge & Kegan Paul; 1985.
[30] Mizoguchi R, Kitamura Y, Borgo S. A unifying definition for artifact and biological functions. Applied Ontology. 2016;11(2):129–154.
[31] Varzi A. Mereology. In: Zalta E, editor. The Stanford Encyclopedia of Philosophy. Spring 2019 ed. Stanford University; 2019.
[32] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. The WonderWeb Library of Foundational Ontologies and the DOLCE ontology – WonderWeb Deliverable D18. LOA-ISTC-CNR; 2003.
[33] Rector A, Rogers J, Bittner T. Granularity, scale and collectivity: when size does and does not matter. Journal of Biomedical Informatics. 2006;39(3):333–349.
[34] Guizzardi G. Ontological foundations for conceptual part-whole relations: The case of collectives and their parts. In: Mouratidis H, Rolland C, editors. Proceedings of the 23rd International Conference on Advanced Information Systems Engineering (CAiSE 2011). Springer; 2011. p. 138–153.
[35] Fine K. Things and Their Parts. Midwest Studies in Philosophy. 1999;XXIII(61-74).
[36] Masolo C, Benevides AB, Porello D. The interplay between models and observations. Applied Ontology. 2018;13(1):41–71.
[37] Fine K. Acts and Embodiment. In: Hinshelwood A, editor. Being and Doing. Oxford University Press; To appear.
[38] Oliver A, Smiley T. Multigrade predicates. Mind. 2004;113(452):609–681.
[39] Ferrario R, Masolo C, Porello D. Organisations and Variable Embodiments. In: Borgo S, Hitzler P, Kutz O, editors. Formal Ontology in Information Systems: Proceedings of the 10th International Conference (FOIS 2018). IOS Press; 2018. p. 127–140.
[40] Uzquiano G. Groups: Toward a Theory of Plural Embodiment. Journal of Philosophy. 2018;115(8):423–452.
[41] Wilson R. A puzzle about material constitution and how to solve it: Enriching constitution views in metaphysics. Philosophers' Imprint. 2007;7(5).
[42] Galton A. How is a collection related to its members. In: Proceedings of the Third Interdisciplinary Ontology Meetings. Keio University; 2010. p. 9–17.
[43] Fine K. Towards a theory of part. The Journal of Philosophy. 2010;107(11):559–589.
[44] Porello D, Bottazzi E, Ferrario R. The Ontology of Group Agency. In: Garbacz P, Kutz O, editors. Proceedings of 8th International Conference on Formal Ontology and Information Systems (FOIS 2014). IOS Press; 2014. p. 183–196.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200672

The Mereological Structure of Informational Entities

Adrien BARTON[a,b,1], Fumiaki TOYOSHIMA[b], Laure VIEU[a,c], Paul FABRY[b] and Jean-François ETHIER[b,2]
[a] *Institut de Recherche en Informatique de Toulouse, CNRS, France*
[b] *GRIIS, Université de Sherbrooke, Quebec, Canada*
[c] *Laboratory for Applied Ontology, ISTC-CNR, Trento, Italy*

Abstract. This article provides the basis of a formal axiomatic system for a mereology of informational entities based on the idea of information fillers that can occupy information slots, such as the same word that can be used in different sentences. It is inspired by Karen Bennett's mereological system that enables a whole to have a part "twice over", but differs from it in several key points, such as the acceptance of empty slots, and the possibility for slots to have slots. Information slots are analyzed as informational entities that can carry aboutness.

Keywords. Mereology, Information content entity, Informational structure

1. Introduction

Documents are a primary source of data. Consider the field of medicine: many data about which medications a patient takes (or is likely to take) are extracted from prescription documents written to him or her by doctors, or from drug dispensing reports written by pharmacists [1]. Forms and surveys are another important source of data.

As we will see below, documents receive growing attention in information systems and ontologies. However, the informational entities that compose such documents are often characterized simply as entities that are "about" something, and their analysis is too basic to enable an accurate and manageable representation of data and information in various fields. To address this problem, we will provide a mereological analysis of the structure of informational entities, as the meaning of a complex informational entity depends on the meanings of its parts (although maybe not only on those). Interestingly enough, this task will lead to a foundational challenge to reconsider classical mereological systems that are traditionally used in formal ontology. We will here only consider documents (and more generally informational entities) that are composed in natural language, excluding pictures, musical partitions, etc. – although this limitation could be lifted.

[1] Corresponding Author: Adrien Barton, IRIT, 118 Route de Narbonne, F-31062 Toulouse Cedex 9, France; E-mail: adrien.barton@irit.fr.
[2] Corresponding Author: Jean-François Ethier, Université de Sherbrooke, 2500 Boulevard de l'Université, Sherbrooke, QC, J1K 2R1, Canada; E-mail: ethierj@gmail.com.

2. Preliminaries and state-of-the-art works

2.1. Mereology among universals or particulars

Mereology is a formal study of the part-whole relation, which constitutes a mainstay of ontological practice. We may consider building a mereology at two levels: between particulars, or between universals. This article will focus on a mereology of particulars of informational entities. Former work by Masolo and Vieu [2] has focused on a mereology of universals (not necessarily those of informational entities). Particulars of informational entities and universals of non-informational entities share some commonalities [3, p. 105-107], such as the ability to be multiply localized. This is reflected in the fact that Bennett's [4] mereological work, used for building a mereology of universals in the work by Masolo and Vieu [2], will also serve as a basis for our formal ontology of informational entity particulars.

2.2. Classical extensional mereology

Different mereological systems have been proposed. The most standard is sometimes informally called "Classical Extensional Mereology" (CEM) (not necessarily in Simon's [5] sense). CEM embraces "ground mereology" [6] according to which parthood is a (partial) ordering relation (reflexive, antisymmetric, and transitive) and that accepts the two following principles (the former being entailed by the latter):

- Weak Supplementation Principle (WSP): If x is a proper part of y, then there is some z such that z is a part of y and z is disjoint from (i.e. does not overlap) x.
- Strong Supplementation Principle (SSP): If y is not part of x, then there is some z such that z is a part of y and z is disjoint from x.

Note that SSP implies that two different entities cannot have exactly the same proper parts. For instance, the upper ontology DOLCE [7] builds upon "general extensional mereology" [6] satisfying both WSP and SSP; whereas the upper ontology Basic Formal Ontology (BFO 2.0) [3] adopts a "minimal extensional mereology" [5] including WSP but not SSP. Although Simons [5] endorses WSP and recommends dropping out SSP, WSP is still controversial [8] (for an overview of the decidability of various mereological theories, see [9]).

Most importantly, CEM is committed to the principle that an entity x cannot have an entity y as a part many times over [4][10]. We will reject this principle below because it turns out to be unsuitable for developing a mereological account of informational entities.

2.3. Informational entities in conceptual modeling and ontology

There are some existing works on informational entities in the domain of conceptual modeling and ontology. For instance, the Functional Requirements for Authority Data (FRAD) [11] introduces an entity named "expression": "The intellectual or artistic realization of a work in the form of alphanumeric, musical, or choreographic notation, sound, image, object, movement, etc., or any combination of such forms." To take another example, the Unified Nations System Document Ontology (UNDO) [12] aims to provide a framework for the formal description of all entities and the relations that hold among them in the documents of the United Nations. Both the FRAD and the UNDO are largely practically motivated and they leave room for meticulous ontological analysis of documents and information. Finally, CIDOC [13] is a lightweight ontology aimed at cultural heritage domain. It includes a mereological relation for information objects ("incorporates"), but its axiomatization is very limited.

Two ontologies based on BFO deal with informational entities. First, the Information Artifact Ontology (IAO) [14] introduces the class *Information Content Entity* (ICE), the instances of which can be documents, databases, and digital images, as a subclass of *Generically Dependent Continuant* (whose instances, intuitively, "can migrate from one bearer to another through a process of copying" [6] [p. 179]). ICEs are about some "portion of reality", a label which encompasses all the BFO particular entities (including other ICEs) but also universals, relations, and "configurations" (e.g., the cat being on the mat). Second, the Information Entity Ontology in the Common Core Ontologies (CCO) [15] identifies three subkinds of the IAO:is_about relation, namely: "describes" (used for e.g., reports and representations), "prescribes" (used for e.g., plans and artifact specifications), and "designates" (used for e.g., names and other identifiers).

Garbacz [16] has proposed an ontology of texts, that include considerations on parthood, precedence, identity and location. Masolo et al. [17] introduced a notion of description based on DOLCE, characterized as follows: "different expressions (…) can be associated to the same description" (generic dependence) and "descriptions must be encoded on (…) physical supports" (concretization). This notion has been exploited within ontology patterns for describing information objects [18], but a full-fledged formal ontology of informational entities has not been pursued within DOLCE yet. Finally, the formal system we will present has some analogies with feature structures – see 5.7 for a short discussion.

3. A classification of informational entities

3.1. A brief consideration on informational entities, aboutness, and semiotics

Documents have a dual face: physical objects on one hand (e.g. a copy of the book *Labyrinths*), and informational content that can be concretized by such physical objects on the other hand (e.g. the informational content shared by all such copies of *Labyrinths*). We will here be interested in this second sense of documents.

Many documents are constituted by sentences, which can be decomposed in words. Words written in alphabetic systems can be decomposed in letters. All those entities will be called here "Informational entities" (IEs). The models of informational entities presented above seem to share the premise that informational entities are about (synonyms: "refers to", "represents", "mentions") something. This is indeed often the case with documents, (declarative) sentences therein, and words that constitute the sentences: e.g., the word "cat" can be generally taken to be about the class *Cat*. It may not be the case of all informational entities, however. For instance, it is debatable whether letters such as "A" are about anything (although one might argue that they are about sounds). Therefore, IEs should not be identified with ICEs from IAO, as they are not necessarily about something.

Aboutness remains a notoriously elusive notion, despite some recent philosophical works [19]. We will not delve into its nature (but see section 5.4), as some groundwork on mereology is necessary to adequately deal with this issue, but will presuppose that aboutness emerges from a semiotic system. Relatively neglected in ontology [20], semiotics analyzes representations in terms of the triad of a sign, an object and an interpreter. That is to say, meaning is "an attribution of significance by some sign users for other sign users for some designated purposes" [20] [p. 120]. Therefore, for example, the letter "A" might be about something, but only when somebody refers to "A" with the intention of conveying some meaning (e.g., an excellent grade) to another person. To take seriously semiotic considerations in ontology may require a foundational investigation into language [20], which is outside the scope of this paper. What we will seek below is a mereological theory of informational entities without assuming any reference to an aboutness relation (but see section 5.4).

3.2. The need for information slots

It seems that some informational entities can be found in several documents: for example, the IE 'flu' can be found in a part $dict_0$ in a medical dictionary that reads 'flu = an infectious disease caused by an influenza virus', and in a (here idealized) diagnosis $diag_0$ written by Dr. House about John Doe that reads: 'John Doe / flu / Dr. House'. This raises two challenges. First, the same IE could appear several times in the same document. For example, suppose that another line in the same diagnosis document reads 'John Doe / asthma / Dr. Jones': 'John Doe' would then appear twice in the document. Second, two IEs that would play different roles would be instances of different classes that would naturally be seen as disjoint (e.g. 'flu' as an instance of *Diagnosed disease specification* in a diagnosis document versus 'flu' as an instance of *Therapeutic indication* in a drug prescription that prescribes to take some medication in case of flu). To account for this, we will introduce, following Bennett [4], the classes of Information Slot (IS) (somewhat akin to CCO's Information Structured Entity, although to our knowledge, the nature of the latter has not been investigated in detail yet) and Information Filler (IF), both being subclasses of the class of IE: the same individual IF 'flu' can be found in both $diag_0$ and $dict_0$, as filling different individual ISs (note that such an ontology of slots and fillers fits also especially well with a machine-readable language such as XML).

Similarly, consider the chains of characters IE_1='ab' and IE_2='ba' (for theories of strings, see [21,22]): we will consider that IE_1 has two ISs '1^{st} $letter_1$[]' and '2^{nd} $letter_1$[]', and that IE_2 has two ISs '1^{st} $letter_2$[]' and '2^{nd} $letter_2$[]'. The same individual filler 'a' occupies '1^{st} $letter_1$[]' and '2^{nd} $letter_2$[]', and the same individual filler 'b' occupies '1^{st} $letter_2$[]' and '2^{nd} $letter_1$[]' (although we will not deal here with the representation of order relations between slots).

3.3. Information slots and information fillers

Suppose that in a hospital, all diagnostic documents have the following structure: 'patient[] condition[] doctor[]'. That is, any diagnostic report at this hospital has (at least) three ISs, each of which can be filled by an IF. When we write "t[x]", x refers to an IF and t to an IS that is filled by x.

We will also consider that IEs can have a structure even if this structure is not filled – that is, they can have a "mere mereological structure" (see section 5.6). This means that in our ontology, an IS does not need to be filled by an IF.

Note that two documents of the same type do not have the same ISs, although they can be filled with the same kinds of IFs (or even the same IFs). Suppose that Dr. House fills the document $diag_0$: '$patient_0$['John Doe'] $condition_0$['Flu'] $doctor_0$['Dr. House']' and $diag_1$: '$patient_1$['Jane Brown'] $condition_1$['Asthma'] $doctor_1$['Dr. House']'. Although $diag_0$ and $diag_1$ have similar structures, they have different particular ISs: '$patient_0$[]' is different from '$patient_1$[]', '$doctor_0$[]' is different from '$doctor_1$[]', etc. However, '$patient_0$[]' and '$patient_1$[]' are instances of the same class *IS for patient name*, '$doctor_0$[]' and '$doctor_1$[]' are instances of the same class *IS for doctor name*, etc. On top of that, the same particular IF 'Dr. House' fills both particular ISs '$doctor_0$[]' and '$doctor_1$[]' – and we could introduce the class *IF for doctor name*, of which 'Dr. House' is an instance. Therefore, there are both particulars and classes of ISs and IFs.

Finally, as we have seen earlier, not only words can occupy ISs: the same letter can appear in several words by occupying several different ISs. For example, in the word 'aa', the same particular IF 'a' occupies the IS '1st $letter_{aa}$[]' and the IS '2nd $letter_{aa}$[]'.

3.4. Information slot as a generically dependent continuant that can be concretized

To analyze ISs and IFs, we anchor them in the IAO ontological framework for ICEs in such a way that informational entities are generically dependent upon their bearers (see [23] for a detailed discussion on generic dependence) and exist by being "concretized" (alternatively, see [24] for an analysis of the general notion of slot in terms of grounding and essence). However, we extend this idea to all informational entities, including those that are not ICEs. The letter "A", for instance (even if it is not about anything, and thus not an ICE), may be concretized as an ink pattern on a paper, or as a pixel pattern on a computer screen.

Suppose there are two concretizations of $diag_0$: a first one printed on paper, that is concretized by the ink pattern p_1, and a second one on my computer screen, concretized by a pixel pattern p_2. Let's call IF_1 = 'John Doe', IF_2 = 'Flu' and IF_3 = 'Dr. House' the three IFs that constitute what we will call the "content" of $diag_0$. IF_1, IF_2 and IF_3 are concretized by (parts of) p_1, as well as (parts of) p_2. In our ontological framework, each IS in $diag_0$ is also concretized (at least) twice, since $diag_0$ is concretized twice (in p_1 and p_2). ISs are in this respect similar to IFs.

A difficulty with ISs lies in pinpointing their concretizations: it might be expected that diagnostic sheets at a hospital should be filled with the name of a doctor at the bottom part of the sheet, without anything indicating the need for such a name on the paper. Even if something indicates it, such as the words 'Doctor name' written on the paper, those words are not an IS, but rather an IF that indicates the (otherwise invisible, but socially determined) existence of an IS. Maybe, if the prescription is concretized on a sheet of paper, such an IS would be concretized by a BFO:*Site* [3, pp. 112-113]. For example, if $diag_0$ is printed on a paper, the slot '$doctor_0$[]' in $diag_0$ is concretized by a site on the printed document that is occupied by the ink pattern on the paper concretizing the IF 'Dr. House'. Another possibility would be that ISs are concretized by some cognitive structure in a collective of agents, reflecting on the social nature of ISs. We will not elaborate further on this question in this paper, and instead focus on the axiomatic mereology of informational entities.

4. An axiomatization of mereological relations among informational entities

We will first present an axiomatical mereological system for IEs (4.1-4.3) inspired by Bennett's system [4], provide a simple model in 4.4, and then show in 4.5 how this system differs from Bennett's.

4.1. Ground axiomatization

4.1.1. Key predicates

Let us adapt Bennett's mereology, where Sty means that t is an IS (what Bennett called a "slot", while she used the notation "P_S" instead of "S") of y, Fxt means that x fills t, and Pxy means that x is a part of y. We introduce as primitive the unary predicate IS = "is a slot", and the binary predicates F and S, where S is defined on the domain IS:

(AX0) **Only slots are slots of something** $Stx \rightarrow ISt$

We then define the following predicates IF = "is a filler" and HS = "having a slot":

(DEF0) **Have-slot and filler** $HSa :=_{def} \exists t\ Sta$

$IFx :=_{def} \exists t\ Fxt$

(the variable symbols "t", "u", "v" will be used for slots, "x", "y", "z" for fillers, and "a" for entities that can be either slots or fillers)

We will build a mereological theory on the domain of fillers and slots. This means that each non-slot entity under consideration is a filler, and therefore fills some slot: $diag_0$, for example, would fill a slot '$diagnosis_0$[]' that may not be a slot of any filler. This conception fits with the idea that there are no "free-floating" IFs, but that they always appear in a context (represented at least partially by the slot they fill) with some social expectations defined by the semiotic system on which they depend.

x is a *proper* part of a filler y if x fills a slot of y:

(DEF1) **Proper filler-parthood** $PPxy :=_{def} IFy \ \& \ \exists t\ (Sty \ \& \ Fxt)$

Note that IFy is imposed to make sure that PP holds only between fillers (trivially, $PPxy \rightarrow IFx \ \& \ IFy$), so as to avoid that a filler would be a proper part of a slot.

We then define parthood on the basis of proper parthood in a similar way as in [5], among fillers:

(DEF2) **Filler-parthood** $Pxy :=_{def} PPxy \lor (IFy \ \& \ x=y)$

4.1.2. First axioms

In this system, only slots are filled, and slots cannot fill:

(AX1) **Only slots are filled** $Fxt \rightarrow ISt$

(AX2) **Slots cannot fill** $Fxt \rightarrow \neg ISx$

Moreover, no entity fills any of its slots (intuitively, the slots of an entity can only be filled by something "smaller" than this entity):

(AX3) **No improper parthood slots** $\neg(Stx \ \& \ Fxt)$

Also, there is at most one filler for a given slot:

(AX4) **Max one occupancy** $Fyt \ \& \ Fzt \rightarrow y=z$

From AX2 and DEF0, we deduce the trivial theorem that fillers and slots are disjoint:

(TH0) **Fillers are not slots** $IFx \rightarrow \neg ISx$

From AX1 and DEF0, we deduce that the domain of F is IF, and its range is IS; whereas the domain of S is IS, and its range is HS (by definition of HS):

(TH1) **Domain and range of F** $Fxt \rightarrow IFx \ \& \ ISt$

(TH2) **Domain and range of S** $Stx \rightarrow ISt \ \& \ HSx$

Like in Bennett's theory, this system does not require that each filler occupies exactly one parthood slot. This is indeed the main motivation of the system, to explain how IEs such as the letter ‘a’ or ‘John Doe’ can occupy several parthood slots of the same entity. That is, an entity can “have a part twice over”, in Bennett’s slogan.

However, contrarily to Bennett’s system (see section 4.5 for a full comparison), a slot may not be filled; and slots can have slots. Indeed, we want to be able to say that on a document, the slot ‘cn_0[]’ (for “complete name”) has as slots ‘fn_0[]’ (for “first name”) and ‘ln_0[]’(for “last name”), even if ‘cn_0[]’ is not filled. Or to say that a slot ‘nominal $group_0$[]’ has a slot ‘$noun_0$[]’, even if unfilled [2]. We even accept in our ontology documents that would be mere slot structures without any IF filling them or any of their slots (for example, my homework is currently only a blank page, but has already a predefined structure and thus some slots – see 5.6).

We accept the axioms stating that S is a strict order relation:

(AX5) **Slot-of irreflexivity** $\neg$Stt

(AX6) **Slot-of asymmetry** Stu $\rightarrow$ $\neg$Sut

(AX7) **Slot-of transitivity** Stu & Suv $\rightarrow$ Stv

4.1.3. Fillling and underfilling

We will now add the relation of underfilling (noted UF): a filler underfills a slot if it fills a slot of this slot:

(DEF3) **Underfilling** UFxt:$=_{def}$ ISt & $\exists$u (Sut & Fxu)

Trivially, something that underfills is a filler:

(TH3) **Only fillers underfill** UFxt $\rightarrow$ IFx

On the other hand, an underfiller of a slot t is not necessarily a part (in the sense of P) of something that fills t (since the larger slot t can remain unfilled). Contrast underfilling with proper parthood, as they look axiomatically very similar: x underfills t if x fills a slot of t *and t is a slot*; whereas x is a proper part of z if x fills a slot of z *and z is a filler.* We can show that both F and UF are strict orders, but those theorems are vacuously true (since the range and the domain of each of those two relations are disjoint).

4.1.4. Slot inheritance

The following axiom will play a pivotal role in our theory (and be discussed extensively in 5.1): if a filler x fills a slot t, any slot of x is a slot of t, and vice versa:

(AX8) **Slots of filler are identical to slots of the filled slot** Fxt $\rightarrow$ (Sux $\leftrightarrow$ Sut)

For example, if ‘John Doe’ fills ‘cn_0[]’, and ‘fn_{JD}[]’ is a slot of ‘John Doe’, then ‘fn_{JD}[]’ is also a slot of ‘cn_0[]’. Consequently, a slot may have a slot for two different reasons. First, it might be because of its own intrinsic structure, such as ‘cn_0[]’ having intrinsically the slots ‘fn_0[]’ and ‘ln_0[]’. Second, it might be because of the structure of its filler, such as ‘fn_0[]’ being filled by ‘Jean-Marc’ and therefore having as slots ‘$1hfn_{JM}$[]’ and ‘$2hfn_{JM}$[]’ (for respectively the first half of the first name and its second half), filled respectively by ‘Jean’ and ‘Marc’ (but it would not have such slots if it was filled by ‘John’).

Using AX8, we can show the following theorems:

- The slots of a part of an entity are slots of that entity too:

(TH4) **Slot of part inheritance** Stx & Pxy $\rightarrow$ Sty

Proof: Let's suppose that Stx & Pxy. Since Pxy, there is a u such that Suy and Fxu. From AX8 and Fxu, we know that for any v: Svx → Svu. Applying this to v=t, from Stx we can deduce Stu. From Stu and Suy, we deduce Sty by AX7 (**Slot-of transitivity**).

- If x is a proper part of y, then y is not a proper part of x:

(TH5) **Proper parthood asymmetry** PPxy → ¬PPyx

Proof: Suppose that PPxy and PPyx. Then Pxy and y fills a slot u of x. By TH4 (**Slot of part inheritance**), u is a slot of y. Thus, y fills one of its slots, which is impossible by AX3 (**No improper parthood slots**).

- If x is a proper part of y that fills t, then x underfills t:

(TH6) **Proper part of a filler underfills the filler's slot** PPxy & Fyt → UFxt

Proof: Suppose that Fyt and PPxy. By PPxy, there is a slot u of y filled by x. By AX8 and Fyt, u is also a slot of t. Thus, x fills a slot of the slot t. That is, x underfills t.

- An underfiller of a slot does not fill this slot (and vice-versa):

(TH7) **Underfiller of a slot does not fill it** UFxt → ¬Fxt

Proof: Suppose that UFxt and Fxt. Since UFxt, x fills a slot u of t. Since Fxt, any slot of t is a slot of x by AX8. Therefore, u is a slot of x. Thus, x fills one of its slots: absurd by AX3 (**No improper parthood slots**).

