Ontology: The Discipline and the Tool

Doug Mayhew[1,2], Dirk Siebert[1]

[1] Institute for Formal Ontology and Medical Information Science, University of Leipzig,
04107 Leipzig, Härtelstraße 16-18, Germany
doug.mayhew, dirk.siebert@ifomis.uni-leipzig.de
[2] Department of Philosophy, University at Buffalo, New York, USA

Abstract. The fields of philosophy and informatics entertain two somewhat different conceptions of ontology. Philosophical ontology is a branch of metaphysics dating back at least to the time of Plato and Aristotle. Ontology in informatics has its origins in the artificial intelligence research of the eighties and nineties. This means that the fields of philosophy and informatics entertain two somewhat different conceptions of ontology and the present paper discusses the relationship between these two conceptions. Differences and similarities are pointed out and variations in methodological approaches are also discussed. Efforts to combine the ontological methodologies and resources of the two fields are surveyed, and actual and potential benefits and drawbacks of such collaborations are examined.

Different Concepts

The fields of philosophy and informatics entertain two somewhat different conceptions of ontology, with correspondingly different notions of what ontology is for. First, we have the age-old conception of ontology as a philosophical discipline. Second, we have the relatively new conception of ontology as an information organization tool, a notion of ontology adapted from the philosophical conception by artificial intelligence researchers and then adopted by the applications-oriented field of informatics. (The philosophical conception was recognized in informatics as early as the late sixties in data modeling research [1].) The two conceptions have their characteristic differences, which show up primarily in what each field thinks ontology is for. The following is a discussion of these differences, but also of the similarities between the two disciplines and of the ways in which they might cooperate.

In philosophy, ontology is the study of what exists. Ontology is thus of a piece with metaphysics, the branch of philosophy that studies the nature of reality, and has been a prominent area of investigation since the beginnings of philosophical inquiry. The individual ontological theories advanced in philosophy are universal, descriptive classifications of the content and structure of reality as a whole. Aristotle provides a classic example in [2], where he describes the world using ten categories—viz., substance and nine kinds of attributes or accidents. Philosophical ontologies have traditionally been expressed informally, using natural language, but Leśniewski's [3] use of an artificial formal language to represent his formal theory of parts (mereology) inaugurated philosophy's use of artificial languages and formal logic in expressing on-

tologies. Examples of contemporary philosophical endeavors which involve the symbolical representation of ontologies can be found in [4] and [5].

Informatics uses ontology predominantly as an information organization tool. Many attribute the initial use of ontology in this way to artificial intelligence research on the facilitation of knowledge sharing and re-use among software agents. As described in [6], ontology in this field often entails the development of customized terminologies (with individually specified meanings for the terms) used to create customized descriptions or models of a particular domain of some actual or constructed reality. These are language dependent ontologies rather than universal theories. An example in the medical domain is SNOMED [7]. Ontologies in informatics are often expressed as logical theories (often using one or other flavor of description logic [8]), semantic networks, or, more recently, in a modeling language such as UML [9]. They might then be converted to programming language code to become part of a software application.

Whereas philosophical ontology has traditionally sought to provide a general ontology of reality as a whole, those working on ontologies in informatics develop domain-specific ontologies designed to meet particular information processing needs and requirements. This difference in scope points to yet another difference. The philosopher seeks knowledge of what exists more or less for the sake of knowledge itself. The informatician, in contrast, is interested in developing ontologies to serve more limited and practical purposes. There is normally very little theoretical work involved, for example, in developing an ontology of a company's product or service line; this is done simply to manage inventory and accounting records. The difference here is in one's motivation for doing ontology.

There are some other differences to note. First, as indicated above, philosophical ontologies are language-independent or universal, whereas many informatics ontologies are restricted to the specific languages in which they are formulated. Another difference turns on the fact that, while philosophy has traditionally been concerned with giving ontological accounts of the natural and continuous reality that everyone experiences, informatics is often working with the closed-world realities, for instance, of particular businesses, the specific products or services with which those businesses deal, the people and activities involved, and so forth—reflecting again their underlying, specifically pragmatic motivations.

Along with the differences, there are also similarities. In both fields, the general idea—leaving aside for the moment issues of domain, scope, degree of refinement, and method of expression—is that ontology provides a means to classify entities, processes, and the relations that hold between them. There is also the commonly—though not unanimously—held belief in both fields that ontology should strive for descriptive accuracy and cross-domain communicability.

Despite the many differences between the philosophical and informatics conceptions of ontology, a number of philosophers and informaticians are now working together in various ontology projects. Individuals from both fields are surveying the relevant literature and exchanging ideas in efforts to improve upon their respective uses of ontology. The benefits and drawbacks of such efforts are discussed in the last section of the paper.

Different Approaches

The conceptual differences discussed above are differences between fields. There are also different conceptual and methodological approaches to ontology within the two fields. An account of the different approaches in philosophical ontology can be found in [10], and a condensed account of philosophical and informatics approaches is provided in what follows.