4.1.5. (Filler-)parthood as a partial order

From the above, we can deduce easily the following theorems (that follow respectively from DEF2, TH5 **Proper parthood asymmetry** and TH4 **Slot of part inheritance**):

(TH8) **Parthood Reflexivity** IFx → Pxx
(TH9) **Parthood Anti-Symmetry** Pxy & Pyx → x=y
(TH10) **Parthood Transitivity** Pxy & Pyz → Pxz

4.2. Filling axioms and theorem

Let's now add two axioms from which we can deduce new filling relations. The first one is a "descending" filling axiom: a slot of a filler is always filled:

(AX9) **Slot of a filler is filled** IFx & Stx → ∃y Fyt

We can deduce trivially from this axiom that a slot of a filler is always filled *by a proper part of the filler*:

(TH11) **Slot of a filler is filled by a proper part** IFx & Stx → ∃y (Fyt & PPyx)

For example, the slot 'fn_{JD}[]' of the filler 'John Doe' is filled by 'John'.

The second axiom is an "ascending" filling axiom: if all slots of a slot are filled, then this slot is also filled:

(AX10) **All sub-slots filled implies slot filled** ISt & [∀u (Sut → ∃x Fxu)] → ∃y Fyt

For example, if the two only slots 'fn_0[]' and 'ln_0[]' of 'cn_0[]' are filled, then 'cn_0[]' is filled (see 5.3 for a brief discussion).

4.3. Weak supplementation

Weak supplementation (and therefore strong supplementation also) clearly does not hold in our ontology. For example, ‘a’ is a proper part of ‘1st letter$_1$[‘a’] 2nd letter$_1$[‘a’]’, but there is no proper part of the latter that would not overlap ‘a’. However, we can accept an axiom akin to weak supplementation inspired by Bennett [4]. It states that if x occupies a slot t of the filler y, then y has a slot u that is neither identical to t, nor a slot of x:

(AX11) **IF Weak Supplementation** IFy & Sty & Fxt → ∃u [Suy & ¬(t=u ∨ Sux)]

We leave adoption of axioms akin to strong supplementation for future work (but see the discussion in 5.1.1 below). Note however an interesting point about extensionality: contrarily to classical mereology, two different fillers can have exactly the same fillers as proper parts, such as ‘1st letter$_1$[‘a’] 2nd letter$_1$[‘b’]’ and ‘1st letter$_2$[‘b’] 2nd letter$_2$[‘a’]’.

4.4. Consistency of the axioms

To provide a model, we will represent the structure of an IE by inclusion of rectangles (representing ISs) and ellipses (representing IFs):
- An ellipse x immediately inside a box t represents Fxt.
- A box t inside (not necessarily immediately inside, because of AX7 **Slot-of transitivity** and AX8) an ellipse or a box z represents Stz.
The following simple model on Fig. 1 (where unique names are used) satisfies all of AX1-AX11, therefore this axiomatic system is consistent:

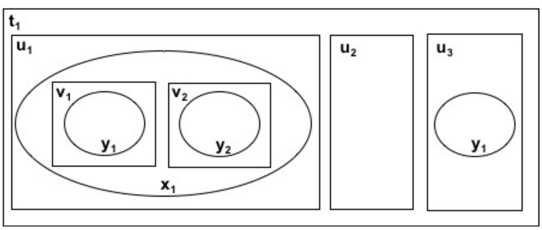


Figure 1. Model of the theory

Future work should investigate further the class of possible models, using a model finder such as Alloy.

4.5. Comparison with Bennett’s mereology

In the following, for an integer *n*, “A*n*” or “T*n*” (e.g. “A3”) refers to an axiom or theorem in Bennett’s system [4], whereas “AX*n*” or “TH*n*” (e.g. “AX3”) refers to an axiom or theorem in our theory. There are several important changes in the system presented above, compared to Bennett’s mereology. First, not all slots are “slots of something”: there can be “free-floating” slots. Second, a filler does not have an “improper slot” that it itself fills (cf. AX3, contra A4): intuitively, a slot is “smaller” than the thing it is a slot of. Third, there can be empty slots, and therefore, contra Bennett’s A7 that states that each slot has a single occupant, we merely accept a weaker axiom of maximum one occupant (AX4). Fourth, slots can have slots, contra A3 – and therefore, we introduced the notion of underfilling in DEF3.

Bennett deduces from her axioms that S is a partial order, but since slots do not have slots in her theory, those order properties are vacuous. In our system, those order properties are quite substantial (and necessary for many of our demonstrations), and we accepted them as axioms. Bennett also accepts as an axiom the Slot of Part Inheritance (A5). In our case, we accepted the more general axiom AX8 (which would have no sense in Bennett's theory, in which slots do not have slots); we then used it to prove Slot of Part Inheritance as a theorem (TH4).

We accepted a weak supplementation axiom on fillers (AX11) similar to Bennett's theorem T13 (which Bennett calls "Slot Weak Supplementation", whereas we called our axiom "IF Weak Supplementation", for reasons that will become clear in 5.1.1), which she deduces from an axiom of "Slot Strong Supplementation" (A8). Note that the last part of Bennett's T13 axiom reads "∃u (Suy & ¬ Sux)", whereas it reads for us "∃u [Suy & ¬ (t=u ∨ Sux)]", where t is the slot of y filled by x mentioned in the left-hand side: indeed, for Bennett, each entity fills one of its slots, but we exclude improper parthood slots, so we need to make sure explicitly that this supplementary slot u is not identical to t.

5. Discussion

We will discuss here several ways in which the system proposed above could be developed.

5.1. The identification of the slots of a filled slot and the slots of its filler

We have proposed above a relatively simple theory of ISs and IFs, that identifies by AX8 the slots of a filler with the slots of the slot it fills. However, this might raise challenges of two kinds.

5.1.1. Challenges

The first challenge is that the theory above would not fit well with a theory of diachronic evolution of the slot structure of a slot (but see the footnote in 5.6 that briefly discusses whether we want to have a diachronic theory of IEs in a first place). Suppose that the slot 'cn_0[]' has the slots 'fn_0[]' and 'ln_0[]', and that the filler 'John Doe' has the slots 'fn_{JD}[]' and 'ln_{JD}[]'. As soon as 'John Doe' fills 'cn_0[]', the slots 'fn_{JD}[]' and 'ln_{JD}[]' appear in 'cn_0[]', and the slots fn_0[] and ln_0[] appear in 'John Doe'. We would like to have means to equate fn_0[] with fn_{JD}[], and ln_0[] with ln_{JD}[] (or at least establish a mapping between them – see next subsection).

A second challenge pertains to supplementation. We have here accepted an axiom of weak supplementation involving both fillers and slots (AX11), similar to Bennett's. However, because our system accepts slots of slots, one might want to introduce axioms of supplementation purely at the level of slots, without any need of mediation by fillers (for an example of two-levels mereology, see the process specification language PSL, that endorses two mereologies, one at the level of activities through the relation "subactivity" and another one at the level of activity occurrences through the relation "subactivity_occurrence" [25]) . For this, we could first define a notion of overlap between slots as follows: s and t slot-overlap ("SO") just in case they share a slot u, one is a slot of the other, or they are identical (this latter mention is important, since a slot that does not have any slot does not share a slot with itself, but we want to state that any slot overlaps itself):

(DEF5) **Slot-overlap** $SOtu :=_{def} ISt \ \& \ ISu \ \& \ [\exists v \ (Svt \ \& \ Svu) \lor Stu \lor Sut \lor s=t]$

This would enable to formulate an axiom of strong supplementation among slots:

(AX12) **IS Strong Supplementation** $ISt \ \& \ ISu \ \& \ HSu \rightarrow [(\neg Sut \ \& \ t \neq u) \rightarrow \exists v \ (Svu \ \& \ \neg SOvt)]$

From this axiom, we could then easily deduce corresponding theorems of weak supplementation and extensionality on slots (using also AX6 to demonstrate the latter):

(TH12) **IS Weak Supplementation** $ISu \ \& \ Stu \rightarrow \exists v \ (Svu \ \& \ \neg SOvt)$

(TH13) **IS Extensionality** $ISt \ \& \ ISu \ \& \ HSt \ \& \ HSu \rightarrow [t=u \leftrightarrow \forall v \ (Svt \leftrightarrow Svu)]$

Ideally, we would want to use AX12 (**IS Strong Supplementation**) to prove **IF Weak Supplementation** as a theorem, instead of accepting it as an axiom (cf. AX11) as we did here. However, **IS Extensionality** is not compatible with our AX8. Indeed, if two slots are filled by the same filler that has slots, they have exactly the same slots by AX8; and by **IS Extensionality** (TH13), they would be identical. This would defeat the goal of this system that aims at enabling an IE to have a part twice (or more) over. To avoid this conclusion, one could relax AX8, for example by introducing a notion of "twin-slot".

5.1.2. Twin-slots

AX8 is equivalent to the conjunction of the two following axioms:

(AX8.1) **Slots of a filled slot are slots of the filler** $Fxt \ \& \ Sut \rightarrow Sux$

(AX8.2) **Slots of a filler are slots of the slot it fills** $Fxt \ \& \ Sux \rightarrow Sut$

To relax AX8, one could relax AX8.1, AX8.2, or both. However, if one (or both) of those axioms is abandoned, we may want to replace them by weaker axioms. For example, it seems sensible that because 'cn_0[]' has two slots 'fn_0[]' and 'ln_0[]', a filler x of 'cn_0[]' should have two slots 'fn_x[]' and 'ln_x[]'. To account for this and relax AX8.1, one could introduce the notion of "twin-slot" of a slot's slot. A minimal requirement for such twin-slot would be that if x fills t and u_t is a slot of t, then there is a slot u_x of x, called "twin-slot of u_t in x", such that u_x is filled by y whenever u_t is filled by y:

(AX8.1') **Twin-slot in a filler** $Fxt \ \& \ Su_tt \rightarrow \exists u_x \ [Su_xx \ \& \ \forall y \ (Fyu_t \leftrightarrow Fyu_x)]$

Using AX8.1', TH11 and AX4, we can show that a slot of t and its twin slot in x are filled by the same unique filler y, proper part of x:

(TH15) **Filling of a twin-slot in a filler** $Fxt \ \& \ Su_tt \rightarrow \exists u_x \ \exists! y \ [Su_xx \ \& \ Fyu_t \ \& \ Fyu_x \ \& \ PPyx]$

For example, if 'John Doe' fills 'cn_0[]', it has two slots 'fn_{JD}[]' and 'ln_{JD}[]' that are twin-slots of 'fn_0[]' and 'ln_0[]'. Both 'ln_{JD}[]' and 'ln_0[]' are filled by 'Doe'; but if 'cn_0[]' was instead filled by 'Jane Smith', then 'ln_0[]' would be filled by 'Smith', whereas 'ln_{JD}[]' would still be filled by 'Doe' (and it would then not be a twin slot of 'ln_0[]').

Similarly, one might also want to introduce the notion of twin-slot of a *filler*'s slot. Future work should determine which of those axioms, and/or others, should be endorsed.

5.2. Slot levels

We may want to define a hierarchy of sublevels among slots. For example, a diagnostic report that would occupy a slot s_0 could have the slots '$patient_0$[] $condition_0$[] $doctor_0$[]', where the slot '$patient_0$[]' is composed by the slots '$patient\_fn_0$[]' and '$patient\_ln_0$[]'.

Note that according to AX7 **Slot-of transitivity**, 'patient_fn$_0$[]' and 'patient_ln$_0$[]' are also slots of s_0. But we might want to state that the slots 'patient$_0$[]', 'condition$_0$[]' and 'doctor$_0$[]' are first-sublevel (or direct) slots of s_0, whereas the slots 'patient_fn$_0$[]' and 'patient_ln$_0$[]' are second-sublevel slots of s_0. We can define the various sublevels as follows (see [26] for an axiomatization):

- A 1^{st}-level slot of s is a slot of s that is not a slot of a slot of s.
- A 2^{nd}-level slot of s is a slot of a slot of s that is not a slot of a slot of a slot of s (etc.)

Note that in our axiomatization, there is no axiom that forces the existence of 1^{st}-level slots, 2^{nd}-level slots, etc. That is, there is no discreteness axiom that rules out the existence of a dense set of sublevels. Such axioms might be added in future work.

5.3. Mereological sum

We may want to formally introduce the mereological sum of several fillers of several slots. For example, elaborating on AX10, we may want to state that if all slots of a slot s are filled, the filler that fills s is the mereological sum of those fillers.

More innovatively, we may want to introduce an entity that would be composed by a filler and the slot it fills, such as the slot 'patient[]$_0$' and its filler 'John Doe'. This would indeed have consequences for aboutness (see section 5.4). A possible way might be to represent this as a mereological sum of the filler and its slot. Such an account could be compared to Koslicki's theory of the composition of material objects [27], which holds that objects have two proper parts: material parts and formal parts. As her analysis goes, for instance, Michelangelo's statue David is composed of an amount of marble (material part) and, say, the "David-wise structure" (formal part). Indeed, Koslicki states that a general notion of structure can be characterized as an entity that offers available "*positions* or *places*" (reminiscent of our "slots"). A mereological sum of a filler and its slot that would be completed by some kind of arrangement among subslots would thus seem to be very much in Koslicki's spirit.

5.4. Aboutness

We will not propose a formal theory of aboutness, but only give a few pointers of how we could extend IAO's theory of aboutness with an aboutness of ISs. We will consider here that an informational entity can be about several kinds of entities, such as a particular, a class, or a state of affairs [14]. This is relatively classical at the level of fillers. For example, in diag$_0$, 'John Doe' is about the particular human John Doe and 'Flu' is about the class *Flu*. However, we suggest that slots can also be about a variety of things. For example, 'patient$_0$[]' would be about the class *Patient* and 'condition$_0$[]' would be about the class *Medical condition*. Then, if we accept the mereological sum of a slot and its filler as explained earlier, the sum of 'John Doe' and the slot it occupies 'patient$_0$[]' might be about the relation of instantiation of the class *Patient* by the particular John Doe. Note however that not all slots are about something – consider e.g. the slot '2^{nd} letter$_0$[]'. Such considerations should be integrated into a full theory of aboutness of ICE – something that still needs to be developed in IAO.

5.5. Refusing supplementation

There might be reasons to refuse AX11 (**IF Weak Supplementation**) in our mereological system. Indeed, we might want to accept to have PPxy while y having as only slots the slot filled by x and the slots of x. For example, suppose that Mr. J's last name is composed by only one letter, 'J'. He needs to fill some administrative form that has the slot 'last $name_0$[]'. He fills it with 'J'. The slot 'last $name_0$[]' is filled with this filler 'J'_{Name}, that is about Mr. J. And this filler has one unique slot 'first $letter_J$[]', that is filled with the filler 'J'_{Letter}, which is a letter that is about nothing. That is, Mr. J fills two slots by drawing the same sign. And by AX8, 'last $name_0$[]' would have a unique slot 'first $letter_J$[]'.

However, following a suggestion by Masolo and Vieu [2], one might instead introduce a relation of composition between a word and the chain of characters it is made of, such that this relation would not be identical to parthood (for more on composition, see [28]). In such a case, 'J'_{Name} would be *constituted* by the character 'J'_{Letter}, but would not *have it as part*, and thus it would not be a counter-example to supplementation axioms. Moreover, such an approach could explain the change in aboutness when we move from letters to words.

5.6. The diachronic identity of documents and creation of ISs

The diachronic identity of documents is a topic that has been little studied. To illustrate its complexity, suppose that I start working on a homework. In front of me, I have a blank sheet of paper. At t_1, I decide that this paper will be the physical carrier of my homework: I decide that I will write my name on the top left, the date on the top right, the body of my text on the paper, divided in three parts. At t_2, I have written my name on the top left. and the date on the top right. At t_3, I have written the three parts of the body of my text.

It would be desirable to have a theory of identity according to which it is the same homework that evolves while I am filling it; that is, that there exists a unique homework IE at t_1, t_2 and t_3. The theory we developed earlier is compatible with such diachronic identity considerations: the homework remains the same document from t_1 to t_3, although some new parts (new IFs) appear[3].

Interestingly, several relevant ISs have arguably already appeared at t_1. Of course, we do not claim that something physically changed in the composition of the sheet of paper when I made this decision at t_1 to write my homework on this sheet of paper. The fact that it changed while not changing in physical structure (thus undergoing a Cambridge change [30]) only emphasizes the cognitive and social nature of ISs (something that is also true for IFs, as they depend on the existence of a semiotic system): it is my cognitive act (maybe mirrored in other cognitive agents) to structure my document with a name, a date, and a body of text in three parts that created those ISs.

Similarly, a database, a patient chart or a drug distribution report can remain identical while changing in content. Note that the number and nature of the ISs of an entity can change in two ways. First, trivially, IFs will themselves have ISs, which will, by slot inheritance, be ISs of the overall documents; therefore, new slots appear (consider again the "Jean-Marc" example in 4.1.4). Second, the ISs of a document can change depending on how agents change another part of the document. Consider for example a response sheet to a poll, where a positive answer to "Are you a national of another country than Canada?" will bring the IS to be filled by the nationality of the respondent.

[3] Note that this raises an interesting difficulty: if I make two copies of the homework at an earlier stage of development and then both evolve in different ways, it seems that they would remain identical despite evolving differently, which is counter-intuitive. This kind of problem has been studied by Parfit in the case of personal identity [29]. One possible solution, in a Parfitian spirit, would then be to abandon the notion of diachronic identity for informational entities.

5.7. Feature structures

As mentioned earlier, this system has analogies with the theory of feature structures [31]. The latter theory also introduces slots, that can be recursively filled by feature structures. A difference is that the root element in this theory is a feature structure, whereas in our theory, the root element is a slot (that is, a filler always fills a slot). Another difference is that feature structures are naturally interpretable as types, whereas fillers are first and foremost tokens (particulars) in our theory (although we can introduce classes of fillers). Finally, classical mereological constraints such as supplementation do not appear in the theory of feature structures.

6. Conclusion

We have introduced an axiomatic system for the mereology of informational entities using the notions of information slots and information fillers. Inspired by Bennett, this system is different in several important respects, in particular in having free-floating slots, slots of slots and empty slots.

This work has been extended by a companion paper investigating the notions of adequate vs. inadequate filling of a slot, as well as levels of slots, with a focus on clinical documents [26]. Important future investigations will include considering to relax AX8, introducing slot supplementation axioms and mereological sums, as well as analyzing the aboutness of slots and fillers. One could also investigate whether unfilled slots could be replaced or complemented by slots filled by an empty filler '∅': for example, a blank exam template might be unfilled before the exam sheets are distributed, whereas if I return my exam sheet empty, one could consider that I have filled the slots of this document with the empty filler. Finally, one should analyze the various forms in which IFs and ISs can be concretized, in particular when a message is conveyed as a process (e.g. spoken language or Morse code). This could strengthen the basis of ontologies of informational entities such as IAO, the ontology of document acts [32], and domain ontologies founded on them, such as the Prescription of Drugs Ontology (PDRO) [33] or the LABoratory Ontology (LABO) [34].

Acknowledgements

We thank four reviewers as well as Benoît Fraikin, Rosalie Plantefève and the audience at a seminar in Sherbrooke university in July 2020 for their insightful comments. FT acknowledges financial support by the SPOR Canadian Data Platform (CIHR).

References

[1] Barton A, Fabry P, Ethier J-F. A classification of instructions in drug prescriptions and pharmacist documents. Proc 10th Int Conf Biomed Ontol ICBO 2019. Buffalo, New York, USA; accepted. p. 1–7.
[2] Masolo C, Vieu L. Graph-Based Approaches to Structural Universals and Complex States of Affairs. In: Borgo S, Hitzler P, Kutz O, editors. Form Ontol Inf Syst Proc 10th Int Conf FOIS 2018. IOS Press; 2018. p. 69–82.
[3] Arp R, Smith B, Spear AD. Building Ontologies with Basic Formal Ontology. The MIT Press; 2015.
[4] Bennett K. Having a part twice over. Australas J Philos. 2013;91:83–103.

[5] Simons P. Parts: A study in ontology. Oxford: Clarendon Press; 1987.
[6] Casati R, Varzi AC. Parts and places: The structures of spatial representation. Cambridge, MA: Mit Press; 1999.
[7] Masolo C, Borgo S, Gangemi A, et al. The WonderWeb library of foundational ontologies and the DOLCE ontology. WonderWeb (EU IST project 2001-33052) deliverable D18. LOA-ISTC-CNR; 2003.
[8] Cotnoir AJ. Strange parts: The metaphysics of non-classical mereologies. Philos Compass. 2013;8:834–845.
[9] Tsai H. Decidability of mereological theories. Log Log Philos. 2009;18:45–63.
[10] Effingham N, Robson J. A mereological challenge to endurantism. Australas J Philos. 2007;85:633–640.
[11] Cancedda F. Functional requirements for authority data: a conceptual model: Final report, December 2008. Boll AIB 1992-2011. 2010;50:131–132.
[12] Peroni S, Palmirani M, Vitali F. UNDO: the United Nations system document ontology. Int Semantic Web Conf. Springer; 2017. p. 175–183.
[13] Bountouri L, Gergatsoulis M. The semantic mapping of archival metadata to the CIDOC CRM ontology. J Arch Organ. 2011;9:174–207.
[14] Smith B, Ceusters W. Aboutness: Towards foundations for the information artifact ontology. Proc 6th Int Conf Biomed Ontol. Lisbon, Portugal: CEUR Workshop Proceedings; 2015. p. 1–5.
[15] Rudnicki R. An Overview of the Common Core Ontologies [Internet]. Buffalo: CUBRC, Inc.; 2019. Available from: https://www.nist.gov/system/files/documents/2019/05/30/nist-ai-rfi-cubrc_inc_004.pdf.
[16] Garbacz P. A formal ontology of texts. In: Ferrario R, Kuhn W, editors. Form Ontol Inf Syst Proc 9th Int Conf FOIS 2016 [Internet]. Amsterdam: IOS Press; 2016. p. 345–358. Available from: http://ebooks.iospress.com/volumearticle/44240.
[17] Masolo C, Vieu L, Bottazzi E, et al. Social Roles and their Descriptions. In: Dubois D, Welty C, editors. Proc 9th Int Conf Princ Knowl Represent Reason KR 2004. Menlo Park, CA: AAAI Press; 2004. p. 267–277.
[18] Arora N, Westenthaler R, Behrendt W, et al. Information object design pattern for modeling domain specific knowledge. 1st ECOOP Workshop Domain-Specif Program Dev. 2006.
[19] Hawke P. Theories of aboutness. Australas J Philos. 2018;96:697–723.
[20] Bateman JA. Ontology, Language, Meaning: Semiotic Steps Beyond the Information Artifact. Ontol Makes Sense Essays Honor Nicola Guarino. Amsterdam: IOS Press; 2019. p. 119–135.
[21] Tarski A. The concept of truth in formalized languages. Log Semant Metamathematics. 1956;2:7.
[22] Corcoran J, Frank W, Maloney M. String theory. J Symb Log. 1974;625–637.
[23] Galton A. On generically dependent entities. Appl Ontol. 2014;9:129–153.
[24] Sattig T. Part, slot, ground: foundations for neo-Aristotelian mereology. Synthese. 2019;1–15.
[25] Grüninger M. Using the PSL ontology. Handb Ontol. Springer; 2009. p. 423–443.
[26] Barton A, Toyoshima F, Ethier J-F. Clinical documents and their parts. Proc 11th Int Conf Biomed Ontol ICBO 2020. accepted.
[27] Koslicki K. The structure of objects. Oxford University Press on Demand; 2008.
[28] Masolo C, Vieu L, Ferrario R, et al. Collective, composites and pluralities. In: Broadaric B, Neuhaus F, editors. Form Ontol Inf Syst Proc 11th Int Conf FOIS 2020. Bolzano, Italy: IOS Press; accepted.
[29] Parfit D. Reasons and persons. OUP Oxford; 1984.
[30] Mortensen C. Change and Inconsistency. In: Zalta EN, editor. Stanf Encycl Philos [Internet]. Spring 2020. Metaphysics Research Lab, Stanford University; 2020. Available from: https://plato.stanford.edu/archives/spr2020/entries/change/.
[31] Carpenter B. The logic of typed feature structures: with applications to unification grammars, logic programs and constraint resolution. Cambridge University Press; 2005.
[32] Brochhausen M, Almeida MB, Slaughter L. Towards a formal representation of document acts and resulting legal entities. Johanssonian Investig Essays Honour Ingvar Johansson His Seventieth Birthd. Walter de Gruyter; 2013.
[33] Ethier J-F, Barton A, Taseen R. An ontological analysis of drug prescriptions. Appl Ontol. 2018;13:273–294.
[34] Barton A, Fabry P, Lavoie L, et al. LABO: An ontology for laboratory test prescription and reporting. In: Barton A, Seppälä S, Porello D, editors. Proc Jt Ontol Workshop 2019 JOWO 2019. Graz, Autriche; 2019. p. 1–9.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200673

The Computer Program as a Functional Whole

C. Maria KEET [a,1]

[a] *Department of Computer Science, University of Cape Town, South Africa*

Abstract. Sharing, downloading, and reusing software is common-place, some of which is carried out legally with open source software. When it is not legal, it is unclear how many infringements have taken place: does an infringement count for the artefact as a whole or for each source file of a computer program? To answer this question, it must first be established whether a computer program should be considered as an integral whole, a collection, or a mere set of distinct files, and why. We argue that a program is a functional whole, availing of, and combining, arguments from mereology, granularity, modularity, unity, and function to substantiate the claim. The argumentation and answer contributes to the ontology of software artefacts, may assist industry in litigation cases, and demonstrates that the notion of unifying relation is operationalisable.

Keywords. Computer program, mereology, unity, granularity

1. Introduction

End users of Information Technology are familiar with software 'apps', or computer programs, that can be launched through a one- or double- mouse click or finger-tap action. From this user experience perspective, the computer program may appear to be one single entity. While it is not impossible that there is only one single file[2], with the increasingly complex programs nowadays, as well as modularisation practices over at least the past 25 years, there are typically multiple files involved in the running of an application for it to carry out its principal function. The components are stored in some specific directory and are artefacts such as plugins, icons for the interfaces, and configuration files that are typically flat text files or XML files. Disregarding those finer-grained details, one still may consider, e.g., TexShop or Firefox, or their respective source code, a single artefact that is downloaded and installed. Why so, or why might it not be a whole upon closer inspection? Why should one even spend time on answering this question?

There may be many practical and financial consequences following from the answer to such questions. This includes having been fought in court and arbitration tribunals, especially concerning copyright infringements, trade secret violations, and patent infringements. Recent and ongoing cases include, among others, Google vs Oracle on possible

[1]Corresponding Author: C. Maria Keet, Department of Computer Science, University of Cape Town, South Africa; E-mail: mkeet@cs.uct.ac.za

[2]though very rare: even code copied in the command line to print "Hello World" requires the built-in `print` function that is stored in another file than the hand-written code

copyright on APIs [1], Cisco vs Arista on copying parts of a computer network management utility tool where Arista eventually payed Cisco $400 million [2], and copyright claims on an enduser app in the health insurance domain [3]. A software litigation case that the author was called in as expert witness concerned, among others, a question essentially of an ontological nature. This is introduced here in a general way and with broader applicability. One party claims that for the computer program, there are the, say, $n = 1000$, individual source files that allegedly have been copied and wants the defendant to be fined for the 1000 copyright violations, once for each source file; hence, not, say, a €100 fine for the one infringement of the program, but a fine of €100000 altogether. The defendant, for obvious reasons, would rather prefer to pay just the €100 for one infringement, if that were to be deemed to apply according to the court. The argument may likewise be constructed for trade secret violations for stealing the intellectual property of an app, for pirated software that was illegally downloaded, and downloading textbooks in website format (which have multiple files cf. the single pdf option). It might be relevant also regarding the number of patent infringements, but this depends on the patent and the country; e.g., in South Africa, one cannot patent computer programs or algorithms.

On the surface, alleged infringements may sound similar to the well-known case of whether the collapsing bridge is one event or multiple collapsing events; however, here it first needs to be established how all those files of a computer program relate, which is not as straightforward as the structural components of a bridge. The main question to answer, then is:

Q1: Is a computer program or the source code a) a (tight) *whole* with *parts*, b) a whole that is a *collection* of *artefacts*, or c) just a *set* of *artefacts* where each element is a separate self-standing item?

This question generates two more specific ones to answer regarding the parts, in order to be able to answer Q1:

Q2: What is the relation between the files of a computer program (resp. source code) and the computer program (resp. source code)?
Q3: What is the relation among the files of a computer program (resp. source code)?

While it may be obvious intuitively to some that a program is a functional whole that has unity, that it generated a litigation case suggests that that view is not held throughout. There is ample documentation of ‘just so’ compositionality of programs—i.e., implying parthood relations—and why it is essential to good software design practices (e.g., [4, 5,6]) and to system design more generally [7]. Yet, to the best of our knowledge, there is no extant argumentation regarding the wholeness of a computer program or what it is that makes it a whole consisting of parts, why, and how from an ontological perspective. The more common question that is investigated in philosophy is the nature of a computer program, like whether it is a process or of some other category [8] or when an artefact can be classified as being a computer program, which are considerations *at that level of granularity* of the artefact, not about its compositional nature. Such arguments do, however, strengthen the argument that a computer program is a whole.

To answer the questions, we take insights from part-whole relations and mereology, the notion of unity, and basics of granularity, to combine them with modes of participation and functional parthood. Computer programs have a main function that is carried out by genuine functional parts and those parts have a specific unifying relation among each other at that finer-grained level of granularity, where one implies the other, therewith making a computer program a functional whole.

In the remainder of the paper, we first introduce some preliminaries in Section 2. We then consider the 'vertical' relation between the whole and its components in Section 3, the 'horizontal' relation between the components in Section 4, and close the argument that software is a proper (complex) whole in Section 5. We conclude in Section 6.

2. Preliminaries

There are multiple aspects one can investigate about computer programs and which mode is considered in such an analysis, such as the typed up source code files, the interpreted or compiled (machine) code that is the executable, or when it is actually running as a process on the computing device. This includes, among others, their ontological nature (whether it is, e.g., a process, a disposition, or an endurant), the nature and multiplication aspects of the 'information object' aspect of a program, identity criteria, and definitions. For instance, the Copyright Act 98 of 1978 of South Africa[3] defines "computer program" as "a set of instructions fixed or stored in any manner and which, when used directly or indirectly in a computer, directs its operation to bring about a result.". With limited space, it is possible to zoom in on only one aspect of computer programs. Specifically, it first needs to be established that the computer program is a whole, and why, regardless of what category that whole is of, if it is a whole. This is needed to inform debates about identity, determine the boundary, and it may assist determining the ontological nature of either mode of the program. Topics that are of relevance within this specific scope are:

- Part-whole relations, and in particular: i) Mereological parthood, and possibly any of its refinements; ii) Mandatory, immutable, and essential parthood; and iii) Collective entities (collections) and sets.
- Identity and unity.
- Granularity.

We introduce each one briefly informally for the reader unfamiliar with them, so as to keep the paper self-contained. The respective formal characterisations can be found in the literature cited. For the most part, those details are not needed for the argumentation on software wholes, unless stated otherwise.

Parts, wholes, and part-whole relations. Part-whole relations have been used and investigated in several fields of study, notably philosophy, ontologies for information systems, conceptual models, linguistics, and NLP (see [9,10,11] and references therein). This has resulted in a fairly stable list of common part-whole relations, including mereological parthood, refinements thereof, and informal ones in natural language utterances only but not mereologically [9]. The two key principles to organise them are transitivity (or not) of the relation and the category of entities that participate in the relation. Regarding the former, e.g.: if a cell is part of a heart and that heart is part of a human, then that cell is part of that human. An entity may thus play the role of 'part' in one relation and the role of 'whole' in another one. Regarding the latter, one could name the parthood that relates only processes to its part-processes as, e.g., *involvement* (i), specialising mereological parthood (p) that does not have any constraint on its relata,

[3]inserted through 1 (g) of Act 125 of 1992 (available at `https://www.gov.za/sites/default/files/gcis_document/201409/act125of1992.pdf`; last accessed: 16 July 2020)

i.e., $\forall xy(i(x,y) \rightarrow p(x,y) \wedge process(x) \wedge process(y))$. With an natural language utterance 'each Eating event *involves* a Swallowing event', one then identifies *involves* as a surface realisation of the *involvement* relation, and thus eating has as part swallowing.

Collectives fall within the scope of parts and wholes and part-whole relations. For instance, Project Team may be considered to be a collective consisting of human beings and one then may communicate that, say, "Dr. X is part of the project team". This is better characterised as a *membership* or *grain* relation between the part or grain Human and the collective Project Team rather than parthood, because transitivity cannot be guaranteed [9]. The nature of a collective entity has been investigated in fields such as conceptual modelling [12], biomedicine [13], and social theory [14], which may endow collectives with features or constraints on the whole or on its constituent parts. Widely-agreed upon key characteristics are that there is an identity to the collective as a whole, with some particular meaning, and it may survive its members, i.e., a member may change or be swapped but the whole keeps its identity. For instance, the rock band Queen has had three bass players before John Deacon, yet was already, and remained, Queen. An example of an additional constraint on the collective is that they can and do perform actions [14] and an additional constraint on a collective's parts may be that those parts must all perform the same role and if they do not, then the entity is more complex than a mere collective [12]. Collective nouns in natural language, such as a fleet and flock thus may, or may not, be considered collectives ontologically, depending on such additional constraints.

Collectives stand in contrast to mere sets, where a change in member changes the identity of the set, its members do not need a common 'binding' feature, and therewith a set does not need to have a specific meaning, and sets have no agency. For instance, your left foot and my laptop is a set with two objects as its members, but it is meaningless ontologically and does not instigate anything.

Orthogonal to the notion of types of part-whole relations is how the part and the whole participate in the relation, which might also be referred to as the 'strength' of the participation of the part and the whole. There are four principal options [15]: *i) optional*: the part P may, or may not, be part of the whole W, or the W may or may not have that P, e.g., a camera that may be part of a car; *ii) mandatory*: some P must be part of W, or vv., but not necessarily that one, e.g., for a house to be a house, it has to have a roof as part, but the roof can be renewed, while still being the same house; *iii) immutable*: that particular P must be part of W (or vv.), for as long as W is in a particular role, e.g., a boxer must have as part his own hands and if he loses a hand he ceases to be a boxer but as human they may continue to live; *iv) essential*: that particular P must be part of W and if it loses that P, the W will cease to be W, e.g., how a particular brain is part of a human.

There are further notions of sharability of the part, which is closely linked with the notions of participation: *i) sharable concurrently*: some P may be part of more than one W at the same time, e.g., a talk may be part of a course and of a seminar series; *ii) sharable sequentially*: P is part of one W at a time but can be part of different Ws at different times, e.g., a removable network card may be part of one PC at one time and part of another one another time; *iii) exclusivity*: P can be part of at most one W, e.g., entities such as the brain, spinal cord, or thalamus as part of a human body.

Identity and unity. Ascertaining the identity of an entity is about examining something as being the same or different at one point in time (synchronic identity) or across time (diachronic identity). To establish the identity of an entity, one uses *identity criteria* that the entity has, which is a stronger notion than *identification criteria* (typically a single artificial attribute) that may be assigned to entities. For instance, one may identify *Tibbles*, the neighbour's cat, by the colour of its fur, its size, and how it meows, whereas artificial attributes of entities, such as one's ID number and the MAC address of a network card, assists with identification. Within the context of programs, it may be of use to also note "striking resemblance" or "comprehensive non-literal similarity" of program code[4], which holds when two pieces of code are identifiably different, but *the semantics/logic of the expression of the idea* is the same. The two typical examples are 1) to take some piece of code and change the names of the variables: they are different documents, but have striking resemblance; 2) to keep all the business logic of the code but translate it line by line from one programming language into another language (say, Java to C++).

Unity [16,17] is a special case contributing to identity, which focuses on the relation that the parts have that make up the whole entity that has some identity. *Unity criteria* are those criteria that have to hold among the parts for it to be a whole. Depending on the nature of the unifying relation, one can identify different types of wholes. Some typical examples are *topological wholes*, such as a tree and a heap of sand, *morphological wholes*, such as a bouquet of flowers and a constellation of stars, *functional wholes*, such as scissors and a bikini, and *social wholes*, which are certain types of collectives, such as a population of a country and a herd of sheep. There is 'something' that binds the parts together to be a meaningful whole with an identity, which stands in contrast to, notably, stuffs (typically indicated with mass nouns, such as water and wood) and arbitrary sets of entities like the set of your left foot and my laptop. Since we only deal with computer programs, we exclude stuffs or amount of matter from the possible relata henceforth.

Guarino and Welty sought to formalise the notion of unity, with a good start made in [16]: with B the generic unifying relation that binds the parts with respect to the whole, and P mereological parthood that is temporally indexed (presumably with time $t \in \mathscr{T}$ as discrete time), their first condition is that

$$\forall xyzt(P(x,y,t) \rightarrow (P(z,y,t) \leftrightarrow B(x,z,t))) \tag{1}$$

Informally: if there is a parthood between x and y as some time (t), then at the same time it holds that for any entity z, that is also part of y at that time, it is in a unifying relation with the part x, and vice versa.

Eq. 1 and the description in [16] assume weak supplementation would apply, under the assumption that $x \neq z$, which they specified as $\forall xy(PP(x,y) \rightarrow \exists z(PP(z,y) \wedge \neg O(z,x)))$ (every proper part PP must be supplemented by another, disjoint [not overlapping $\neg O$], part), because of the implication: there must be another part of y, being z, so that they can relate through B. Guarino and Welty want to exclude any mereological sum that is just a contingent whole. This can be seen in analogy to the difference between mere sets and collectives. Noting a typo in their formalisation in [16] (corrected in their lecture slides on OntoClean), and adding quantification, we formalise it as

$$\neg\forall xzt(B(x,z,t) \leftrightarrow \exists y(P(x,y,t) \wedge P(z,y,t))) \tag{2}$$

That is, it does not hold for all x and z that when they are related through B at time t they are also a part of y, or: x and z may be bound in some way, but they are not therefore also part of a whole y because of that. This raises a few questions about the

[4]The famous cases where this got established are Franklin vs. Apple (1984) and Whelan vs. Jaslow (1987).

exact configuration when Eqs. 1 and 2 are taken together. For instance, can or do they have to be different Bs that relate x and z? Can they be part of different wholes, or: are the ys in Eqs. 1 and 2 supposed to refer to the same whole (which they currently do not)? Weak supplementation aims to rule out the possibility that $x = z$, in that there cannot be a whole with a single *proper* part, but unity is declared with P, not PP, so would unity still apply just in case there is only one part? These finer-grained details about the configuration brings us to the last preliminary: granularity.

Granularity. This refers to the level at which one operates or analyses and that there are different levels [18]. For instance, at level L_i, some amount of water is a type of stuff, but at its finer-grained level L_{i-1}, there is a set or collection of H_2O molecules, or at L_i one has an instance of Team and at L_{i-1} are its grains, Human Beings. The level of analysis may be of importance ontologically. For instance, Organisation is a social entity at level L_i that has one or more instances of Human as members at L_{i-1}, but it is only the former that must have a bearer with a single physical address that holds for that entity at that level of granularity, and that attribute is not inherited by its constituent parts, since, e.g., the company's employees typically live elsewhere. Similarly, it is the program that has, at that level of granularity L_i, the property of directing the computer's operation to bring about a result. That is, the whole entity at L_i has properties of its own, which can be very different from its parts or grains residing at L_{i-1}.

Note that granularity focuses on the more/less details of analysis, which is different from prioritisation at the same level of granularity. For instance, there may be direct parts x, y, and z at level L_{i-1} of whole w at L_i and one is interested only in x and y. This does not push z into a finer-grained level L_{i-2}, since an ontologist's interest is a separate matter.

3. Software components and their relation to the source code and program

We first discuss some important features of computer program as entity, with its parts, construction and use, and how those files relate to the source code or machine code. This will be illustrated with a relevant program that the author contributed to and whose principal components are shown in Fig. 1: it verbalises an ontology in isiZulu [19]. Also, henceforth, a distinction is made between the *source code* as artefact and the *compiled code* as artefact, since multiple source code files may compile into one compiled artefact that is a digital file. For compiled code, one also can include interpreted code, since the arguments also hold for it as it is also executable machine code. x source code files may result in y (with $x \neq y$) files of the software application, and the application has one 'point of entry' to launch it irrespective of the number of files it needs for its proper operation.

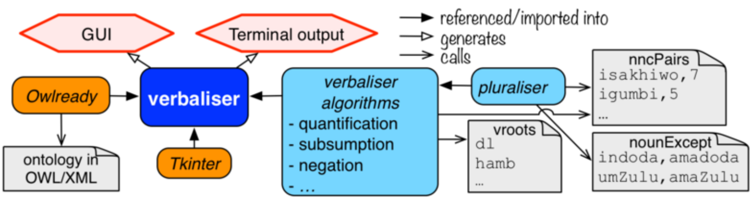


Figure 1. Principal components of the OWL verbaliser for isiZulu (source: based on [19]).

3.1. Modularisation and compositionality

Good design principles entail *modularisation* and *compositionality*—be it for software engineering or other engineering (e.g., [4,5,6,7]). Informally: this means breaking up a large problem into smaller ones, to devise solutions for the smaller problems and to put those together—declaring relations or rules-as-relations to specify how the components interact—to solve the larger problem. Benefits of this approach include, among others, manageability, reusability, quality control, and maintainability. For instance, with the isiZulu verbaliser (Fig. 1), the Python module Owlready is reused, and the vocabulary is easy to maintain by just updating a text file. A mereology-based approach joined with category theory for multiple types of part-whole relations has been applied to component-based software architectures [20], but not at the level of code. The principle can straightforwardly be extended by assuming that each component of the architecture is realised by one file. Practically, each component may be divided up into smaller components to realise the purpose of the component, which brings us to the next point.

There is a difference between what the artefact is designed to do, i.e., its function, and *how* that is realised. This follows directly from the modularisation, since there may be multiple ways of breaking up the design into smaller components. For some simple artefact, such as a hammer, these options are limited, but this is not so for software artefacts. Even a fairly straight-forward webpage, where all content could be declared in one single HTML file, can be split up into multiple files: the text to be displayed written in an `.html` file, the layout relegated to a style sheet in `.css` called from the `.html` file, and some javascript in a separate `.js` file also called from the `.html` file. For programs, e.g., Java generates separate `.class` files for each class declared in the code—fewer classes thus result in fewer files, and while a single C source file will result in a single program file, in the intermediate steps there are other files and processes involved: when compiling the source file into machine code, `.lib` files are called by the linker to generate the program file. It is possible to devise source code that consists of one file, does not make sense to divide up, and does not import anything, but this is a corner case exception rather than the norm (e.g., it may appear in an introduction to programming course, in the first lecture), especially for production-level computer programs.

Related to function and modularisation is the *user experience* of the computer program, where a function may be realised by more or less well integrated components that may be automatically invoked or manually started. Manual actions may give the user the impression of it being distinct but related components, whereas automated processes will provide the impression of it being one entity. This may be by design. For instance, one could by default block the running of scripts in spreadsheets, so when there is one, the user has to approve of its execution; when running scripts is enabled by default, on the other hand, end-users will not be aware that it is executing the script as a sub-process of the spreadsheet process or even that there is a script running.

3.2. Mode of participation in the relation of the files in the software

We have to consider the mode of participation in the relation between the files of a program, and, more precisely, assess how the optional, mandatory, immutable, and essential participation apply to source code and the compiled program. We will address this under the assumption that the program is free of bugs.

Optional participation of a file: In the interest of optimal design, there should be no optional files either in the source code or in the running program—if it can continue to operate without any bugs without that file, then that file is redundant and should be removed. Nonetheless, there very well may be sub-optimal source code and programs, especially when some modules or libraries are imported: operationally redundant files may not have been removed to save oneself testing whether it could be removed. For the running example: the programmers did not check whether all the files of the imported Owlready are used. Probably they are not, since the tool only fetches data from the OWL file but does not modify it, but there are functions stored in separate files to modify the ontology, such as `instance_editor.py` that is part of Owlready.

Mandatory participation of a file: the file has to be present, but it can be swapped for another without altering the function of the program. For the running example, this applies as well: the code looks up nouns in the `nncPairs.txt` file for any axiom it has to verbalise. If the file is not there, it will throw an error, but it can be overwritten with a different file also called `nncPairs.txt` that has the same outward function and minimal (but possibly more) content, e.g., one that has more nouns with noun classes listed in the file. The function remains the same regardless.

Immutable participation of a file: applying the definition of immutable, then when the program performs a particular role, it is essential to it. It is debatable whether this mode of participation would be applicable for either source code or compiled code. If at some point it has to perform some function *x*, then it is part of the specs at all times. The verbaliser has two 'modes' of operation—terminal-based and an end user interaction with a GUI. Tkinter is only essential for the GUI operation mode, which still performs the essential function of verbalising ontologies, so `Tkinter` might be an immutable part of the verbaliser. Yet, if the requirement is 'to be able to operate in two modes, terminal and with end user formatting', then omitting `Tkinter` breaks the overall functionality, and would therewith be at least mandatory if not essential, rather than immutable.

Essential participation: remove that particular file and the program is broken. In praxis, there are different levels of 'broken', such as minor bugs but the main function still can be achieved, major issues so that it only works partially, and the program not being able to run at all. From the binary viewpoint of 'bug-free or not', these differences do not matter: if not all specified functions work when that particular file is removed, then it is an essential file. For the OWL verbaliser, this may hold for Owlready: if any other version than `Owlready-0.3` is not backward or forward compatible, then that version is essential to the function of verbalising ontologies by the isiZulu verbaliser.

In sum, for bug-free operation, straight-forward mandatory participation in the (presumably part-whole) relation is the default case, with optional participation amounting to time-constrained coding and essential participation would be brittle coding.

Regarding sharability of a program's files: it depends on whether a file has to have a lock on it when open because of concurrency. If so, then that component file of the program is only sequentially sharable. Due to the unclear ontological status of programs, exclusivity is debatable depending on the choices—among others, whether copies count as distinct and whether it is the file as information artefact, etc. It does not matter for the notion of whole, and therefore we leave this aspect for future work.

3.3. Functional parthood

Since there is the compositionality thanks to modular design, and the types of files with their contents are distinct rather than subsets, the applicable relation between those component files and the whole—Source code or Executable [compiled/interpreted] computer program—is that of *parthood*. One may argue that it would also be a *proper* parthood in most cases, since it is unusual for production-level software to consist of one file only, where the part-files may or may not be shared either sequentially or concurrently, depending on the exact configuration of the code.

Since multiple part-whole relations have been proposed, this relation may be refined at some stage. Currently, however, because the ontological status of a file and of a computer program is not fully settled [8], a refinement along the line of [9] is not possible yet. The program does have a specific assigned function, however, such as Text processing for *MS Word* and Song management for *iTunes*, which each have sub-functions, such as Text formatting, Printing, and Playlist creation, i.e., there are functions and part-functions, and they in turn may have part-functions, such as Change text to bold face, Add song to playlist etc. Many types or categories of function have been identified and the ontology of function [21] identifies 89 of them. Also, several definitions of function have been proposed to assist with specifying functions. The Oxford Dictionary[5] states that function is an "activity that is natural to or the purpose of a person or thing". Yet, a function of an object might never be realised and so then cannot be an activity that is happening; e.g., a newly bought hammer may never be used. Also, an object can have one intended function but may be used for something else; e.g., an old cooking pan may obtain the function of flower pot. These sort of complications have been investigated in detail and the aim is to avail of that here and use it as-is, rather than digress in details. Mizoguchi et al. [22] did so most recently, who define function as "a role played by a behavior specified in a context", after observing that a "function is necessarily supported by the structure and/or properties of the things" and that "one of the most significant properties which function must have is implementation-independence." [21]. While this definition is less straightforward than the Oxford one, it is more accurate and at least it is broad enough to include the hammer and cooking pot cases or, say, repurposing a Web browser in full screen mode as a presentation application.

Taking function as such and proceeding with the 'vertical' relation between the part and the whole, then other than to realise the overall function, there must be component functions that contribute to that and the structures that realise that would thus be "functional parts" of the whole in some way [23,24]. The definition of *functional part* by Mizoguchi and Borgo [23] is: "Given an entity A and a behavior B of it, a functional part for that behavior is a mereological part of A that, when installed in A, has a behavior that contributes to the behavior B of A." This is somewhat underspecified when taken in isolation, since it has to be seen in the context of their definition of function. We refine the definition as follows:

Definition 1 (Functional part) *Given an entity x and a behaviour of type B of it that plays the role of function F in that context c, a functional part for that behaviour of type B is a mereological part of x that, when installed in x, has a behaviour of type D that contributes to B of x, where D is plays the role of part function F'.*

[5]`https://www.lexico.com/definition/function`; last accessed on 14 July 2020)

Mizoguchi and Borgo informally describe four types of functional parts; it is clear from the context in the paper, that it is to be understood as *proper* part. A part is a *genuine* functional part if it is installed correctly in x, from a structural viewpoint; a *replaceable* functional part is as specified for ‘mandatory’, above; a *persistent* functional part is either an essential or an immutable part; and a *constituent* functional part is a generic part regardless its assigned position as it may be temporarily taken out physically [23, p6]. A possible further interpretation with respect to a role theory, as alluded to in [23], may be of interest, but these four listed already help clarify the general modes of participation for files as expounded upon in the previous section. Under the assumption that source and compiled code each have more than one file and there are no optional files, then

- Computer program source code: all files are genuine functional parts and replaceable. The files have to be in specific locations to be called.
- Computer program compiled/machine code: all files are genuine functional parts and possibly also persistent (if not, then they are replaceable). The files have to be in the right folders and only there.

This settles the ‘vertical’ relation between the parts and the whole. We will turn to the relation among the parts in the next section.

4. The relation that binds the components

Taking stock, the files participate mostly mandatorily in the genuine functional parthood relation with the source code or compiled and executable program, normally there is more than one file involved, and each file either contributes to some function or performs one or more functions to contribute to the specified function(s) of the program. In terms of applying Eq. 1, a first step would be the following:

$$\forall xyzt(genuineFP(x,y,t) \rightarrow (genuineFP(z,y,t) \leftrightarrow B(x,z,t))) \quad (3)$$

Observe that it is time indexed, since it was so in the original (Eq. 1) and there is no harm in keeping the door open for code versioning; hence, it is more precise and stable. This brings us to question Q2 posed in Section 1: resolving the “unifying relation” B for the source code and the compiled program, and related loose ends from the formalisation of unity in Section 2. Characterising B from Eq. 3, i.e., the relation among the parts, ensures we arrive at either it being a (functional) whole since it has unity, or, if it cannot be specified, that then there is no whole after all. The exposé on unity in [16] does not have even one example of such a unifying relation, however, nor do the OntoClean examples [17]. Others, such as [14,25], do not fare better, like with a mere “x is unified in the right way” [25, p180], or go the other extreme with non-reusable situation-specific instance-level relations like “carrying out research in the same sub-area of the area of distributed systems in the University of Twente” [12, p159]. For instance, while Bikini is a functional whole consisting of two parts, the particular type of unifying relation that holds between the two parts, for any bikini, is left undetermined. Assessing programs, this unifying relation needs to be ascertained, however, since it would help explain *why* it is a whole.

Let us first consider two cases with common sense examples. For a morphological whole, such as the Constellation example (e.g., the stars making up the *Sagittarius* sign), B might be that all the stars involved share that they participate in the connecting the drawing lines relation to make up the figure. For social wholes, such as Electorate of a country, its parts or members of the collection—the voters—share that they stand in a relation fellow adult citizen of some country. Applying this to Eq. 1, then the latter example would be formalised as follows: $\forall xyzt(VoterIn(x,y,t) \rightarrow (VoterIn(z,y,t) \leftrightarrow fellowAdultCitizen(x,z,t)))$. So, if The Daily Show presenter *Trevor Noah* is a voter in the *Electorate of South Africa*, then if there is another voter, say, the national rugby team's captain *Siya Kolisi*, in the *Electorate of South Africa*, then *Trevor Noah* and *Siya Kolisi* are fellow adult citizens and vv.; hence, it can be applied.

Since the notion of unity seems to be usable, let's now consider computer programs: what is it that binds the files, other than the vague 'being another component'? Since source code is compiled, with as consequence that the files comprising the source code are different from the file(s) comprising the compiled program, the relation among the file is, perhaps, different. In both cases, we use the 'realistic case' argument, i.e., setting aside the corner case of one main file without calling anything else ever, since we are interested in practical applications. We shall address each in turn.

Recall that the whole is the source code for the computer program, such as *Firefox source code*, which operates at L_i level of granularity, regardless how that is organised. The components and the organisation thereof—i.e., the files that make up the source code like a file with the `main()` and imports—operate at L_{i-1} level of granularity. This is likewise for the computer program, such as the *Firefox executable app,* with the compiled code. There are various ways to link files to the main file; e.g., through an `import` statement in, among others, Python and Java, an `#include` in C and C++, and other coupling mechanisms [5, ch9]. An example of cascading imports is depicted in Fig. 2, which form a tree that may have files with more than one parent. Mathematically, it is a directed graph with a top-node and the other nodes may have more than one incoming edge and more than one outgoing edge, and ideally it is acyclic. A circular import may not be forbidden by the syntax of a programming language, but it is considered an anti-pattern and, consequently, there is an *Acyclic Dependencies Principle* with strategies for breaking cycles to foster good design [5, ch9]. Any two component files of the source code, being genuine functional parts, thus relate through that directed graph of dependencies, either further down in the hierarchy or upward and into another branch. One thus always can construct a path between any two nodes (ignoring the direction of the edge). Let us call that the $SCgraphPath$ property, then Eq. 3 can be updated into Eq. 4:

$$\forall xyzt(genuineFP(x,y,t) \rightarrow (genuineFP(z,y,t) \leftrightarrow SCgraphPath(x,z,t))) \quad (4)$$

We still need to verify it is not just a contingent whole; either the following holds:

$$\neg\forall xzt(SCgraphPath(x,z,t) \leftrightarrow \exists y(genuineFP(x,y,t) \wedge genuineFP(z,y,t))) \quad (5)$$

or, at least, that it holds with Eq. 2's generic P. Is it possible that x and z are (genuine functional) parts of some whole y but not do stand in at $SCgraphPath$ relation to each other, or v.v.? This is indeed very well possible: they may be parts of, e.g., a library y of header files or x and z are in a repository of modules y. In that case when in a library or repository, x and z are neither in the $SCgraphPath$ relation nor, arguably, $genuineFP$ but another part-whole relation, since there they are members of a collection.

Lastly, Eq. 1 implicitly assumed weak supplementation, although not formalised as such, and thus that there ought to be more than one source code file. There exist source code that has no `import` or `#include` statement, although this is rare for production-

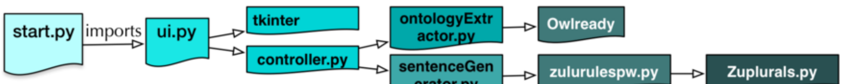


Figure 2. Series of actual `import` statements of the isiZulu OWL verbaliser, for the Python files only (it calls other files as well, not shown).

level software. Still, with the current formalisation, this is not a problem, since 1) weak supplementation is asserted for *PP*, not *P*, but *P* is used in the unity axiom and 2) for "$\neg\forall xz$" in the axiom to hold, even just one example satisfies it, which we just have with the module library; hence, this is not a problem even for the corner case.

We thus can define source code to be a functional whole as follows:

Definition 2 (Program source code as a functional whole) *For any computer program (y) in source code form that has one or more genuine functional parts* $x_1,\ldots,x_n$ *with* $n \geq 1$ *and* $1 \leq i,j \leq n$*, and time* $t \in \mathscr{T}$ *where* $\mathscr{T}$ *is the set of discrete time point, it holds that* $\forall x_i x_j y t(genuineFP(x_i,y,t) \rightarrow (genuineFP(x_j,y,t) \leftrightarrow SCgraphPath(x_i,x_j,t)))$. *For at least some whole (y′),* $SCgraphPath(x_i,x_j,t)$ *does not hold, i.e.,:* $\neg\forall x_i x_j t(SCgraphPath(x_i,x_j,t) \leftrightarrow \exists y'(genuineFP(x_i,y',t) \wedge genuineFP(x_j,y',t)))$.

Next, how can Eq. 3 be finalised for the executable program? At that stage, the explicitly stated to be imported files have been included, so *SCgraphPath* is not applicable. Yet, there still may be multiple files that are part of the computer program that "brings about a result"; examples include an `.ini` file for initialisation setting, an `.xml` file with the locale setting for menu options in a different language, and a dictionary to load for spellchecking. Some of these files could have been put 'inside' the executable, but from an engineering and usability viewpoint, they allow for more flexible customisation of the program and *shareability* of files when accessible separately. In any case, regardless the variation in configuration, if there is more than one file for the compiled code, they are *linked*—in the sense of the compilation processes in compilers, not the generic term 'linked'—so that for compiled code, we obtain a similar definition as for the source code functional whole, but then $linked(x_i,x_j,t)$ rather than with $SCgraphPath(x_i,x_j,t)$. There are several differences between the two relations: the latter is mutable and the former is not; the latter can have those files swapped, the former not; they hold between different types of files; and the latter has the links manually specified by the programmer's design, whereas this happens automatically in the compilation by the optimisation algorithms.

Can x_i,x_j be linked but not be part of a computer program y, or be genuine functional part but not linked, i.e., that the bi-implication does not hold? For code, there are *object code libraries* that are used in the linking stage of the compilation process, which x_i,x_j are part of and while being part of that, they are not linked. With sub-optimal program code and that optional participation, surely also at least one example can be found where either x_i or x_j is not a *genuineFP* of y. We thus obtain the following definition:

Definition 3 (Compiled program as a functional whole) *For any computer program (y) in compiled form that has one or more genuine functional parts* $x_1,\ldots,x_n$ *with* $n \geq 1$ *and* $1 \leq i,j \leq n$*, and time* $t \in \mathscr{T}$ *where* $\mathscr{T}$ *is the set of discrete time points, it holds that* $\forall x_i x_j y t(genuineFP(x_i,y,t) \rightarrow (genuineFP(x_j,y,t) \leftrightarrow linked(x_i,x_j,t)))$. *For at least some whole* y'*,* $linked(x_i,x_j,t)$ *does not hold, i.e.,:* $\neg\forall x_i x_j t(linked(x_i,x_j,t) \leftrightarrow \exists y'(genuineFP(x_i,y',t) \wedge genuineFP(x_j,y',t)))$.

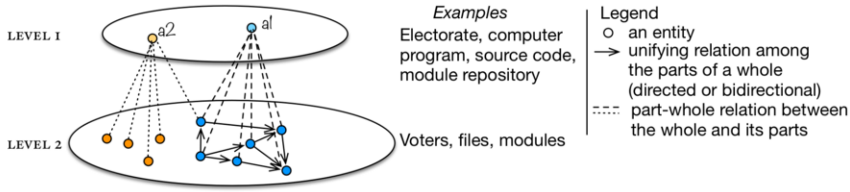


Figure 3. Levels of granularity, with entities residing in each level, which may be related 'horizontally' at the same level and 'vertically' between the parts and the whole residing in different levels. The orange object a2 visualises a collective with its members and the blue a1 visualises an integral whole with its parts.

The situation is visualised in Fig. 3, which shows two of the possibly many levels of granularity, each having entities residing at that level. The 'vertical' relations between the whole at level L_1, and its parts at the finer-grained level L_2 are typically a part-whole relation, but may be different [18]. The relation among the parts at a particular level is the unifying relation, such as being in a source code graph and begin a fellow adult citizen of a country. Informally, this constructs a triangle of relations between the entities: if you have one vertical side, there must be the other two.

5. Computer programs as wholes, collections, or set?

Having answered Q2 and Q3—*genuineFP*, *SCgraphPath*, and *linked*—we now can turn to Q1: is it a whole, a mere collection, or just a set? It has been argued colloquially that OH Dean puts forth in his Handbook of South African Copyright Law that the component files of a computer program should be considered to be a set of independent files for the purpose of copyright infringement assessments. However, clearly, a program is not just a set of discrete independent files, for the files have a specific relation to each other, and they need to have that tight coupling to achieve the proper functioning of the program. Sets, on the other hand, do not pose any such conditions on their members.

That it is not a collection either may be less obvious, and the arguments may depend on the definition of collection. With the requirement on the parts or grains to have to perform the same role [12,13], the program is *not* a collective, because its components do not have all the same function or role nor are they necessarily the same type of file. Moreover, a crucial aspect of modularisation in software development is to separate the different functions into different modules and files. Thus, when following established software design principles, it would never be a collective with this requirement.

Considering then Copp's criteria for collective [14]: the program can and does perform actions, therewith meeting one of the three criteria. He focusses on *social* collectives only, and therefore also proposes the requirement that a collective is composed of persons and has a "plausible theory of the legal system", which clearly do not apply. He generalises collective so that possibly they are "mereological sums of collective stages linked by a unity relation" where the "unity" refers to a property of the whole to track the collective through time to determine diachronic identity and the mereological sum applies to aggregating the time-slices, not about the parts being summed into a whole. Diachronicity is not applicable here as a criterion, however, supporting yet further the notion that software is not a collective. Thus, also by Copp's criteria and definition, a computer program is not a collective.

It does raise again the topic of mereological sum: could a program be just a mereological sum? Bare mereological sums permit a sort of ‘contingent unity’, however, like a sum of your right index finger and right thumb. This is not the case with program code, since each unity is *deliberate* by design and has properties at the level of the whole. Guarino and Welty tried to avoid simple mereological sums by assuming that “An object x is an *intrinsic whole under* ω if, at any time where x exists, it is a contingent whole under ω.” [16] where their ω is B here. That is, B is essential to the whole, whereas just a mereological sum need not to have B holding among its parts all the time or even at some time. For instance, a Python module repository may contain the modules Owlready v0.3 and NLTK v2, which is a mereological sum and also a collective, yet the modules are independent from each other and can be removed; the files in one’s ‘Downloads’ folder have even less unity. The files of a particular version of the source code or compiled program, however, *always* will have a binding relation to each other for as long as they exist; hence, then indeed is an intrinsic whole under $SCgraphPath$ or $linked$, respectively.

Having used both the process of elimination and having provided supporting arguments as to what makes a whole a whole (recall Definitions 2 and 3), a computer program—be it in source code form or as compiled code—qualifies as being a *whole*. It is a *functional* whole, since the function aspect comes from the whole having a function, which has ‘sub’ or part-functions (recall Section 3.3), as per design of the artefact. Thus, then the source code and the computer program are a *functional whole*. The definitions implicitly also indicate the *boundary* of a program: those files in the source code graph/linked are surely ‘in’, with their function calls and inclusion by reference. That means other files not in that graph are ‘out’. Notably, there are also loose coupling mechanisms, such as system calls for an application to request a service from the OS directly or through an API and pipes between programs. Excluding such loose coupling mechanisms is also in line with GNU’s licensing interpretations on open source software for distinguishing between whether there are two programs or one with two parts[6]. If such loose coupling were to have to been included and somehow the OS would become part of the app due to a system call or API to that extent, then no non-open source program would legally be allowed to run on an open source OS, which is clearly not the case. More consequences may ensue from the definitions and arguments provided, which is left for future work.

6. Conclusions

An argumentation was presented that a computer program—be it as source code or as compiled (machine) code—is a functional whole, and why. The unifying relation among the parts (files) is the graph for source code and being linked for compiled code. The relation between the component files and the program is one of functional parthood, since the files perform (a) subfunction(s) of the function of the program. These additional insights into computer programs and the notion of the internal structure of wholes may assist practically with, among others, litigation cases in software development, illegal downloads, and copyright infringements, as well as more generally by having demonstrated that the notion of a unifying relation is indeed operationalisable.

[6] `https://www.gnu.org/licenses/gpl-faq.en.html#MereAggregation`; last accessed: 16-7-2020.

References

[1] Samuelson P. API copyrights revisited. Communications of the ACM. 2019;62(7):20–22.

[2] Brachmann S. Arista pays cisco $400m to end patent litigation at district court and itc; 2018. Last accessed: 8-6-2020. *IPwatchdog*. Available from: `https://www.ipwatchdog.com/2018/08/08/arista-pays-cisco-400m-end-patent-litigation-district-court-itc/id=100102/`.

[3] Cairns P. Does afrocentric have a R1bn problem?; 2018. Last accessed: 8-6-2020. *The Citizen*. Available from: `https://citizen.co.za/business/1979904/does-afrocentric-have-a-r1bn-problem/`.

[4] Bjørner D, Eir A. In: Compositionality: Ontology and Mereology of Domains – Some Clarifying Observations in the Context of Software Engineering. Springer Berlin Heidelberg; 2010. p. 22–59.

[5] Noback M. Principles of Package Design: Creating Reusable Software Components. 1st ed. Apress; 2018.

[6] Prehofer C, van Gurp J, Bosch J. Compositionality in software product lines. In: Emerging Methods, Technologies, and Process Management in Software Engineering. John Wiley & Sons; 2008. p. 21–42.

[7] Tripakis S. Compositionality in the Science of System Design. Proceedings of the IEEE. 2016;104(5):960–972.

[8] Turner R, Angius N, Primiero G. The Philosophy of Computer Science. In: Zalta EN, editor. The Stanford Encyclopedia of Philosophy. spring 2019 ed. Stanford University; 2019. .

[9] Keet CM, Artale A. Representing and Reasoning over a Taxonomy of Part-Whole Relations. Applied Ontology. 2008;3(1-2):91–110.

[10] Varzi AC. Mereology. In: Zalta EN, editor. Stanford Encyclopedia of Philosophy. fall 2004 ed. Stanford; 2004. `http://plato.stanford.edu/archives/fall2004/entries/mereology/`.

[11] Winston ME, Chaffin R, Herrmann D. A taxonomy of partwhole relations. Cognitive Science. 1987;11(4):417–444.

[12] Guizzardi G. Ontological Foundations for Structural Conceptual Models [PhD Thesis]. University of Twente, The Netherlands. Telematica Instituut Fundamental Research Series No. 15; 2005.

[13] Rector A, Rogers J, Bittner T. Granularity, scale and collectivity: When size does and does not matter. Journal of Biomedical Informatics. 2006;39:333349.

[14] Copp D. What Collectives Are: Agency, Individualism and Legal Theory. Dialogue. 1984;23:249269.

[15] Artale A, Guarino N, Keet CM. Formalising temporal constraints on part-whole relations. In: Proc. of KR'08. AAAI Press; 2008. p. 673–683. Sydney, Australia, Sept. 16-19, 2008.

[16] Guarino N, Welty C. Identity, Unity, and individuality: towards a formal toolkit for ontological analysis. In: Horn W, editor. Proc. of ECAI'00. IOS Press, Amsterdam; 2000. p. 219–223.

[17] Guarino N, Welty C. An Overview of OntoClean. In: Handbook on Ontologies. Springer Verlag; 2009. p. 201–220.

[18] Keet CM. A Formal Theory of Granularity [PhD Thesis]. KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy; 2008.

[19] Keet CM, Xakaza M, Khumalo L. Verbalising OWL ontologies in isiZulu with Python. In: ESWC'17 Satellite Events. vol. 10577 of LNCS. Springer; 2017. p. 59–64.

[20] Lê DTM, Janicki R. In: Peters JF, Skowron A, editors. A Categorical Approach to Mereology and Its Application to Modelling Software Components. Berlin: Springer Berlin Heidelberg; 2008. p. 146–174.

[21] Mizoguchi R. Functional Ontology of Artifacts. In: Interdisciplinary Ontology Conference 2008. Keio University Press Inc.; 2008. p. 1–10.

[22] Mizoguchi R, Kitamura Y, Borgo S. A unifying definition for artifact and biological functions. Applied Ontology. 2016;11:129–154.

[23] Mizoguchi R, Borgo S. A Preliminary Study of Functional Parts as Roles. In: Proc. of FOUST-II – JOWO'17. vol. 2050 of CEUR-WS; 2017. p. 9p. 21-23 September 2017, Bolzano, Italy.

[24] Vieu L, Aurnague M. Part-of Relations, Functionality and Dependence. In: Categorization of Spatial Entities in Language and Cognition. John Benjamins, Amsterdam; 2005. .

[25] Koslicki K. Substance, Independence, and Unity. In: Feser E, editor. Aristotle on Method and Metaphysics. Palgrave McMillan; 2013. p. 169–195.

V. Methods

This page intentionally left blank

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200675

Model-Finding for Externally Verifying FOL Ontologies: A Study of Spatial Ontologies

Shirly STEPHEN and Torsten HAHMANN [1]
[a] *School of Computing and Information Sciences, University of Maine, USA*

Abstract. Use and reuse of an ontology requires prior ontology verification which encompasses, at least, proving that the ontology is internally consistent and consistent with representative datasets. First-order logic (FOL) model finders are among the only available tools to aid us in this undertaking, but proving consistency of FOL ontologies is theoretically intractable while also rarely succeeding in practice, with FOL model finders scaling even worse than FOL theorem provers. This issue is further exacerbated when verifying FOL ontologies against datasets, which requires constructing models with larger domain sizes.

This paper presents a first systematic study of the general feasibility of SAT-based model finding with FOL ontologies. We use select spatial ontologies and carefully controlled synthetic datasets to identify key measures that determine the size and difficulty of the resulting SAT problems. We experimentally show that these measures are closely correlated with the runtimes of Vampire and Paradox, two state-of-the-art model finders. We propose a definition elimination technique and demonstrate that it can be a highly effective measure for reducing the problem size and improving the runtime and scalability of model finding.

Keywords. ontology verification, first-order logic, satisfiability, model finding, definitions

1. Introduction

Recently, more and more first-order logic (FOL) ontologies have become available, ranging from upper ontologies such as DOLCE or GFO to domain ontologies for space, processes, or the geosciences. But using and reusing them requires extensive prior *evaluation* [1, 2], including, at the very least, verification of their logical consistency [3]. This includes (1) verifying an ontology's *internal consistency* that rules out contradictions between axioms by constructing *any* model, and (2) checking the ontology's *external consistency* with datasets that are representative of the ontology's intended domain or application by constructing a model from these datasets. While ontologies specified in OWL and other Description Logics (DL) are routinely verified both internally and externally even against large datasets (i.e., ABoxes) [4–6], FOL ontologies are currently verified only internally

[1]This material is based in part upon work supported by the National Science Foundation under Grant Numbers III-1565811 and OIA-1937099.

if at all. The reasons are manifold. Most importantly, the very few model-finding experiments with FOL ontologies that have been published in the literature are rather discouraging; model finders routinely time out. They have only been successful where very small models with fewer than 20 individuals exist. For example, Bos states that model finding "doesn't seem to scale up very well" [7] after using Paradox and Mace2 to construct models with up to 17 individuals for rather simple first-order logic theories, while Baumgartner et al. conclude that model finding "is a struggle with more than 20 individuals" [8]. Others [9,10] have also reported little success with FOL model finding though without elaborating further. Despite these experiences, we revisit the feasibility of model finding with FOL ontologies here for several reasons:

- Logical verification of FOL ontologies with and without data is key to the larger endeavor of developing and reliably (re)using FOL ontologies;
- Model-finding with ontologies has much broader utility for other reasoning tasks such as query answering, data cleaning, or identifying which ontologies or ontological assumptions are consistent with a dataset; and
- Prior results are 10 years or older with improvements in model finders and computing resources, especially working memory, since.

We specifically want to identify the key factors that limit model finding with FOL ontologies and potential approaches for improving its scalability.

To construct a model for a FOL ontology, traditional FOL model finders such as Paradox [11] or Vampire [12,13] translate the ontology into an equi-satisfiable Clausal Normal Form (CNF) and then instantiate those clauses for increasing domain sizes. This constructs a series of propositional satisfiability (SAT) problems on which standard SAT solving techniques are used to determine satisfiability (e.g., by constructing a model) or unsatisfiability, in which case the next domain size is tried. While SAT solvers capably handle large SAT problems, SAT-based model finders rarely succeed in constructing models for FOL ontologies. For this reason, we want to better understand what makes model finding with FOL ontologies – with or without data – so difficult in practice. We specifically want to quantify the size of the resulting SAT problems. To do that, we formalize the concept of a FOL ontology with data and define various size measures on its FOL-CNF and SAT conversions relative to the ontology's axioms and dataset. We show that not the number and length of sentences in the ontology's axiomatization, but the size of its signature and, especially, the number of binary predicates and predicates of higher arities (including functions, though we concentrate on predicates) are a critical source of the explosion in the SAT problem size. To address this issue, we introduce a technique – optional definition elimination (ODE) – for reducing the size of the signature. We theoretically quantify and empirically test the effectiveness of this technique on different sets of eliminated definitions using three ontologies. Because model finding results are highly susceptible to seemingly minuscule differences, we carefully control both the numbers of distinct objects (the domain size d) and the number of relational assertions for each predicate via constructing synthetic sample datasets for our experiments. Our results show that ODE can dramatically reduce the size of the resulting SAT problems and significantly speed up and scale up model finding in practice.

While we systematically study different sets of definitions for elimination and different sizes of ABoxes with a total of over 3,000 samples, we were quite restricted in the ontologies we could use for a number of reasons: (1) FOL ontologies typically only contain structural knowledge without datasets of their own (i.e. their ABox is empty); (2) because many FOL ontologies are used as reference ontologies, suitable data sources are not readily available; (3) many FOL ontologies are not even shared in computer-interpretable formats; and (4) FOL ontologies with defined predicates of arity greater 2 and for which any data is available are hard to come by. The chosen spatial ontologies CODI [14], RCC [15] and INCH [16][2] are ideally suited because all of them are available from COLORE at colore.oor.net, they all contain many defined binary predicates (though no predicates of higher arity), and we can easily extract suitable datasets from standard GIS databases using already implemented qualitative spatial operations.

2. Preliminaries

First-Order Logic Syntax and Semantics: A FOL ontology $\mathcal{O}$ is a set of sentences in FOL with equality[3]. Its nonlogical symbols, i.e., all constants, function symbols, and predicates, form its *signature* $\lambda(\mathcal{O})$. For simplicity, we consider here only ontologies with predicates and constants in their signature, because each n-ary function symbol can be encoded as a $n+1$-ary predicate symbol by adding axioms that capture its functional nature[4]. Each predicate symbol $\Omega \in \lambda(\mathcal{O})$ has an arity $a(\Omega) \geq 1$ and constants have arity 0. An *atom* is an expression of the form $\Omega(t_1, ..., t_n)$ where every t_i is a term – typically either a constant or a variable. A *literal* is an atom or its negation $\neg\Omega(t_1, ..., t_n)$. A *FOL formula* is constructed from atoms using the logical connectives $\wedge, \vee, \rightarrow, \leftrightarrow$ and $\neg$ and/or the quantifiers $\forall$ and $\exists$. A *FOL sentence* is a closed formula, that is, all variables are bound. A formula is *ground* if it does not contain any variables, that is, constants are the only terms therein. A FOL *clause* is a special kind of FOL sentence, namely a disjunction of a set of n literals, i.e. $L_1 \vee ... \vee L_n$.

An *interpretation* of an ontology $\mathcal{O}$ is a tuple $\mathcal{I} = \langle D, \Phi, \Psi \rangle$ over a non-empty domain D where Φ maps variables to individuals in D and Ψ maps nonlogical symbols in $\lambda(\mathcal{O})$ to individuals (for constants), sets (for unary predicates) and n-ary relations (for predicates of arity ≥ 2). An interpretation $\mathcal{I}$ under which all sentences in $\mathcal{O}$ are true (i.e., are *satisfied*) is called a *model*. An ontology is *consistent* (or *satisfiable*) if it has some model. Two ontologies are equi-satisfiable if they are both either satisfiable or unsatisfiable.

2.1. SAT-Based Model Finding with FOL Ontologies

The Mace-style finite-model building approach [11, 18, 19] used in popular ATPs (Paradox, Vampire, iProver) uses a two-stage process of, first, clausification and, second, propositional instantiation to convert a FOL ontology into a set of propositional clauses before handing them off to a SAT solver.

[2]The results for INCH are not further discussed here for space reasons, see [17] for details.
[3]We treat equality as a primitive logical predicate.
[4]Because constants typically represent objects from the domain of interest, we include them to allow specifying factual knowledge, i.e., data points.

Clausification: Through applying Skolem's algorithm (see, e.g. [20]) a FOL ontology is converted into an existential quantifier-free *clausal normal form* (CNF) – a set of FOL clauses. This process may introduce additional Skolem constants and functions. Thus, the resulting FOL-CNF ontology, which we refer to as $\mathcal{O}_{\text{FOL-CNF}}$, is not necessarily logically equivalent but still equi-satisfiable to the original ontology $\mathcal{O}$. The size of the *signature* of $\mathcal{O}_{\text{FOL-CNF}}$ is determined as follows:

Definition 1. *Let $\mathcal{O}_{FOL\text{-}CNF}$ be an ontology's FOL-CNF representation. Then*

1. *$sf_{a=n}(\mathcal{O}_{FOL\text{-}CNF})$ denotes the set of n-ary Skolem functions introduced by skolemization. If treated as predicates*[5]*, the set $sf_{a=n}(\mathcal{O}_{FOL\text{-}CNF})$ adds that many $(n+1)$-ary predicates to $\mathcal{O}_{FOL\text{-}CNF}$.*
2. *$\Omega_{a=n}(\mathcal{O}_{FOL\text{-}CNF}) = \{\Omega \in \lambda(\mathcal{O}) \mid a(\Omega) = n)\} \cup sf_{a=n-1}(\mathcal{O}_{FOL\text{-}CNF})$ defines the set of predicates of arity n, which includes the n-ary predicates from $\mathcal{O}$ as well as any newly introduced $(n-1)$-ary Skolem functions.*

The size of $\mathcal{O}_{\text{FOL-CNF}}$ itself is defined in terms of its number of clauses:

Definition 2. *Let $\mathcal{O}_{FOL\text{-}CNF}$ be an ontology's FOL-CNF representation treated as set of clauses. Then for any single clause $C \in \mathcal{O}_{FOL\text{-}CNF}$, the clause-width $w(C)$ is the number of FOL literals therein.*
The formula-width of $\mathcal{O}_{FOL\text{-}CNF}$ is the maximal clause-width of all clauses in $\mathcal{O}_{FOL\text{-}CNF}$, defined as $W(\mathcal{O}) = \max\{w(C) | C \in \mathcal{O}_{FOL\text{-}CNF}\}$.

Propositionalization of the FOL-CNF ontology: This second step involves instantiating all variables within the FOL-CNF clauses over all combinations of individuals from a fixed domain. This requires first fixing the domain size (i.e. the number of distinct individuals) via a set of inequalities [21]. If the domain size is not known in advance, the model finder starts with domain size 1 and incrementally increases it each time the search space is exhausted. If, for example, the smallest model has 8 individuals, then the model finder will prove 7 SAT instances to be unsatisfiable before finding an 8th one that is satisfiable.

Note that propositionalization instantiates every predicate of arity n with d^n propositional variables in $\mathcal{O}_{\text{CNF-d}}$. For example, each binary predicate leads to d^2 and each ternary predicate to d^3 propositional variables.

Lemma 1. *Let $\mathcal{O}_{CNF\text{-}d}$ be the propositional instantiation of $\mathcal{O}_{FOL\text{-}CNF}$, a FOL ontology in CNF form with maximal arity a^*, over a domain with d individuals. Then $\mathcal{O}_{CNF\text{-}d}$ has $P_v = \sum_{i=1}^{a^*} \left(d^i \cdot |\Omega_{a=i}|\right)$ propositional variables.*

Likewise, each clause in an FOL-CNF ontology is instantiated for every combination of its (implicitly universally quantified) variables, leading to the following number of propositional clauses P_c.

Lemma 2. *Let $\mathcal{O}_{FOL\text{-}CNF}$ be a FOL-CNF ontology where C_v denotes the subset of clauses with v distinct FOL variables per clause (we refer to v as the*

[5] We only treat n-ary functions here as (n+1)-ary predicate symbols in order to approximate their influence on the SAT problem size. Model finders typically treat them differently. Because of that, we do not take into account the additional axioms one would need to capture their functional nature when treated as predicates.

variable density), and v^* is the maximal number of variables in any clause in $\mathcal{O}_{FOL\text{-}CNF}$ (the maximal variable density). Then for a domain size d, $\mathcal{O}_{CNF\text{-}d}$ has $P_c = \sum_{i=0}^{v^*} \left(d^i \cdot |C_{v=i}|\right)$ propositional clauses.

Thus, the 'size' of the propositional instantiation $\mathcal{O}_{\text{CNF-d}}$ can be jointly described using P_c and P_v; their ratio $r = \frac{P_c}{P_v}$ describes its clause density. While our approach for calculating P_v and P_c is rather naive and only a worst-case measure, preprocessing techniques built into modern model finders are meant to reduce these numbers. Nevertheless, we will show in Section 5 that these calculated measures are closely correlated to the experimental runtimes of model finders.

3. SAT-based Model Finding for FOL Ontologies with Data

For simply proving the internal consistency of a FOL ontology, no data (i.e. *ground facts*) are needed. However, to prove that an ontology is consistent with a given dataset, we need to take the dataset's size into account when estimating the size of the resulting SAT problem. To investigate how the size of $\mathcal{O}_{\text{CNF-d}}$ changes with different amounts of data, we adapt the notions of Terminological Box (TBox), Relations Box (RBox), and Assertion Box (ABox) from Description Logic (DL) ontologies [22, 23]. The TBox and RBox capture axioms that constrain the interpretations of concepts (i.e., unary predicates) and roles (i.e., binary predicates), respectively. We will not distinguish between them, but draw the distinction between the TBox (for all terminological axioms) and the ABox, the latter of which captures assertions about individuals, i.e., ground statements about an individual being an instance of a particular concept or being related to another individual via a particular relation.

3.1. Assertion Box (ABox) and Terminology Box (TBox)

A FOL ontology can mix structural knowledge and assertions about individuals, even in a single sentence. Because clausification tends to separate those to some degree, we define an ontology's ABox in terms of its FOL-CNF version.

Definition 3. *Let $\mathcal{O}$ be an ontology with signature $\lambda(\mathcal{O})$ and let $\mathcal{O}_{FOL\text{-}CNF}$ be its corresponding set of FOL-CNF clauses. Then the FOL assertion box* ABox($\mathcal{O}$) *is the subset of $\mathcal{O}$'s sentences that only yield ground clauses in $\mathcal{O}_{FOL\text{-}CNF}$ that do not use symbols outside $\lambda(\mathcal{O})$*[6]*.*

While an ABox may contain disjunctive knowledge – reflected in ground clauses with multiple literals – many clauses are so-called *unit clauses* consisting of only a single literal, which are *facts*. In our experiments, we limit the ABox to such unit clauses. For simplicity, we further require that the ABox itself, and not just its clausal conversion, is represented as a set of ground clauses. In other words, *the ABox is the dataset* we want to verify an ontology against.

Definition 4. *An* ABox($\mathcal{O}$) *is called factual iff it contains only unit clauses.*

[6]Clauses that are ground but use newly introduced Skolem constants or functions are not considered part of the ABox as the Skolem symbols arise from existential quantifiers.

The spatial ontologies (CODI, RCC, and INCH) we use in our experiments rely – like many other ontologies – only on unary and binary predicates. If the ABox for such an ontology is factual, it consists of two kinds of assertions: *class assertions* (e.g., *ArealRegion*(‘*penobscotCounty*’)) and *relational assertions* (e.g., *Inc*(‘*i95*’,‘*penobscotCounty*’)). A special kind of assertions we (and many model finders) add, are so-called *distinctness assertions* that ensure that distinct constants denote distinct individuals (e.g., ‘*i95*’≠‘*penobscotCounty*’).

A FOL ontology's TBox captures its structural, i.e., non-factual knowledge. It consists of all sentences that either yield non-ground clauses or that contain Skolem symbols after conversion to CNF-FOL:

Definition 5. *Let $\mathcal{O}$ be a FOL ontology and* ABox($\mathcal{O}$) *its ABox. Then its FOL terminology box is defined as* TBox($\mathcal{O}$) $= \mathcal{O} \setminus$ ABox($\mathcal{O}$).

For an ontology with a factual ABox, the TBox will not contain any ground clauses except possibly ones involving Skolem symbols.

3.2. The Size of the Resulting SAT Problem

The following example demonstrates the clausification and propositionalization of an FOL ontology and the number of propositional variables and propositional clauses that are created in the process.

Example 1. *Consider $\mathcal{O}_{RCC\text{-}s}$ as a small subset of RCC's FOL axiomatization that consists of one axiom and two definitions with signature $\lambda(\mathcal{O}) = \{C, P, PP\}$ denoting contact $C(x,y)$, parthood $P(x,y)$, and proper parthood $PP(x,y)$.*

$$(\sigma_C)\ C(x,y) \rightarrow C(y,x)$$
$$(\sigma_P)\ P(x,y) \leftrightarrow \forall z[C(z,x) \rightarrow C(z,y)]$$
$$(\sigma_{PP})\ PP(x,y) \leftrightarrow P(x,y) \wedge \neg P(y,x)$$

Clause 1	$\neg C(x,y) \vee C(y,x)$.	**Clause 2**	$PP(x,y) \vee \neg P(x,y) \vee P(y,x)$.
Clause 3	$P(x,y) \vee C(f(x,y),x)$.	**Clause 4**	$P(x,y) \vee \neg C(f(x,y),y)$.
Clause 5	$\neg PP(x,y) \vee P(x,y)$.	**Clause 6**	$\neg PP(x,y) \vee \neg P(y,x)$.
Clause 7	$\neg P(x,y) \vee \neg C(z,x) \vee C(z,y)$.		

Table 1. FOL-CNF clauses for TBox($\mathcal{O}_{RCC\text{-}s}$). Clauses are joined by conjunctions.

The FOL-CNF version of $\mathcal{O}_{RCC\text{-}s}$ (Table 1) contains 7 *clauses with* 4 *nonlogical symbols, which in addition to the 3 predicates from $\mathcal{O}_{RCC\text{-}s}$ includes one binary Skolem function f which can be encoded as a ternary predicate. Propositionalization for domain size* $d = 20$ *yields the following number of propositional variables:*

$$P_v = |\Omega_{a=2}| \cdot d^2 + |\Omega_{a=3}| \cdot d^3 = 3 \cdot 20^2 + 1 \cdot 20^3 = 9,200$$

Out of the 7 clauses, one clause has 3 FOL variables (clause 7) while the other six all have 2 FOL variables (FOL variables in different clauses are different for the purpose of propositionalization). For an ABox($\mathcal{O}$) *with exactly one relational assertion, namely PP(‘m’,‘n’), exactly one ground clause is added. Then for domain size* 20 *the following number of propositional clauses are created:*

$$P_c = |C_{(v=3)}| * d^3 + |C_{(v=2)}| * d^2 + |C_{(v=1)}| * d^1 + |C_{(v=0)}| * d^0$$
$$= 1 \cdot 20^3 + 6 \cdot 20^2 + 0 \cdot 20^1 + 1 \cdot 20^0 = 10,401$$

Thus the number of propositional variables in the SAT representation is largely dependent upon the number and arity of predicates – each predicate of arity a results in d^a propositional variables for domain size d. This number determines the search space for the propositional SAT problem, which consists (without using any heuristics) of 2^{P_v} possible interpretations. A simple ontology with b binary and u unary predicates (and no other non-logical symbols) then yields $(2^b)^{d^2} \cdot (2^u)^d$ interpretations, which is exponential in both the number of binary predicates – and more generally the number of predicates of highest arity – and the domain size d. While modern SAT solvers are able to deal with thousands of variables and tens of thousands of clauses [24] via highly effective strategies for pruning the search space, P_v and P_c quickly grow into the millions even for ontologies with modestly-sized signatures with only a handful of binary predicates. But this also suggests that improvements can be realized by reducing the total number of predicates, especially those of highest arity to construct larger and more realistic models. Definition elimination, as formalized in Section 4, can accomplish this when many predicates are defined. But we first look more closely at how the ABox impacts the size of the resulting SAT problem.

3.3. The Impact of the Ontology's ABox on the Size of the SAT Problem

The composition of ABoxes can vary widely: it may contain a handful or thousands of facts, and some predicates may be used much more than others. In the extreme case, many predicates may only rarely or not at all be used in an ABox. To study the impact of the ABox in a more systematic way, we need to carefully control its size and makeup. To do that, we introduce the idea of an *(r-d)ABox* where d is the domain size (i.e., the number of distinct individuals in the domain) and r is a factor that controls how many relational assertions the ABox contains for each of the ontology's predicates of arity greater than 1.

Definition 6. *Let $\mathcal{O}$ be an ontology and D a domain of individuals. ABox($\mathcal{O}$) is called an (r-d)ABox iff it only contains the following assertions:*

1. *For each $\Omega \in \lambda(\mathcal{O})$ with arity $a(\Omega) \geq 2$, ABox($\mathcal{O}$) contains exactly r ground positive assertions (i.e. of the form $\Omega(d_1, d_2, \dots)$) and exactly r ground negated assertions (i.e. of the form $\neg\Omega(d'_1, d'_2, \dots)$ where $d_i, d'_i \in D$;*
2. *ABox($\mathcal{O}$) contains at most one sentence of the format $\Omega(d)$ for each $d \in D$ where Ω is a unary predicate (i.e. $\Omega \in \lambda(\mathcal{O})$ with $a(\Omega) = 1$)[7];*
3. *Distinctness assertions of the form $d_i \neq d_j \in$ABox($\mathcal{O}$) for each pair $(d_i, d_j) \in D \times D$ with $d_i \neq d_j$.*

Note that these ABoxes are stratified in the sense that all non-unary predicates are used equally many times. While this may rarely happen in practical datasets[8], it allows us to avoid introducing noise caused by unrelated differences between ABoxes. Building on Lemmas 1 and 2, the size of the SAT problem resulting from an ontology with an (r-d)ABox can be calculated as:

[7]This criteria captures the idea that each individual in the domain can be asserted to be a member of some class; but this restriction has a limited impact on the overall size of the resulting SAT problem, which will be dominated by the number of relational assertions.

[8]To estimate the size of SAT problems resulting from practical datasets, we could treat r as an maximal number of assertions that use one predicate. But as it turns out, r mostly influences the number of propositional clauses but rarely the number of propositional variables.

Lemma 3. *Let $\mathcal{O}$ be a FOL ontology with ABox($\mathcal{O}$) being an $(r\text{-}d)$ABox thereof. Then the resulting propositional SAT problem contains*

- $P_v = \sum_{i=1}^{a^*} d^i \cdot |\Omega_{a=i}| + r \cdot \sum_{i=1}^{a^*} d^i \cdot |sf_{A,a=i}|$ *propositional variables; and*
- $P_c = \sum_{i=0}^{v^*} d^i \cdot |C_{T,v=i}| + r \cdot \sum_{i=0}^{v^*} d^i \cdot |C_{A,v=i}|)$ *propositional clauses.*

where a^* denotes the maximum arity of predicates (including those resulting from Skolem functions) and v^* the maximal variable density in $\mathcal{O}_{FOL\text{-}CNF}$.

P_v and P_c are driven by the maximal arity of predicates (a^*) and the maximal variable density (v^*), respectively, while the ABox's contribution (the second term in each equation) is rather small for factual ABoxes without definition elimination: P_v will not change at all because ground unit clauses yield no Skolem functions, while P_c increases by the number of facts contained in the ABox. Even for an ABox with thousands of facts, this is negligible compared to the number of clauses generated from the TBox for growing domain sizes.

4. Definition Elimination

We will now formalize optional definition elimination as a technique that reduces an FOL ontology's signature size. The following example shows how it can potentially reduce the size of the resulting SAT problems.

Example 2. *We reuse the* TBox *from Example 1 with $\lambda = \{C, P, PP\}$. One binary Skolem function (analogous to a ternary predicate) is introduced by clausification. Propositionalization yields* $68,800$ *and* $531,200$ *propositional variables for domain sizes* 40 *and* 80*, respectively. Adding one binary predicate O (overlap), which is explicitly defined by* (σ_O) $O(x,y) \leftrightarrow \exists z[P(z,x) \wedge P(z,y)]$*, almost doubles the number of propositional variables to* $134,400$ *and* $1,049,600$ *for $d = 40$ and 80, respectively. This increase is largely attributable to an additional ternary predicate that captures the introduction of a binary Skolem function as the result of eliminating the existential quantifier in σ_O. While the number of FOL-CNF clauses increases from 7 to 10, one of the new clauses has a variable density of 3. This leads to an eight-fold increase in the number of propositional clauses from* $73,600$ *to* $640,000$ *(for $d = 40$) and from* $550,400$ *to* $5,120,000$ *(for $d = 80$).*

In this example the addition of just one binary predicate causes large increases in P_v and P_c. But because the added predicate O is explicitly defined, its removal does not change the ontology's satisfiability and O's interpretation can be easily constructed after finding a model. To formalize this approach, we first define optional definitions and the DBox as a maximal set of optional definitions that can be easily removed from an ontology. They are based on the notion of explicit definitions [25] as special types of TBox sentences:

Definition 7. *An explicit definition of an n-ary predicate $\Omega \in \lambda(\mathcal{O})$ in an ontology $\mathcal{O}$ is a sentence $\sigma \in$ TBox(O) of the form*

$$\forall x_1, \ldots, x_n[\Omega(x_1, \ldots, x_n) \leftrightarrow \alpha(x_1, \ldots, x_n)]$$

wherein α is a formula with x_1 to x_n as only free variables and with $\lambda(\mathcal{O}) \setminus \Omega$ as the only nonlogical symbols. Then Ω is said to be explicitly defined in T.

Optional definitions are explicit definitions of predicates that are not used in other sentences of the ontology's TBox:

Definition 8. *An explicit definition* $\sigma \in \mathrm{TBox}(\mathcal{O})$ *of a symbol* $\Omega \in \lambda(\mathcal{O})$ *is an* <u>*optional definition*</u> *in* $\mathcal{O}$ *iff* Ω *does not appear in any sentence in* $\mathrm{TBox}(\mathcal{O}) \setminus \sigma$.

Note that after removing an optional definition from $\mathrm{TBox}(\mathcal{O})$, other previously non-optional explicit definitions can also become optional in $\mathrm{TBox}(\mathcal{O}) \setminus \sigma$.

Example 3. *In* $\mathcal{O}_{RCC\text{-}s}$*, initially* σ_{PP} *is an optional definition but* σ_P *is not because it is used in the definiens of* σ_{PP}*. After removing* σ_{PP}*,* σ_P *becomes optional and can also be removed.*

Because of this effect we recursively define larger *definition sets*, with the maximal one being referred to as the ontology's DBox:

Definition 9. *A* <u>*definition set*</u> *of an ontology* $\mathcal{O}$ *is defined recursively as:*

B. *The set of all optional definitions in* $\mathrm{TBox}(\mathcal{O})$ *forms a definition set;*
R. *For any definition set* D *of* $\mathcal{O}$ *and for any optional definition* σ *of* Ω *in* $\mathrm{TBox}(\mathcal{O}) \setminus D$*, the set* D' *defined as follows is a definition set:*
$D' = \{\varsigma' | \varsigma \in D \text{ and } \varsigma' = \varsigma[\Omega(x_1, \ldots, x_n)/\alpha(x_1, \ldots, x_n)]\} \cup \{\sigma\}$,
that is, D' *substitutes all occurrences of* Ω *in other definitions in* D *by* Ω*'s definiens from* σ *and adds* σ *as a new definition to the set.*

Definition 10. *For an ontology* $\mathcal{O}$*,* <u>$\mathrm{DBox}(\mathcal{O})$</u> *is a definition set such that no optional definition exists in* $\mathrm{TBox}(\mathcal{O}) \setminus \mathrm{DBox}(\mathcal{O})$.
$\Omega \in \lambda(T)$ *is* <u>*optionally defined*</u> *in* $\mathcal{O}$ *iff* Ω *does not appear in* $\mathrm{TBox}(\mathcal{O}) \setminus \mathrm{DBox}(\mathcal{O})$.

To study how removing optionally defined predicates impacts the SAT representation, we also need to substitute the eliminated predicates in the ABox without changing the ontology's semantics. This is achieved by replacing assertions that use optionally defined predicates by *defined assertions.*

Definition 11. *Let* $\mathcal{O}$ *be an ontology and* D *some definition set of* $\mathcal{O}$.
Then <u>$\mathbf{ABox_D}(\mathcal{O})$</u> $= \mathrm{ABox}(\mathcal{O})\big[\bigcup_{\sigma_i \in D}[\Omega_i(x_1, \ldots, x_n)/\alpha_i(x_1, \ldots, x_n)]\big]$.
Any sentence $\sigma \in \mathrm{ABox}_D(\mathcal{O})$ *with* $\sigma \notin \mathrm{ABox}(\mathcal{O})$ *is called a* <u>*defined assertion*</u>.

In other words, $\mathbf{ABox_D}(\mathcal{O})$ is $\mathcal{O}$'s ABox with all optionally defined predicates Ω_i from D (which typically would be the entire DBox of $\mathcal{O}$) substituted by their definiens α_i. Such an ABox with defined assertions may no longer only contain only ground unit clauses. Defined assertions may contain variables or Skolem functions introduced during the substitution. For example, a fact O('i95', '295w') results in the defined assertion $\exists z[P(z, \text{'i95'}) \wedge P(z, \text{'295w'})]$ if O is substituted by its definiens from Example 2. But the substitution of facts from the ABox by defined assertions ensures maintaining (non-)satisfiability. This follows directly from the well-known relationship between explicit and implicit definability formalized by Beth's definability theorem [25]. The following theorem captures this; the straightforward proof is provided in [17, Sec. 5.3]:

Theorem 1. *Let* $\mathcal{O}$ *be a FOL ontology and* D *be a definition set of* $\mathcal{O}$*. Then there is a bijection between the models of* $\big(\mathrm{TBox}(\mathcal{O}) \setminus D\big) \cup \mathrm{ABox}_D(\mathcal{O})$ *and the models of* $\mathrm{TBox}(\mathcal{O}) \cup \mathrm{ABox}(\mathcal{O})$*, that is, every model of* $\big(\mathrm{TBox}(\mathcal{O}) \setminus D\big) \cup \mathrm{ABox}_D(\mathcal{O})$ *can be uniquely expanded into a model of* $\mathrm{TBox}(\mathcal{O}) \cup \mathrm{ABox}(\mathcal{O})$.

Again, such a bijection exists specifically where $D = \text{DBox}(\mathcal{O})$, the maximal set of optional definitions that can be easily removed without altering the ontology's semantics. This idea forms the basis of our strategy for improving model finding because $\big(\text{TBox}(\mathcal{O}) \setminus \text{DBox}(\mathcal{O})\big) \cup \text{ABox}_D(\mathcal{O})$ has a smaller signature than $\mathcal{O} \equiv \text{TBox}(\mathcal{O}) \cup \text{ABox}(\mathcal{O})$ but is equi-satisfiable.

The effect of ODE on the resulting SAT problem's size is illustrated next.

Example 4. *Consider the ontology $\mathcal{O}_{RCC\text{-}s}$ again with $\beta \equiv PP(\text{'m'}, \text{'n'})$ as the only assertion in the ABox. Now applying ODE only on PP removes σ_{PP} from the TBox and substitutes all occurrences of PP in the ABox with its definiens $P(x,y) \wedge \neg P(y,x)$. β will become $\beta' \equiv P(\text{'m'}, \text{'n'}) \wedge \neg P(\text{'n'}, \text{'m'})$. ODE reduces the ontology's signature from three to two binary predicates. For a domain size* 20*, this reduces the number of propositional variables from* $3 * 20^2 + 1 * 20^3 = 9,200$ *to* $2 * 20^2 + 1 * 20^3 = 8,800$*. Likewise, the number of propositional clauses is slightly reduced from* 10,400 *to* 10,000*. A much larger decrease can be realized by eliminating syntactically more complex definitions, such as the definition σ_P of P, which contains an existential quantifier. σ_P is optionally defined after σ_{PP}'s removal. Its removal eliminates the ternary predicate, leading to a SAT problem with only* $2 * 20^2 = 800$ *propositional variables that arise from its TBox*[9].

It becomes clear that optional definition elimination is especially useful for ontologies that have many optionally defined predicates. This is the case for many spatial ontologies, such as the RCC or CODI, which we use next to study the benefit of ODE in more detail with respect to the estimated number of propositional variables and clauses and the actual change in runtimes of model finders.

5. Calculated and Experimental Results

Materials and Experimental Setup: We now present our quantitative and experimental results for two ontologies of qualitative space: CODI [14] and RCC [15]. For space reasons we do not discuss results from a third ontology, the INCH Calculus[10], which are included in [17]. Complete FOL axiomatization of CODI[11] and RCC[12] are available in the COLORE github repository. CODI and RCC each have a terminology of a total of 21/6 mostly binary predicates, with 8/5 of those being explicitly defined. The main reason for working with these ontologies is that their existing semantic alignment with the Simple Features Access (SFA) Model [26] – an FOL ontology that bridges qualitative spatial relations with standard geometric representations as used in GIS databases – allows easy extraction of datasets from standard GIS datasets.

For constructing (r-d)ABoxes (cf. Def. 6) we first extracted a *Master ABox* consisting of 425 spatial objects (i.e. individuals) related via 130,256 ground relational assertions (4,937 positive ones and 125,319 negated ones) that use the binary predicates (within, overlaps, intersects, crosses and touches) from the SFA standard. These map to the binary predicates used by CODI and RCC. To con-

[9]The overall number of propositional variables would be larger if the ABox heavily uses the eliminated predicate, as that would reintroduce some variables via Skolemization.

[10]http://colore.oor.net/inch/inch_calculus.clif

[11]http://colore.oor.net/multidim_space_codi/codi_basic.clif

[12]http://colore.oor.net/mereotopology/rcc_basic.clif

struct sample datasets from this Master ABox, we used a stratified sampling process to semi-randomly pick 5, 10, 15, or 20 relational assertions for each binary relation in the ontology's signature that relate between 20 to 50 individuals (to fix the domain size). We further added distinctness assertions (inequalities) for these individuals and, for CODI, class assertions for each unary predicate. This systematic construction of ABoxes helps attribute performance differences to the removal of specific predicates as all predicates are used with equal frequency. It avoids biasing or even hiding the effect of definition elimination that could result from real datasets, of which one expects a large variations in predicate use: some may be used much more than others while others potentially not at all.

We conducted our experiments with the state-of-the-art model finders Paradox [11] and Vampire [27][13] using 10 dataset samples for each ontology, signature, and combination of parameters, resulting in a total of 2,080 problems for CODI and 840 problems for RCC. To discount external effects and extreme outliers (since problems are created semi-randomly, some are extremely difficult), we only plot the *low mean* – the mean of all 10 samples that terminated before the mean plus one standard deviation ($\mu + \sigma$). Cases where the majority of samples did not terminate are indicated as such in our graphs.

5.1. Growth of the Number of Propositional Variables and Clauses

The dashed lines in Fig. 1 show how P_v differs across different sets of eliminated predicates. P_v increases polynomially with increasing d (cf. Lemma 3) while r has no impact for CODI and little impact on RCC except for case 1 there. RCC case 1 is special as it is the only case that introduces clauses with a variable density of 3 as shown in the RCC's ABox measures in Table 2. P_v decreases significantly when more and more defined predicates are eliminated. For example, the elimination of the 5 binary predicates *SC*, *Inc*, *PO*, *PP* and *C* from CODI (from case 13 to case 1) reduces P_v by roughly two-thirds even though 9 out of 14 binary predicates are preserved.

The number of propositional clauses P_c is exponential in the maximal variable density over all FOL-CNF clauses, polynomial in d, and increases linearly (though minimally) with increasing r. Overall, P_c grows in step with P_v. However, overeager ODE can lead to (1) more FOL-CNF clauses, (2) much longer clauses (i.e. more clauses with a width ≥ 3), and (3) clauses with higher variable density. The latter two things happen when eliminating all defined predicates, including P, from RCC as shown by the ABox calculations for case 1 in Table 2, which results in the large increases in P_c and P_v shown in Fig. 1. This is caused by the nesting of optional definitions in RCC: *EC* is defined in terms of O and both *PP* and O are defined in terms of P. The construction of defined assertion then leads to multiple nested quantifiers, resulting in a high variable density after clausification.

5.2. Experimental Results

While Vampire is consistently faster than Paradox, the model finding times of both exhibit a similar pattern (especially for CODI) that is also closely correlated to the number of propositional variables. As more defined predicates are

[13]We also experimented with IProver but the runtimes are very inconsistent but for larger domain sizes always significantly greater than those for Vampire.

CODI		TBox					Basic ABox ($r = 1$)				
CODI Case	**defined predicates included**	$\Omega_{a=1}$, $\Omega_{a=2}$	$C_{T,v}$ with v FOL variables			C_T with $w \geq 3$	$C_{A,v}$ with v FOL variables			C_A with $w \geq 3$	$\Omega_{a=1}$
			v=3	v=2	v=1		v=2	v=1	v=0		
1	- (all cases include 22 other predicates)	13, 9	3	32	30	31	12	12	3	7	6
2	PP	13, 10	3	35	30	33	12	12	2	7	6
3	C	13, 11	4	34	30	33	12	9	4	7	5
4	C, PP	13, 12	4	37	30	35	12	9	4	7	5
5	PO	13, 11	4	34	30	33	12	9	5	7	5
6	PO, PP	13, 12	4	37	30	33	12	9	4	7	5
7	PO, PP, C	13, 14	5	39	30	35	12	6	6	7	4
8	Inc	13, 14	5	41	30	43	3	10	5	5	4
9	Inc, PP	13, 13	5	44	30	43	3	10	4	5	4
10	Inc, PP, C, PO	14, 16	7	48	30	45	3	4	8	5	2
11	SC	13, 12	8	34	30	36	9	8	5	2	4
12	SC, PP	13, 13	8	37	30	38	9	8	4	2	4
13	SC, PP, C, PO, Inc	13, 20	12	50	30	51	0	0	10	0	0

RCC		TBox					Basic ABox ($r = 1$)					
RCC Case	**defined predicates included**	$\Omega_{a=2}$	$C_{T,v}$ with v FOL variables			C_T with $w \geq 3$	$C_{A,v}$ with v FOL variables				C_A with $w \geq 3$	$\Omega_{a=1}, \Omega_{a=2}$
			v=3	v=2	v=1		v=3	v=2	v=1	v=0		
1	- (all cases include C)	1	0	1	1	0	4	16	20	5	32	7, 10
2	P	3	1	4	1	2	0	0	8	12	8	3, 2
3	P, PP	4	1	7	1	3	0	0	8	10	6	2, 0
4	P, O	5	2	6	1	3	0	0	1	14	5	0, 0
5	P, PP, O	6	2	9	1	4	0	0	1	12	1	0, 0
6	P, PP,O, EC	7	2	12	1	5	0	0	1	9	0	0, 0
7	P, PP, O, EC, NTTP	8	3	11	1	8	0	0	0	8	0	0, 0

Table 2. Each case represents version of CODI/RCC with a different set of its explicitly defined binary predicates included. Statistics for the resulting size of the FOL-CNF ontology for the TBox only (left) and an ABox with $r = 1$ (i.e. that includes exactly one positive and one negated assertion for each binary predicate, including the non-optional predicates).

eliminated and P_v decreases, the runtimes decrease as well[14]. The speed-up can be significant. For example for $d = 20$, case 1 contains less than half the number of propositional variables (3,860) than case 13 (8,260), which contains five additional binary predicates. This leads to a runtime deduction from 345s to 7s and from 713s to 134s for $r = 5$ and $r = 20$, respectively. This reduction is even more dramatic for $d = 30$, where the runtimes decrease from 30,000s (over 8h) to 16s and 164s for $r = 5$ and $r = 20$, respectively.

While there are slight differences about how well certain cases perform (e.g., cases 11 and 12 are more difficult for Paradox, whereas cases 8 to 10 are more difficult for Vampire), invariably CODI case 13 (without ODE) takes the longest for both solvers, with timeouts encountered consistently for $d = 30$ (Paradox) or $d = 50$ (Vampire). Case 1, which performs the most aggressive ODE, terminates fastest for Paradox throughout. However, the Vampire results show that removing as many definitions as possible does not always yield the best performance, in fact, case 2 which retains the definition of *PP* performs better. Overall, CODI demonstrates that ODE can improve model-finding scalability noticeably: In the best cases the model finders can construct models with up to 50 individuals in the same times previously needed for half that domain size.

By looking at the RCC results, an even more nuanced story emerges. While the runtimes mostly follow the trend of P_v, the steep increase in P_c and P_v in

[14]Beware that the cases are not ordered by P_v, and for CODI not even by the number of binary predicates preserved in the FOL-CNF ontology as shown column 3 of Table 2.

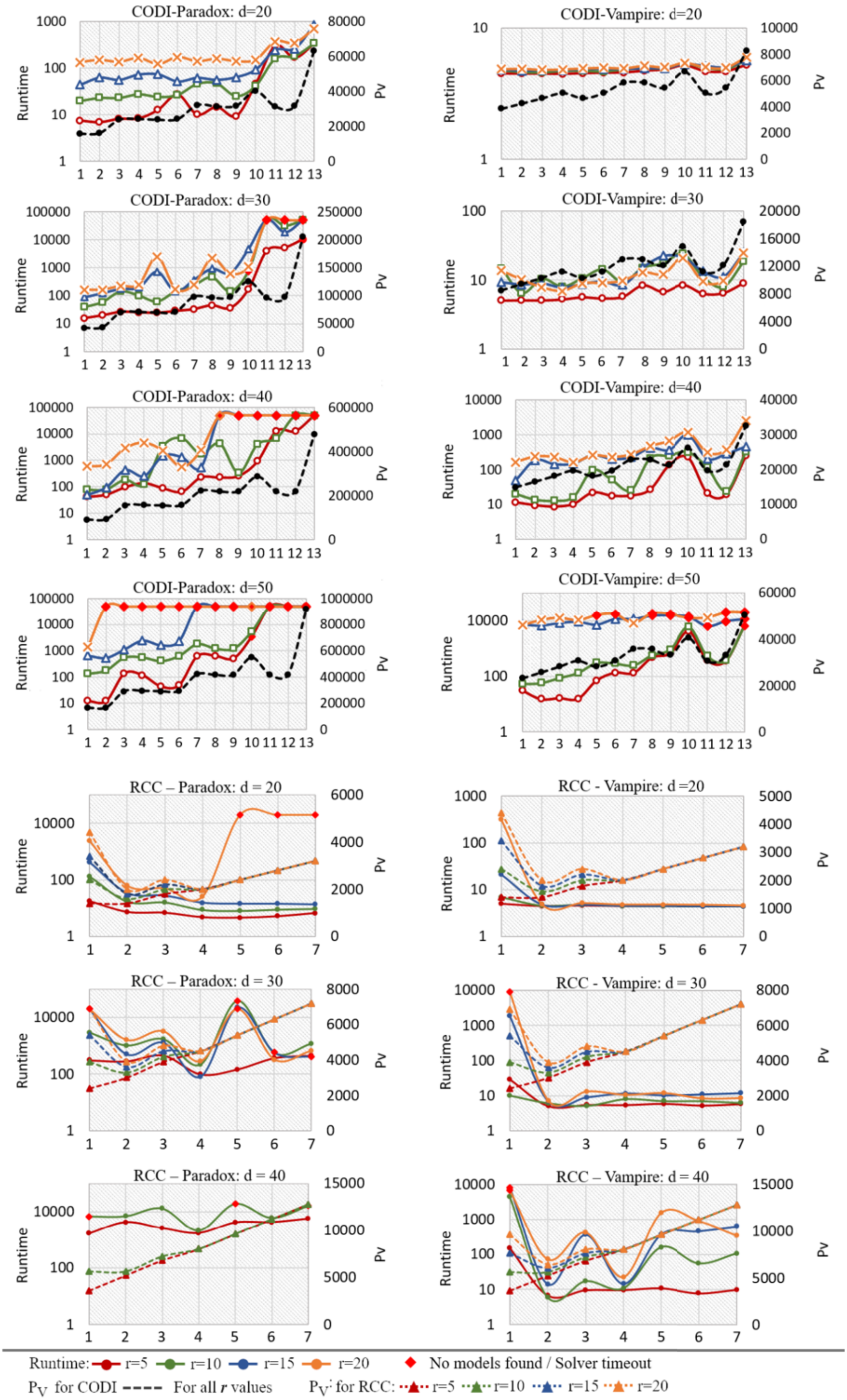


Figure 1. Low mean model finding times for CODI and RCC for different r-dABoxes using Paradox and Vampire. The numbered cases along the x-axis correspond to the case numbers in Table 2. The calculated number P_v (dashed) is plotted against the secondary axis.

case 1 yields a performance worse than without any ODE at all. P_v and P_c are the lowest in case 2, which removes all optionally defined predicates except for P. This case keeps both the number of newly introduced Skolem functions and the number of clauses with higher variable density relatively low. This is the best case for Vampire for both domain sizes 20 and 30. Paradox performs slightly better on case 4, which additionally retains O and results in even fewer clauses of highest variable density ($C_{A,1}$ is 1 compared to 8 as in cases 2 and 3).

In summary, we find that the quantitative measures of the FOL-CNF ontologies are good indicators for the expected runtimes of the employed SAT-based model finders. The results suggest that ODE is beneficial as long as the number of clauses with high variable density (i.e. with more variables) increases only marginally. Because such increases arise primarily from the introduction of quantifiers and Skolem functions in defined assertions, the optimal extent of ODE could be easily automatically determined during the clausification process.

6. Related Work

The elimination of either implicitly [28] or explicitly defined concepts [29] has been studied for expressive DL ontologies, but not for FOL ontologies. Verification of FOL ontologies has mostly focused on theorem proving [3, 30] as a means to find inconsistencies or to prove competency questions, while model finding has been dismissed as infeasible. Prior studies of automated reasoning over FOL ontologies, such as [9, 31, 32] have focused on the feasibility of theorem proving (as a means for query answering) rather than model finding. Those studies have shown that theorem proving scales poorly for large ontologies, but they also noted that the challenges for model finding are even greater. For example, in the study in [10] Paradox generated only models up to size 5 and Darwin timed out for most cases, while [9] also reports to have constructed only very small models. Likewise, no models for domain sizes larger than 17 were found by [7] using Paradox and Mace2 even for simple FOL theories. Their limited success let the authors in [8] conclude that model finding is "a struggle with more than 20 individuals". Our study pushes this limit to construct models with up to 120 individuals (for CODI) through the simple techniques of eliminating optional definitions.

While heuristics for improving SAT solving via various preprocessing techniques abound, most are only employed after clausification and propositionalization of an FOL axiomatization. Only few techniques (see [19] for a recent overview) have been successfully lifted to FOL model finders. For example, syntactic predicate elimination is performed on the CNF version of an ontology by Mace4 [18], but in our experiments, Mace4 did not even get close to the performance of Paradox or Vampire. Likewise, iProver eliminates some non-self-referential predicates [33] but behaved much more unpredictably than Paradox or Vampire and did not scale as well as Vampire.

7. Conclusion and Future Work

This work is only a first step in systematically studying the feasibility of model finding for FOL ontologies and in identifying avenues to improve FOL model finding in practice. We have shown that the size of an ontology's signature, especially the number of predicates of arity two or higher, has an outsized impact on the size of models that can be created, while the size of the dataset is much less critical. We have also shown that optional definition elimination (ODE) can effectively push the size of models that can be constructed. For example, ODE reduces the model finding time by over an order of magnitude for CODI, which in turn allowed us to construct models containing up to 120 individuals in the same amount of time previously needed to construct models with about 40 individual. This is a small but noticeable improvement over previous reports that indicated difficulties constructing any models with 20 or more individuals [7, 8]. Thus, our results are evidence that external verification of mid-sized FOL ontologies can be feasible in practice, at least for ontologies with only unary and binary predicates and wherein many terms are defined. However, as perhaps expected, it is also clear that model finding for FOL ontologies will likely never compete with the size of datasets that DL ontologies can be externally verified against.

Future Work It has become clear that ODE cannot be applied blindly, especially when dealing with ground facts that use the predicates slated for elimination. The example of RCC showed that eliminating the wrong predicate can significantly decrease model finding performance. However, simple measures, such as the number of quantified variables, the number and arity of introduced Skolem functions, and the number of ABox facts that use a specific predicate can help decide which predicates to eliminate. Future work needs to develop and study specific heuristics that employ such measures for automatically deciding and applying ODE as a preprocessing step for FOL model finding. More work is also necessary to study the effectiveness of ODE for predicates of higher arities and to test ODE on a more diverse set of FOL ontologies, including much larger ontologies.

Acknowledgments We thank the four anonymous reviewers for their sage feedback which greatly contributed to an improved final version of this paper.

References

[1] Vrandečić D. Ontology Evaluation. In: Handbook on Ontologies. 2nd ed. Springer; 2009. p. 293–313.

[2] Gómez-Pérez A. Ontology Evaluation. In: Handbook on Ontologies. 1st ed. Springer; 2004. p. 251–274.

[3] Grüninger M, Hahmann T, Hashemi A, Ong D. Ontology verification with repositories. In: Conf. on Formal Ontology in Inf. Systems (FOIS-10). IOS Press; 2010. p. 317–330.

[4] Xu J. Improving ABox reasoning for efficient and scalable object queries over large ontologies [PhD Thesis]. University of Miami; 2015.

[5] Dentler K, Cornet R, Ten Teije A, De Keizer N. Comparison of reasoners for large ontologies in the OWL2 EL profile. Semantic Web. 2011;2(2):71–87.

[6] Srinivas K. OWL Reasoning in the Real World: Searching for Godot. In: Description Logic Workshop (DL 2009), Jul 27-30, Oxford, UK. CEUR Vol. 477; 2009. p. 2:1–10.

[7] Bos J. Exploring model building for natural language understanding. In: Workshop on Inference in Computational Semantics, Sep 25-26, Nancy, France; 2003. p. 41–55.

[8] Baumgartner P, Fuchs A, De Nivelle H, Tinelli C. Computing finite models by reduction to function-free clause logic. Journal of Applied Logic. 2009;7(1):58–74.

[9] Pease A, Sutcliffe G. First Order Reasoning on a Large Ontology. In: CADE-21 ESARLT Workshop, Jul 17, Bremen, Germany. CEUR Vol. 257; 2007. p. 2:1–24.

[10] Schneider M, Sutcliffe G. Reasoning in the OWL2 Full ontology language using first-order automated theorem proving. In: Conf. on Automated Deduction (CADE-23), Jul 31-Aug 5, Wroclaw, Poland. Springer; 2011. p. 461–475.

[11] Claessen K, Sörensson N. New techniques that improve MACE-style finite model building. In: CADE-19 Workshop on Model Computation, Miami Beach, FL; 2003. p. 11–27.
[12] Kovács L, Voronkov A. First-order theorem proving and Vampire. In: Conf. on Computer Aided Verification (CAV 2013), Jul 13-19, St. Petersburg, Russia. Springer; 2013. p. 1–35.
[13] Biere A, Dragan I, Kovács L, Voronkov A. Experimenting with SAT solvers in Vampire. In: Mexican Intern. Conf. on Artif. Intell. (MICAI'14), Nov 16-22, Tuxtla Gutierrez, Mexico. Springer; 2014. p. 431–442.
[14] Hahmann T. CODI: A Multidimensional Theory of Mereotopology with Closure Operations. Applied Ontology. 2020;15(3):251–311.
[15] Randell DA, Cui Z, Cohn AG. A spatial logic based on regions and connection. In: Conf. on Principles of Knowledge Representation and Reasoning (KR'92), Oct 26-29, Cambridge, MA. Morgan Kaufmann; 1992. p. 165–176.
[16] Gotts NM. Formalizing commonsense topology: the INCH calculus. In: Symp. on Artif. Intell. and Math. (ISAIM 1996), Fort Lauderdale, FL; 1996. p. 72–75.
[17] Stephen S. Improving Model-Finding for Integrated Quantitative-Qualitative Spatial Reasoning with First-Order Logic Ontologies [PhD Thesis]. University of Maine; 2020.
[18] McCune W. Mace4 reference manual and guide. arXiv preprint cs/0310055. 2003.
[19] Janota M, Suda M. Towards Smarter MACE-style Model Finders. In: Conf. on Logic for Programming, Artificial Intelligence and Reasoning (LPAR-22), Nov 17-21, Awassa, Ethopia. EPiC Series in Computing 57; 2018. p. 454–470.
[20] Ben-Ari M. Mathematical logic for computer science. Springer; 2012.
[21] Zhang H, Zhang J. MACE4 and SEM: A Comparison of Finite Model Generators. In: Bonacina MP, Stickel ME, editors. Automated Reasoning and Mathematics: Essays in Memory of William W. McCune. Springer; 2013. p. 101–130.
[22] De Giacomo G, Lenzerini M. TBox and ABox reasoning in expressive description logics. In: Conf. on Principles of Knowledge Representation and Reasoning (KR'96), Nov 5-8, Cambridge, MA. Morgan Kaufmann; 1996. p. 316–327.
[23] Horrocks I, Sattler U, Tobies S. Reasoning with individuals for the description logic SHIQ. In: Conf. on Automated Deduction (CADE-17), Jun 17-20, Pittsburgh, PA,. Springer; 2000. p. 482–496.
[24] Gomes CP, Kautz H, Sabharwal A, Selman B. Satisfiability solvers. In: Handbook of Knowledge Representation. Elsevier; 2008. p. 89–134.
[25] Beth EW. On Padoa's method in the theory of definition. Indagationes Math. 1953;15:330–339.
[26] Stephen S, Hahmann T. Formal Qualitative Spatial Augmentation of the Simple Feature Access Model. In: Conf. on Spatial Information Theory (COSIT-19), Sep 9-13, Regensburg, Germany. LIPIcs Vol. 142; 2019. p. 15:1–14.
[27] Riazanov A, Voronkov A. The design and implementation of Vampire. AI Commun. 2002;15(2–3):91–110.
[28] Ten Cate B, Franconi E, Seylan I. Beth definability in expressive description logics. Journal of Artificial Intelligence Research. 2013;48:347–414.
[29] Wang K, Wang Z, Topor R, Pan JZ, Antoniou G. Eliminating Concepts and Roles from Ontologies in Expressive Description Logics. Comput Intell. 2014;30(2):205–232.
[30] Katsumi M, Grüninger M. Theorem proving in the ontology lifecycle. In: Conf. on Knowledge Engineering and Ontology Development (KEOD 2010), Oct 25-28, Valencia, Spain; 2010. p. 37–49.
[31] Hinrichs TL, Genesereth MR. Extensional reasoning. In: CADE-21 ESARLT Workshop, Jul 17, Bremen, Germany. CEUR Vol. 257; 2007. p. 6:1–10.
[32] Ramachandran D, Reagan P, Goolsbey K. First-orderized research Cyc: Expressivity and efficiency in a common-sense ontology. In: AAAI'05 Workshop on Context and Ontologies, Technical Report WS-05-01. AAAI Press; 2005. p. 33–40.
[33] Khasidashvili Z, Korovin K. Predicate elimination for preprocessing in first-order theorem proving. In: Conf. on Theory and Application of Satisfiability Testing (SAT 2016), Jul 5-8, Bordeaux, France. Springer; 2016. p. 361–372.

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

doi:10.3233/FAIA200676

Ontology-Driven Cross-Domain Transfer Learning

Mattia FUMAGALLI [a1], Gábor BELLA [b], Samuele CONTI [b] and Fausto GIUNCHIGLIA [b]

[a] *Conceptual and Cognitive Modeling Research Group (CORE), Free University of Bozen-Bolzano, Bolzano, Italy*
[b] *Department of Information Engineering and Computer Science (DISI) - University of Trento, Italy*

Abstract. The aim of transfer learning is to reuse learnt knowledge across different contexts. In the particular case of *cross-domain transfer* (also known as *domain adaptation*), reuse happens across different but related knowledge domains. While there have been promising first results in combining learning with symbolic knowledge to improve cross-domain transfer results, the singular ability of ontologies for providing classificatory knowledge has not been fully exploited so far by the machine learning community. We show that ontologies, if properly designed, are able to support transfer learning by improving *generalization* and *discrimination* across classes. We propose an architecture based on *direct attribute prediction* for combining ontologies with a transfer learning framework, as well as an ontology-based solution for cross-domain generalization based on the integration of top-level and domain ontologies. We validate the solution on an experiment over an image classification task, demonstrating the system's improved classification performance.

Keywords. transfer learning, ontology, domain adaptation, ontologies and machine learning, generalization

1. Introduction

A long-standing ambition of research in artificial intelligence has been to achieve and exceed the human ability for generalization and telling things apart. While in the last decade, statistical approaches and machine learning (ML) in AI have been vastly successful in solving complex tasks, most solutions proceed by learning stand-alone models from large amounts of raw or annotated data. *Transfer learning* has been the research area that tries to simulate generalization by applying learned models to new tasks [1,2,3].

However, despite undeniable progress in recent research on transfer learning techniques, generalization of ML knowledge is still considered an open problem, with quantitative results in real-world scenarios rarely reaching the level of practical applicability [4,5,6]. Recently, ML communities have started experimenting with the integration of resources and methods of formal knowledge representation into learning systems in the hope of surpassing the current performance plateau of pure learning-based systems.

[1]This paper was written under a contract with the University of Trento.

As *"formal, explicit specifications of a shared conceptualization"* [7], ontologies have been designed to enable the reusability of information, as rich semantic data structures encoding previously acquired knowledge, whether general or domain-specific.

The main focus of research in this area has so far been to address the semantic gap between symbolic and learning-based (statistical or neural) representations of knowledge. There have been significant cases of success which proved the feasibility of the reuse of symbolic knowledge in transfer learning tasks [8,9,10,11]. However, no particular attention was made as to how the ontological design of these resources affects the key abilities of the transfer learning framework to *generalize* and to *discriminate* across classes. Likewise, there have been few attempts at reusing the vast amounts of existing ontological and other (formal or semi-formal) domain knowledge resources [12,13].

Our paper proposes a novel solution for integrating ontologies into state-of-the-art transfer learning methods, in order to increase their ability for generalization and discrimination across classes and, ultimately, their classification performance. Our solution consists of (1) a theoretical framework that justifies the use of ontologies in the context of transfer learning and exploits them more deeply than state-of-the-art methods; (2) a practical architecture for combining learning with the generalization and discrimination ability provided by ontologies; (3) a method for combining domain and top-level ontologies within the architecture in order to increase the generalization ability in cross-domain transfer tasks; and (4) an experimental validation of the theory and architecture over a cross-domain image classification task.

The paper is structured as follows. Section 2 provides theoretical motivations for the successful use of ontologies for transfer learning. Section 3 describes the architecture and method for combining a learning framework with ontologies. Section 4 validates the solution on cross-domain image classification using both domain and top-level ontologies. Sections 5 and 6, finally, describe related work, conclusions, and perspectives.

2. The Role of Ontologies in the Transfer of Knowledge

While the range of transfer learning techniques is vast [1,2,14], their common goal is knowledge reuse: what is learnt for completing a task in a particular domain, should be reusable in other, (to a certain extent) overlapping domains. Thus, an encyclopedic text corpus annotated for the recognition of person and product names can be reused to recognize patient and drug names in medical texts, or a system trained to classify facial expressions as happy or sad could be reused for the detection of mental health problems.

The idea of our paper, in one sentence, is to inject ontological knowledge into a transfer learning system in order to increase and make explicit the overlapping knowledge, thereby improving the performance of the transfer.

We begin by defining the notions of *domain*, *(classification) task*, and *transfer* within the probabilistic learning paradigm. We then link these notions to *ontological domains*, *classes* and *properties*, and show how ontological knowledge can be used to solve problems of knowledge generalization within the learning paradigm. The contents of this section provide the motivations of the architectural solution presented in section 3.

2.1. Transfer within the Machine Learning Paradigm

In the following, we provide a standard probabilistic formalization of the notions of domain and task in ML by adopting the definitions provided in [2].

A domain $D = \{\mathscr{X}, P(X)\}$ *consists of a feature space* $\mathscr{X}$ *and a marginal probability distribution* $P(X)$ *over the feature space, namely* $X = \{x_1, ..., x_n\} \in \mathscr{X}$. *A* task $T = \{\mathscr{Y}, P(Y|X)\}$ *consists of a label space* $\mathscr{Y}$ *and a conditional probability distribution* $P(Y|X)$ *that is learned from the training data consisting of pairs* $x_i \in X$ *and* $y_i \in Y$.

Given a source domain D_S*, a corresponding source task* T_S*, as well as a target domain* D_T *and a target task* T_T*, the objective of* transfer learning *is to improve the learning of a conditional probability distribution* $P(\mathscr{Y}_T|\mathscr{X}_T)$ *in* D_T *with the information gained from* D_S *and* T_S *where* $D_S \neq D_T$ *or* $T_S \neq T_T$.[2]

We will call *cross-task transfer learning* the case when $T_S \neq T_T$, and *cross-domain transfer learning* when $D_S \neq D_T$. In the practice of transfer learning, the cross-domain case most often means $D_S \subset D_T$ and $\mathscr{Y}_S \subset \mathscr{Y}_T$, i.e. that the source domain and label space are *extended* (or with a more widely used term: *adapted*) by fusing them with the target domain and label space. In other words, the system learns to recognize new classes without "forgetting" about previous ones.

The work discussed in [2] identifies four main scenarios where transfer learning is needed: (1) $\mathscr{X}_S \neq \mathscr{X}_T$: difference in the feature spaces; (2) $P(X_S) \neq P(X_T)$: difference in the marginal probability distributions; (3) $\mathscr{Y}_S \neq \mathscr{Y}_T$: difference in the label spaces; and (4) $P(Y_S|X_S) \neq P(Y_T|X_T)$: difference in the conditional probability distributions. The first two cases are considered as cross-domain transfer while the last two cases are cross-task transfer within the same domain. In general, all transfer learning setups solve a problem of mapping unseen data (features, labels) to what has already been learnt.

2.2. Transfer within the Ontological Paradigm

To understand how ontologies can support such a mapping, we draw a parallel between the scenarios above and ontology-based classification. For the scope of our work, we consider only the taxonomical and class–property relationships encoded in ontologies.

We model an ontology taxonomy *(that we abbreviate simply as "ontology" in the rest of the paper*[3]*) as* $\mathscr{O} = \langle C, P, I, \Phi, \Psi \rangle$. *We take* $C = \{c_1, ..., c_n\}$ *to be the set of* classes *of* $\mathscr{O}$*,* $P = \{p_1, ..., p_n\}$ *the set of* properties *of* $\mathscr{O}$*, I a binary relation such that* $I \subseteq C \times P$*, which expresses which classes* are associated to *which properties.* Φ *and* Ψ *represent the "class hierarchy" and the "property hierarchy", respectively: a set of directed edges in the form of* $c_i \overset{is-a}{\rightarrow} c_j$ *(resp.* $p_i \overset{is-a}{\rightarrow} p_j$*) where* $c_i \in C$ *(resp.* $p_i \in P$*) is the child and* c_j *(resp.* p_j*) the parent of the directed edge.*

---

[2]Notice that, in this paper, the notion of "domain" is heavily grounded on transfer learning related work. This is also the case for the notion of "cross-domain classification". We are aware that these concepts can be long debated, especially considering their interpretation in the context of KR and ontological analysis. We foresee a deeper analysis of the semantics of these terms, across different research areas, as immediate future work.

[3]For brevity, we allow ourselves this simplification in terminology with respect to the canonical definition of the term.

We say that a class is *associated to* a property when the latter is used to describe the former, and that a property is *associated to* a class with the dual meaning. Furthermore, class c is *in the domain of a property* p, $c \in \mathrm{Dom}(p)$, when c is associated with p.[4]

According to [15], *domain ontologies "describe the vocabulary related to a generic domain (such as medicine, or automobiles)"*. A domain ontology $\mathcal{O}_D$ is then an ontology with classes, properties, hierarchies, and instances that are specific to—though not necessarily exclusive of—the domain. Within the ontological paradigm, cross-domain classification means classifying against a different domain ontology $\mathcal{O}_{D_T}$. The classes that form the source and target label spaces, $C_S \subseteq C_{D_S}$ and $C_T \subseteq C_{D_T}$, resp., are classes of two distinct ontologies, with possible heterogeneity in naming, hierarchical structure, granularity, etc. These major divergences are analogous to the distinct feature spaces and marginal probabilities in the probabilistic paradigm.

2.3. Ontology Design for Transfer Learning

The core idea underlying our paper is that symbolic knowledge resources, when properly organized according to ontological principles, provide two major tools for cross-domain transfer: *generalization* and *discrimination*. While generalization uncovers the overlap between the semantic spaces across domains, discrimination provides distinctive properties of classes as key features to the transfer learning process.

By discrimination *we understand the requirement that, in an ontology, classes are used to represent in an abstract but formal way the properties that are definitional of a group of instances in any possible world* [7].

While the idea that classes are characterized by their properties is in itself almost trivial, the stronger formal ontological requirement that sibling classes must be distinguished between each other by at least one discriminating property is far from being universally applied in real-world knowledge resources.[5] The use of such ontology-derived discriminating properties as features lends more discriminating power to ML-based classification methods.

As a theoretical background for *generalization*, we reuse the *theory of abstraction* from [16], further explored in [17]. We adopt as a typology of generalization the three major kinds of abstraction defined there:

- *predicate abstraction* where a predicate (in our case, a class) is mapped to a more general one, e.g. $\mathit{Car}(X) \rightarrow \mathit{Vehicle}(X)$;
- *domain abstraction* where a constant or function (in our case, a property) is mapped to a more general one, e.g. $\mathit{fatherOf}(X) \rightarrow \mathit{parentOf}(X)$;
- *propositional abstraction* where one or more arguments of a predicate are dropped, e.g. $\mathit{isSiblingOf}(X,Y) \rightarrow \mathit{Sibling}(X)$.

Thus, by generalization *we understand the application of predicate, domain, or propositional abstraction operations to any class or property of an ontology.*

Generalization helps the transfer learning method take advantage of the hierarchical relatedness of classes and properties beyond mere equivalence based on their labels.

[4]Notice that, this particular interpretation of the notions of "class" and "property" is derived from research work on embedding rich knowledge structures into ML models. For this paper, we relied on previous research results, see [12] for further details.

[5]For instance, `schema.org` has many classes defined that only differ in their names.

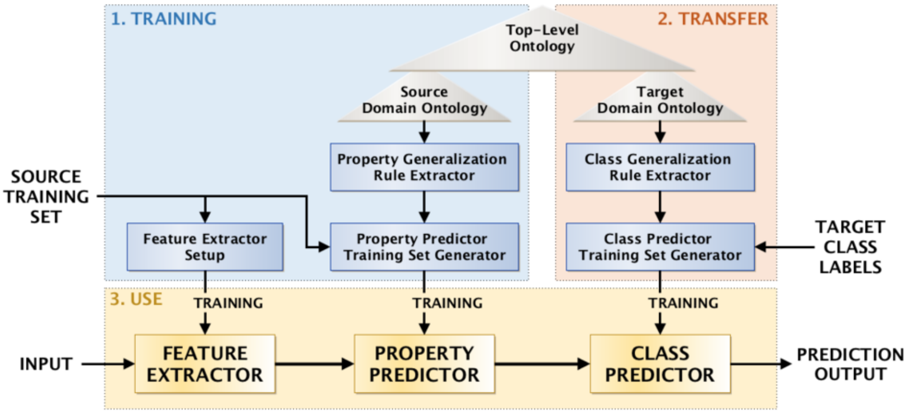


Figure 1. Architecture of the ontology-based cross-domain transfer learning method.

For example, in a *named entity recognition* task, a general learning model trained on *organization* classes can be reused in the medical domain to recognize names of *clinics* (predicate abstraction). Likewise, a model that recognizes the property of *having legs* can be used to identify *having limbs* (domain abstraction).

In a cross-domain scenario, however, the lack of a unifying theory for the two domains may prevent abstraction from providing results. In this case, the integration of two domain ontologies with a "cross-domain" ontology (e.g. a *core* or *top-level ontology*) may enable abstraction. The use of top-level ontologies is thus a crucial element of our solution for supporting cross-domain transfer.

Generalization and discrimination are used in combination for efficient transfer. For example, generalization allows the discovery that *Neurologist* (from the source domain of *Neurology*) and *Surgeon* (from the target domain of *Surgery*) are both *Doctors*. Then, discrimination via the property *operates-on-brain* allows the distinction within the target domain between *Neurosurgeon* and, say, *Vascular surgeon*.

3. Architecture and Process

Figure 1 shows the high-level architecture of our solution. It is divided into three main phases: *(1) training*, *(2) transfer*, and *(3) use*. The cross-domain classification tasks, which are the goal of the architecture, are executed in the *use* phase by a three-step pipeline inspired by the *Direct Attribute Prediction* (DAP) technique.

DAP is a state-of-the-art transfer learning method described, among others, in [18]. The main principle behind DAP is, rather than performing a direct classification on the input, to start by predicting the properties (attributes) of the object in input and then predict the class based on the properties. The mapping of the learning-based (neural, probabilistic, etc.) and ontology-based representations is achieved through the use of ontology class and property labels as the label spaces of both property and class prediction.

The strengths of DAP are its ability to *predict unseen classes* and to *simplify the prediction task* (for sensory input such as images, it is often easier to predict a small set of properties as an intermediate step rather than to predict classes directly). Property pre-

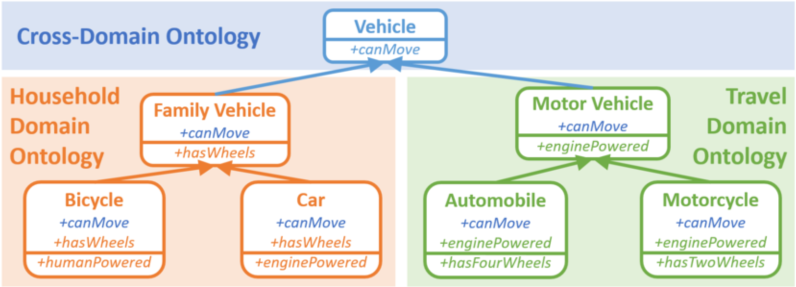


Figure 2. Example (toy) ontologies demonstrating ontology design principles that result in efficient prediction.

diction is also *more generic:* while the class definitions are often specific to the domain of use, it is less the case for properties that encode lower-level information, therefore property predictors are "transferred" more easily. Accordingly, the pipeline consists of:

1. a *Feature Extractor* $f_E : \mathscr{I} \mapsto \mathscr{X}_S$ that takes the input data $\mathscr{I}$ to be classified (text, images, etc.) and extracts a high number of low-level patterns or *features*;
2. a *Property Predictor* $f_P : \mathscr{X}_S \mapsto P_{D_S}$ that takes low-level features as input and predicts a set of higher-level *properties* from them;
3. a *Class Predictor* $f_C : P_{D_S} \mapsto \mathscr{Y}_T$ that emits a class label based on the properties previously predicted.

The purpose of the *training* and *transfer* phases preceding the use of DAP is to train the pipeline for cross-domain classification. The *training* phase is run initially and *only once*, in order to set up and train the feature extractor and the property predictor based on an initial *source training set*. The *transfer* phase is run every time the system needs to be adapted to a new domain.

3.1. Ontology Design Principles for Transfer Learning

The difference of our architecture with respect to state-of-the-art DAP pipelines resides in its use of ontologies—designed by expert communities according to ontological principles—as sources of class labels, property labels, and class–property associations.

In a cross-domain scenario it is typically not possible to find a single ontology that describes the classes and properties of both the source and the target domain adequately. We therefore apply a standard ontology engineering approach—so far unexplored in the context of transfer learning—where two distinct *source* and *target domain ontologies* are interconnected with a *top-level ontology* (in the following: TLO, as depicted in Figure 1). The TLO can be a genuine top-level ontology such as DOLCE or PROTON but, in order not to be too restrictive, we also allow other kinds of higher-level ontologies capable of subsuming the source and target domain ontologies. The root nodes of the domain ontologies need to be connected by *is-a* relations to the nodes of the TLO.

While an experimental account on the influence of all ontology design principles on transfer learning performance is beyond the scope of this paper (and is foreseen as future work), we provide a simple illustration of how an ontology design that is aware of the generalization and discrimination principles helps improve transfer results. As an example, let us consider an image classification task that uses the very simple ontologies in Figure 2. Suppose we have a Property Predictor trained on different kinds of family vehicles and other objects from the *household domain*. We would like to reuse this predictor to find cars on a different set of images related to the *travel domain*. In order to improve transfer, we reuse the tiny domain ontologies and TLO from Figure 2.

Discrimination is supported by the domain ontologies by always associating distinct property sets to non-equivalent classes, e.g. *Automobiles* and *Motorcycles* are distinguishable by the number of wheels (let us not consider the *Piaggio Ape* or the *Reliant Robin* this time). However, when comparing the property set {*hasFourWheels*} specific to *Automobile* with {*enginePowered*} specific to *Car*, no overlap is found, making property-based cross-domain transfer impossible without relying on the class hierarchy.

Generalization helps in this case: through property inheritance from domain and top-level superclasses, the property sets are extended to {*canMove*, *enginePowered*, *hasWheels*} and {*canMove, enginePowered, hasFourWheels*}, respectively. *Car* and *Automobile* have now two properties out of three in common, which the Class Predictor can use to infer the right class label output. At the same time, the *canMove* property gained from the TLO also provides discriminative power, e.g. for the distinction between vehicles and other household objects.

In case a property hierarchy is available, domain abstraction can also be applied as further generalization. Let us suppose that the TLO defines $\Psi_{TLO} = \{hasFourWheels \overset{is\text{-}a}{\rightarrow} hasWheels, hasTwoWheels \overset{is\text{-}a}{\rightarrow} hasWheels\}$. Then classification results are further improved as the similarity of the property sets of *Automobile* and *Car* are further increased, as the property set of *Automobile* will be extended to {*canMove*, *enginePowered*, *hasFourWheels*, *hasWheels*}.

3.2. Training Phase

In this first phase, a *Feature Extractor* f_E and a *Property Predictor* f_P are set up and trained. This is done only once for any given type of input, although the two components may need to be re-adapted if the input data representation changes considerably.

Feature extraction is a classic early step of machine learning processes whose purpose is to reduce the complexity of the input by extracting only its characteristics that are the most relevant with respect to the subsequent prediction task. Cross-domain transfer learning exploits the phenomenon that low-level features are often domain-independent and are thus reusable across domains. Thus, in cross-domain scenarios such as ours, feature extraction needs to be broad and generic as opposed to concentrating on domain and task-specific features. Recent trends in deep learning have been to use huge "universal" pre-trained models as the basis for subsequent domain tasks, such as *ImageNet* or, in the case of text, *BERT*. The use of pre-trained models avoids the need to customize and train feature extraction for property prediction.

The Property Predictor is typically a machine learning component that maps the features $\mathcal{X}_S$ extracted from the input to higher-level *properties*. This supposes the existence of a training set (marked as *source training set* in Figure 1) that needs to be labeled with property names. Therefore, the two challenges are: (1) the choice of properties to use for annotation; (2) the annotation task itself.

The properties used for annotation are provided by the *Property Generalization Rule Extractor* component. All properties P_S are extracted from the *source domain ontology* $\mathcal{O}_{D_S}$ and the TLO $\mathcal{O}_{D_{TLO}}$ (Figure 1): true to the nature of transfer learning, at the moment of training, the target domains are not yet known. If a property hierarchy Ψ_{D_S} is available in $\mathscr{O}_{D_S}$ then, according to *domain abstraction* (see section 2.3), each property is enriched into a property set with its ancestor properties. If $X_i \subset \mathscr{X}_S$ is the set of features corresponding to the i^{th} input data element observed, and $\text{Ann}(X_i, p_j)$ its annotation by a property p_j, then we model domain abstraction as:

$$\text{Ann}(X_i, p_j) \wedge (p_j \overset{is\text{-}a}{\rightarrow} p_k) \Rightarrow \text{Ann}(X_i, p_k) \tag{1}$$

For example, if an image of a person was annotated by a property *"has-leg"* then a new super-property *"has-limb"* will be added as annotation, provided that such a super-property exists in the property hierarchy Ψ_{D_S}.

The once-and-for-all annotation task itself is performed by the *Property Predictor Training Set Generator* component. In practice, this component can be implemented in multiple ways depending on the data available: supervised or unsupervised, automated or manual. In another DAP setup for image classification, [19] uses an unsupervised NLP-based method to extract property names from captions associated to images. In our experiment in section 4, we manually extended the annotations of an existing dataset, over a 12.500-image corpus.

3.3. Transfer Phase

The goal of the transfer phase is to adapt the DAP pipeline to specific classification domains and tasks. Adaptation is achieved by automatically training the *Class Predictor* to determine the most likely class(es) based on the properties predicted and the classification labels $\mathscr{Y}_T$ of the target domain and task. As in other DAP-based solutions, the Class Predictor is implemented as a machine learning component. The source of both the predicted class labels $\mathscr{Y}_T$ and the property–class associations used for training is the *target domain ontology*; it can, however, also be the TLO if the target classes are very general (Figure 1). The TLO also plays an important role by connecting the source and target domain ontologies. Having a common TLO increases the amount of properties shared between source and target classes, improving cross-domain transfer.

The role of the *Class Generalization Rule Extractor* component is to extract class–property associations (in the form of $\text{Dom}(p_i, c_j)$, meaning that the class c_j is in the domain of the property p_i, using the term "domain" in the mathematical sense here) from the entire ontology (i.e. the combination of the three ontologies). The associations are *generalized* because *predicate abstraction* (formula 2) and *domain abstraction* (formula 3) are taken into account:

$$\text{Dom}(p_i, c_k) \wedge (c_j \overset{is\text{-}a}{\rightarrow} c_k) \Rightarrow \text{Dom}(p_i, c_j) \tag{2}$$

$$\text{Dom}(p_i, c_k) \wedge (p_i \overset{is\text{-}a}{\rightarrow} p_j) \Rightarrow \text{Dom}(p_j, c_k) \tag{3}$$

meaning, respectively, that subclasses "inherit" the properties of their superclasses and that if a property is associated to a class then its superproperties are also associated to it.

From the set of class–property associations, the *Class Predictor Training Set Generator* component automatically generates a training corpus for the Class Predictor, taking as input parameter the subset of class labels $\mathscr{Y}_T$ that need to be predicted (it is highly unlikely that prediction should cover *all* domain ontology and TLO classes). For each class $c_i \in \mathscr{Y}_T$ and each property p_j such as $\text{Dom}(p_j, c_i)$ according to the abstraction rules above, a training instance (c_i, p_j, w_{ij}) is generated where w_{ij} is a weight computed by the Training Set Generator. In the simplest of setups, $w = 1$ for every training instance. Finer-grained training can be achieved, however, by computing weights according to additional characteristics of the ontology, such as the distance in the *is-a* hierarchy of a class or a property from the root, or by taking meta-properties (e.g. rigidity) into account.

The ability of quantitative reasoning with weights justifies our use of a machine learning component for class prediction, as opposed, e.g., to a simple logical inference engine. The Class Predictor accepts weighted input both as training in the transfer phase (as generated by its Training Set Generator) and for prediction in the use phase (given that the Property Predictor provides a weighted output). A further reason for the use of machine learning in the transfer phase is its ability to accumulate evidence, which can be exploited by combining training sets derived from multiple target domain ontologies.

3.4. Use Phase

In the use phase, the DAP pipeline, including the adapted Class Predictor, is used to solve the cross-domain classification problem. The Feature Extractor $f_E : \mathscr{S} \mapsto \mathscr{X}_S$ component takes sensory inputs $s_i \in \mathscr{S}$ and outputs the associated features $x_i \in \mathscr{X}_S$. The output of the Property Predictor $f_P : \mathscr{X}_S \mapsto P_S \times \mathbb{R}[0;1]$ consists of a list of confidence values (between 0 and 1) for each property, associated to each input sensory observation. This output is then given as input to the Class Predictor $f_C : P_S \times \mathbb{R}[0;1] \mapsto \mathscr{Y}_T \times \mathbb{R}[0;1]$. It runs a prediction for each class in $\mathscr{Y}_T$ and for each class outputs a confidence value again between 0 and 1.

4. Evaluation on Image Classification

The goal of our evaluations was to understand the impact of ontology-based knowledge on transfer learning in general and on the results obtained from our DAP-based architecture in particular. For comparability with state-of-the-art results, we opted for an image classification problem, reusing and extending the *aPascal-aYahoo* dataset and annotations[6] introduced for the first time in [20] and also reused in [18]. This dataset consists of the union of a subset of the *PASCAL VOC 2008* dataset and of images that were collected using the Yahoo image search engine, popular for image classification tasks.[7] The dataset covers a wide range of people, animals, as well as artefacts such as vehicles, furniture, etc. We manually sampled this dataset for over 12,000 images relating to either of the two domains used in our experiment: the *household domain* with 7,490 images about everyday objects around the house, serving as our training set, and the *travel domain* with 5,203 images about holidays and travel-related objects, serving as our test set. As part of cross-domain transfer, we evaluated both *cross-task* performance with the prediction of *unseen classes* (i.e. not used in training), and *same-task* performance where the same classes are predicted across domains.

The evaluations covered three setups:

- a *SoA "Flat" Classifier* that implements the state-of-the-art DAP pipeline as described in [18], based on flat class–property lists without a hierarchy nor ontological design;

[6] http://vision.cs.uiuc.edu/attributes/

[7] http://www.pascal-network.org/challenges/VOC, https://images.search.yahoo.com

- a *Domain Ontology Classifier* that uses a source domain ontology for training and a target domain ontology for testing;
- a *Domain+TLO Classifier* that, in addition to using the same domain ontologies, also uses a top-level ontology.

Our goal with introducing the second setup—that lacks the TLO—was to get an idea of the effect of the TLO itself on the results.

4.1. Ontology Design

As basis for our classification task, we used the *PASCAL Visual Object Classes*[8] resource. While it is not itself an ontology, it is widely used in the image classification literature, and for cross-comparability with SoA results we decided to adopt it.

From this database we extracted 20 *base classes* that best describe the images from the source and target domains. Seven of the 20 classes belonged to the household domain, eight to the travel domain, and five were shared between the two domains. These base classes correspond to *basic-level categories* [21], such as *bicycle*, *cat*, or *boat*. We then organized these domain-related classes and properties into two domain ontologies, by introducing superclass domain hierarchies following the principles in section 3.1. As reference material, we used the linked data and ontological analysis service *Ontobee*, and in particular ontologies such as FOODON and NCIT[9].

Based on these resources, we proceeded to build the two taxonomies, shown in Figure 3, as follows. As leaves, we created *base classes* (*Cat*, *Dog*, *Train*, etc., in green) that have a one-to-one correspondence with the original aPascal classes (that we call *aPascal-Cat*, *aPascal-Dog*, etc., in white). As the aPascal classes did not respect the principle of discrimination (for example, the property set of *aPascal-Train* was a subset of that of *aPascal-Boat*), wherever it was necessary we extended the "legacy" aPascal attribute sets with definitional properties[10] on the base class level (e.g. *Train* had all properties from *aPascal-Train* as well as the new definitional property *has-locomotive*). Note that the property sets of base classes shared across the two domains (e.g. *Car* or *Dog*) typically overlap only partially, as it would be the case with real-world domain ontologies.

Then, continuing in a bottom-up manner, we created new levels of domain-specific superclasses based on the domain hierarchies retrieved using *Ontobee*, each time introducing discriminative properties as necessary. We call these *mid-level classes*. Finally, we introduced three layers of *top-level classes* shared across the two domains (in blue in Figure 3): the root class *Entity*, then *Agent* and *Product*, and finally *Vehicle*, *Instrument*, *Living-thing*, and *Furniture*. The ontologies were built *a priori* and then unchanged during the experiments.

Thus, the final *household* domain ontology contains 27 classes, the *travel* ontology 26 classes, and the shared top-level ontology seven classes. The 63 properties reused from Pascal VOC were completed by 20 domain and 8 top-level properties, the total number of properties reaching 91. We did not organize the properties into a hierarchy and thus did not rely on domain abstraction for our results, which we leave as future work.

[8] `http://host.robots.ox.ac.uk/pascal/VOC/voc2005/index.html`

[9] `http://purl.obolibrary.org/obo/FOODON_03490214`, `www.ontobee.org/ontology/NCIT`

[10] As a theoretical reference we followed [22]; in particular, we took inspiration from the notion of *rigid properties*. However, we did not carry out a deep ontological analysis for each class. Our major objective was to identify, for each class, properties that were *essential*, i.e. always present for all instances in the dataset.

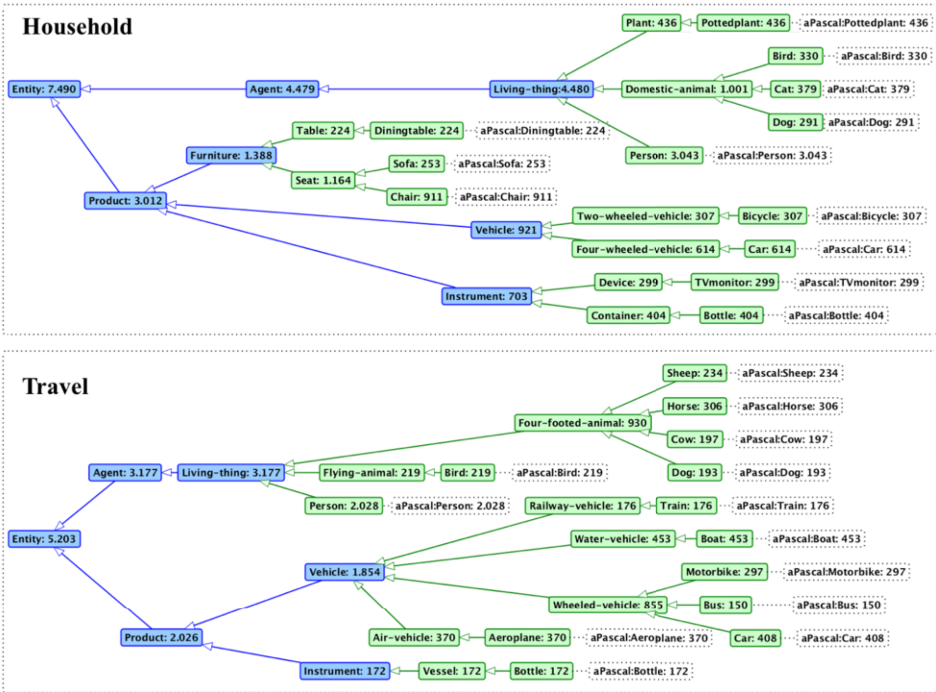


Figure 3. Selected source (training) and target (test) domains, with the number of corresponding images.

4.2. Predictor Setup

We have implemented all three components of the DAP pipeline (Figure 1) for all three setups using the *RapidMiner*[11] machine learning framework.

Feature Extractor. For this component, in all three setups we relied on the process described in [20]. For each image, we reused the precomputed *color*, *texture*, *edge orientation*, and *HoG* features that the authors extracted from the bounding boxes of the objects (as provided by the PASCAL VOC annotation) and released as part of the dataset. We deliberately used the same features for cross-comparability with works such as [18] that reuse the same datasets. The feature output was then fed into the Property Predictor.

Property Predictor. For the SoA Flat Classifier, we trained the Property Predictor on the original 63 *aPascal* properties that characterize shape, material, and the parts of the visible object. (For instance, images about motorbikes were annotated with properties such as *"plastic"*, *"metal"*, or *"engine"*.) For the Domain Classifier, this set was completed by 20 discriminative domain properties, and for the Domain+TLO classifier with the extra eight TLO properties as well. As one of the contributions of this paper, we publish the annotated dataset free for research purposes.[12] The annotated dataset was then divided according to the two domains: the *household* domain data set (7.490 images) served as training data and the *travel* domain (5.203 images) served as test, allowing us to examine cross-domain behavior. As seen in Figure 3, the two domain classes overlap

[11] http://www.rapidminer.com
[12] https://github.com/Matt-81/ontology-driven-TL

only partially: seven and eight classes are specific to the household and the travel domain, respectively. Using the annotations, we trained a *k-Nearest-Neighbor* classifier for each property (this method provided a good balance between performance and speed).

Class Predictor. We implemented the Class Predictor as a multi-label classification task. We again used *k-Nearest-Neighbor* classifiers in all three setups, which we trained on class–attribute associations extracted from the two ontologies using the *Class Generalization Rule Extractor*. For the SoA Flat classes, only the original aPascal classes and properties were used. The Domain Classifier was trained on all source and target domain ontology classes (base and mid-level) and all inherited properties. The Domain+TLO Classifier, in turn, was trained on all of the above plus the top-level classes and properties.

4.3. Experiments and Results

Our experiments measured classification performance with respect to two transfer learning scenarios: a *same-task cross-domain* and a *cross-task cross-domain* scenario. The test set of images was run through the three DAP pipeline setups using their respective property and class predictors.

Same-task cross-domain classification. Here, the goal was to classify unseen images into *seen* classes, i.e. that are defined in both source and target domains. The results, in terms of F-measure, are shown in Figure 4 for all five base classes shared between both domains: *Person*, *Car*, *Bird*, *Bottle*, and *Dog*. We also present results for the shared mid-level and top-level classes *Living-thing*, *Agent*, *Product*, *Vehicle*, and *Instrument*. The SoA Flat Classifier obviously could not provide any results for top-level and mid-level classes, as it does not use the entire ontologies. Similarly, the Domain Classifier did not have results for the two top-level classes as it does not use the TLO. In the case of base classes, where all three setups could be tested, the two ontology-based pipelines consistently reach higher scores, the difference with respect to the SoA running from moderate (4% for *Person*, 7% for *Car*), to very high (17% for *Bird*, 35% for *Bottle*). The exception is *Dog*, which all pipelines consistently misclassified as *Cat*. With respect to the TLO, only in the case of *Bottle* do we observe a major positive effect. It plays a more important role, however, for mid-level classes: the prediction of *Vehicle* and *Instrument* is improved by the TLO in a major way (by 15% and 45%, resp.). This also explains the improvement of the TLO on *Bottle*: it is the only *Instrument* class in the target ontology, and the latter is discriminated by a top-level attribute. For more general classes, the extra discriminative properties provided by the TLO have a clearly positive effect.

Cross-task cross-domain classification. Here, the goal was to classify unseen images into unseen classes, i.e. that are not part of the training household domain but are defined within the target travel domain ontology. The results on all eight unseen classes can be found in the right-hand side of Figure 4. Although to greatly varying degrees, all three classifiers were able to detect all unseen classes, demonstrating the efficiency of Direct Attribute Prediction. As in the same-task scenario, improvements with respect to the SoA classifier were consistent, from 2% (for *Sheep*) up to 33% (for *Train*). When comparing the Domain and the Domain+TLO classifiers, the differences between the two are minor, with slight improvements for certain classes and slight deteriorations for others. Overall, we do not observe a major benefit from the TLO.

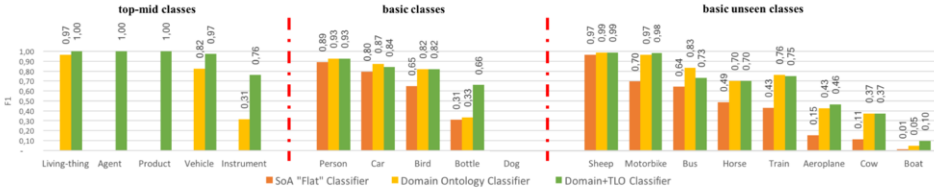


Figure 4. Prediction results (F1): same-task on seen top/mid (left) and base classes (middle), cross-task on unseen base classes (right), for the three classifier setups. Top-level results could only be obtained by the Domain+TLO classifier. Mid-level results were obtained by the two ontology-based classifiers.

Table 1. Overall accuracy over the entire dataset.

Classes	*SoA Flat Classifier*	*Domain Classifier*	*Domain+TLO Classifier*
base	**58.50%**	**72.84%**	**74.59%**
mid-level	–	**85.32%**	**95.46%**
top-level	–	–	**100%**

Overall accuracy. Finally, in Table 1 we report global accuracy results across the entire test dataset, for the three setups and on the three class levels (top, mid, base). On base classes, a major difference (+14.34%) is observed for the domain-ontology-based setup with respect to the SoA. Further improvement due to the TLO is moderate (+1.75%). The positive effect of the TLO is more obvious on the top and mid-level, where the TLO is clearly enabling high-precision and high-recall prediction. The TLO also seems to ensure a fully accurate coarse-grained distinction between *Agents* and *Products* (i.e. animate and inanimate objects).

Discussion. While we are careful not to draw overly general conclusions from experimenting with a single set of ontologies, we still observe a few salient phenomena. First of all, the application of the principle of discrimination, by making sure that every class is distinguished from its siblings by at least one property, had a major positive effect on the results, as shown by the improvements with respect to the SoA Flat Classifier. The effect of generalization through the TLO, on the other hand, was minor on base classes but of a remarkably high quality (95–100%) on mid-level and top-level classes. We attribute these results to the way our Class Predictor works: in the property sets of leaf classes, the few properties inherited from the top level had a relatively low weight with respect to the number of properties introduced on lower levels. A more efficient setup, e.g. based on *hierarchical learning* [23], would take into account the high-quality predictions on top-level and mid-level classes as input for base class prediction (so that, e.g. chairs are not classified as *Person* if they were given the top-level class of *Product*). We foresee this improvement of our setup as immediate future work.

5. Related Work

Our work is a follow-up to recent efforts that introduce formal or semi-formal knowledge into transfer learning setups and, in particular, *Zero-shot learning* [10,18]. These works, and the ones cited in the introduction, use simple knowledge organization schemes, such as flat class–attribute associations designed bottom-up, without following any theoretical principles. More recent work [24,9] also used class hierarchies, effectively relying on generalization, but still in an ad-hoc informal manner, without paying attention to how the hierarchy is organized. Some works focus on how to generalize information by reusing already existing *unstructured* domain information, usually extracted from natural language text such as image captions. [11] generates a taxonomy generalizing the information codified in the data populating the training set.

While in our experiments we also used custom built ontologies, our approach and motivations were different: our goal was to explore and demonstrate the effect of ontology design principles—generalization and discrimination—on transfer learning results. Exploiting well-established principles from *knowledge representation* and *ontological analysis* [25,7,15], we propose guidelines by which existing top-down ontological resources can be efficiently reused as knowledge sources for cross-domain transfer. In our experiment, the semantic information are not extracted from a text corpus and the resulting ontology is reusable for future tasks. A further contribution of our paper is the use of various abstraction operations (proposed earlier in [16] and further explored in [17]) in the aim of more fully exploiting the generalization ability of ontologies. Works on the role of discrimination and the subsumption relation in order properly to exploit a hierarchical class structure, such as [26,27], were also considered.

Finally, works focusing on bridging the gap between "high-level" knowledge and "low-level" (i.e. perceptual) information also overlap with our efforts [28]. The role of ontologies on image annotation and management has been studied in [29,30,31]. [32] and [33] explore the different yet complementary functions of knowledge-based *classification* and perception-based *identification*. While we share motivations with these works insomuch as we aim to integrate different forms of knowledge into a single system, our focus in this paper is specifically the transfer learning problem.

6. Conclusion and Perspectives

While our work is inscribed in the general line of recent efforts that aim at combining symbolic (top-down) and learning-based (bottom-up) knowledge, we also consider it as a starting point for the investigation of the influence of formal knowledge (e.g. ontology) design on the performance of the overall combined system. Based on our first encouraging results, we plan to develop an evaluation method and framework for existing top-level and domain ontologies with respect to their performance in prediction tasks. This involves theoretical research that examines the impact of various aspects of formal ontology characteristics on prediction, as well as an actual service that evaluates existing ontologies in terms of their prediction ability.

Acknowledgement. This paper was supported by the *WeNet* project, funded by the European Union (EU) Horizon 2020 programme under grant number 823783.

References

[1] Weiss K, Khoshgoftaar TM, Wang D. A survey of transfer learning. Journal of Big data. 2016;3(1):9.

[2] Pan SJ, Yang Q. A survey on transfer learning. IEEE Transactions on knowledge and data engineering. 2009;22(10):1345–1359.

[3] Wang W, Zheng VW, Yu H, Miao C. A survey of zero-shot learning: Settings, methods, and applications. ACM Transactions on Intelligent Systems and Technology (TIST). 2019;10(2):1–37.

[4] Zhuang F, Qi Z, Duan K, Xi D, Zhu Y, Zhu H, et al. A Comprehensive Survey on Transfer Learning. arXiv preprint arXiv:191102685. 2019.

[5] Wang M, Deng W. Deep visual domain adaptation: a survey. Neurocomputing. 2018;312.
[6] Glavas G, Litschko R, Ruder S, Vulic I. How to (properly) evaluate cross-lingual word embeddings. arXiv preprint arXiv:190200508. 2019.
[7] Guarino N, Oberle D, Staab S. What is an ontology? In: Handbook on ontologies. Springer; 2009. p. 1–17.
[8] Chen J, Lécué F, Pan JZ, Horrocks I, Chen H. Knowledge-based transfer learning explanation. In: Sixteenth International Conference on Principles of Knowledge Representation and Reasoning; 2018. .
[9] Kono H, Kamimura A, Tomita K, Suzuki T. Transfer learning method using ontology for heterogeneous multi-agent reinforcement learning. International Journal of Advanced Computer Science & Applications. 2014;5(10).
[10] Palatucci M, Pomerleau D, Hinton GE, Mitchell TM. Zero-shot learning with semantic output codes. In: Advances in neural information processing systems; 2009. p. 1410–1418.
[11] Liu B, Yao L, Ding Z, Xu J, Wu J. Combining ontology and reinforcement learning for zero-shot classification. Knowledge-Based Systems. 2018;144:42–50.
[12] Giunchiglia F, Fumagalli M. Entity Type Recognition – dealing with the Diversity of Knowledge. In: 17th International Conference on Principles of Knowledge Representation and Reasoning; 2020. .
[13] Bella G, McNeill F, Leoni D, Real FJQ, Giunchiglia F. Diversicon: Pluggable Lexical Domain Knowledge. Journal on Data Semantics. 2019;8(4):219–234.
[14] Xian Y, Lampert CH, Schiele B, Akata Z. Zero-shot learning: a comprehensive evaluation of the good, the bad and the ugly. IEEE transactions on pattern analysis and machine intelligence. 2018;41(9).
[15] Guarino N. Understanding, building and using ontologies. International Journal of Human-Computer Studies. 1997;46(2-3):293–310.
[16] Giunchiglia F, Walsh T. A theory of abstraction. Artificial intelligence. 1992;57(2-3):323–389.
[17] Giunchiglia F, Fumagalli M. Teleologies: Objects, actions and functions. In: International conference on conceptual modeling. Springer; 2017. p. 520–534.
[18] Lampert CH, Nickisch H, Harmeling S. Attribute-based classification for zero-shot visual object categorization. IEEE transactions on pattern analysis and machine intelligence. 2013;36(3):453–465.
[19] Lampert CH, Nickisch H, Harmeling S. Learning to detect unseen object classes by between-class attribute transfer. In: IEEE Conference on Computer Vision and Pattern Recognition; 2009. p. 951–958.
[20] Farhadi A, Endres I, Hoiem D, Forsyth D. Describing objects by their attributes. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE; 2009. p. 1778–1785.
[21] Rosch E, Mervis CB, Gray WD, Johnson DM, Boyes-Braem P. Basic objects in natural categories. Cognitive psychology. 1976;8(3):382–439.
[22] Guarino N, Welty C. A formal ontology of properties. In: International Conference on Knowledge Engineering and Knowledge Management. Springer; 2000. p. 97–112.
[23] Bouvrie JV. Hierarchical learning: Theory with applications in speech and vision. MIT; 2009.
[24] Al-Halah Z, Stiefelhagen R. How to transfer? zero-shot object recognition via hierarchical transfer of semantic attributes. In: IEEE Winter Conference on Applications of Computer Vision; 2015. p. 837–843.
[25] Guizzardi G. On ontology, ontologies, conceptualizations, modeling languages, and (meta) models. Frontiers in artificial intelligence and applications. 2007;155:18.
[26] Guarino N, Welty C. Identity and subsumption. In: The Semantics of Relationships. Springer; 2002. p. 111–126.
[27] Johansson I. Four Kinds of 'Is A' Relation. Applied Ontology: An Introduction, Frankfurt: ontos. 2008:235–254.
[28] Coradeschi S, Loutfi A, Wrede B. A short review of symbol grounding in robotic and intelligent systems. KI-Künstliche Intelligenz. 2013;27(2):129–136.
[29] Kuang Z, Yu J, Li Z, Zhang B, Fan J. Integrating multi-level deep learning and concept ontology for large-scale visual recognition. Pattern Recognition. 2018;78:198–214.
[30] Donadello I, Serafini L. Mixing low-level and semantic features for image interpretation. In: European Conference on Computer Vision. Springer; 2014. p. 283–298.
[31] Porello D, Cristani M, Ferrario R. Integrating ontologies and computer vision for classification of objects in images. In: Proceedings of the Workshop on Neural-Cognitive Integration in German Conference on Artificial Intelligence; 2013. p. 1–15.
[32] Giunchiglia F, Fumagalli M. Concepts as (Recognition) Abilities. In: FOIS; 2016. p. 153–166.
[33] Fumagalli M, Bella G, Giunchiglia F. Towards understanding classification and identification. In: Pacific Rim International Conference on Artificial Intelligence. Springer; 2019. p. 71–84.

This page intentionally left blank

Formal Ontology in Information Systems
B. Brodaric and F. Neuhaus (Eds.)

Author Index