Approaches in traditional philosophical ontology fall under two basic divisions, which we might call substance versus process and reductionist versus non-reductionist. Substance-based ontologies focus on substances or things as the essential constituents of reality. Process-based ontologies describe reality primarily in terms of processes, flux, or change. Reductionist approaches claim that reality is accurately described in terms of one basic type of constituent, usually either substance or process. It is rather difficult to find examples of pure substance or process reductionism, since almost all those who put forward a substance-based ontology also admit processes of one sort or another and vice versa. One finds an essentially substance-based, reductionist ontology in materialist metaphysical doctrines like that presented in [11]. An example of process-based reductionism can be found in [12]. Non-reductionist ontologies are concerned with providing an exhaustive ontological account of reality at all levels, from the micro- to the macroscopic, including categories of both substance and process. Aristotelian metaphysics is a good example, as is the ontological work in [13] and [14].

Within informatics there are also two basic divisions, the reference versus applications division and the logic-based versus non-logic-based division. Proponents of reference ontologies [5] advocate the creation of overarching, descriptively adequate ontologies accompanied by a rich formal representation. Applications ontology is focused on low-level ontologies designed to represent the taxonomical structure of specific domains, and proponents of applications ontology praise its advantages for the practical purposes of many information systems. Commerce, research, and information-based applications developed using ontological methods of organization range from simple yet extensive domain-specific terminologies to standards development projects involving software interoperability, information search and retrieval, automated inferencing, and natural language processing. A good example of applications ontology that handles a range of these tasks can be found in [15].

Examples of research being done in both reference and applications ontology can be found in [16, 17, 18]. Much of the commercial work done within ontology is applications ontology. Among reference and applications ontologists, one can find a further division between logic- and non-logic-based approaches. Logic-based approaches rely primarily on description logic to develop a model of some domain. Non-logic-based approaches, exemplified in network-based structures (e.g., connectionist systems and semantic networks), seek to develop models that more closely resemble the observable structure and workings of human cognition. An example of this approach is seen in [19].

Not every ontological effort in either field adheres strictly to one approach or the other. Hybrid approaches are common in philosophy and informatics, and such approaches often turn out to be useful in advancing both philosophical understanding and information systems performance. The non-reductionist approach in philosophy is

itself a hybrid approach. One can find ontological projects in informatics and especially in AI that use logic in combination with non-logic-based approaches—e.g., in some of the latest software agent designs [20]. Then, of course, there are recent projects that combine approaches from the two fields of philosophy and informatics [16, 17, 18]. Hybrid approaches, as one might guess, tend to be richer in what they can express and how they express it, and thereby more complex and time consuming during the design phase.

Philosophy and Informatics Working Together

Though ontology has a long history in philosophy, it is probably safe to say that today it is known primarily from its association with artificial intelligence and informatics. This is based partly on the fact that there are many more people involved in doing research on ontology in informatics than in philosophy. There are, however, a few philosophers studying ontology as a discipline reaching beyond its philosophical origins. There are also informaticians exploring the philosophical origins of what they once thought of primarily as an information organization tool. Efforts to combine philosophical and informatics research in ontology are still in their infancy but growing steadily. Let us consider some of the actual as well as some of the potential benefits and drawbacks of philosophers and informaticians working together in ontology.

Benefits

Researchers on both sides stand to benefit, first of all, from simply being exposed to a different perspective. Combining perspectives is a good way to initiate the development of new ideas, and new ideas can turn into concrete improvements in the ways in which difficult problems are confronted. Following are some of the benefits, both mutual and one-sided, that could be reaped from the exchange of ideas and methods across the fields of philosophy and informatics.

Consider the difference between the grand-scale ontologies of philosophy and the domain-specific ontologies of informatics. Philosophers are trained to look at the big picture, to notice the content, structure, and relations of reality as a whole, and to develop a general ontological theory based on their investigations. Relations between the different aspects of reality can then be expressed in terms of the general ontological theory. This kind of training could be passed on to informaticians willing to exchange ideas with philosophers. Informaticians would learn how to develop a wider, yet cohesive view of reality into which their domain specific concerns could be integrated. One resulting potential benefit to informatics is a way to seriously improve upon systems and software interoperability and standards development.

While pure philosophical ontology is not pursued for the sake of practical ends, there are benefits that the philosopher might gain from studying informatics. Recall that informatics ontologies are often models of domains—e.g., law, commerce, and administration—where the entities are created entirely by the actions, physical and verbal, of human beings. These models can be generalized, and the result used to uncover prevalent features of our everyday lives. Philosophical efforts to account for

that have surfaced only recently in works like [21]. Philosophical ontology should ultimately seek to give an account of every aspect of reality, and the examination of informatics ontologies could result in benefits to philosophers working toward this end by providing new families of examples and also new types of problems with which to grapple.

Attention to the conceptual differences and the different approaches observed between and within the two fields may reveal possibilities for hybrid approaches that can take advantage of the best features on either side. This is a good first step in developing rich and exhaustive ontologies. For example, an ontology or ontological method like that proposed in [22], that can account for different views or partitions of reality, would be useful in both reference and applications settings. If philosophers and informaticians continue to compare the results of their research efforts, then perhaps, to the benefit of both fields, a universal descriptive ontology that covers any kind of content at every level of granularity [22] can be developed.

There is a mutual benefit associated with the practice of formalizing ontologies. The primary reason for expressing an ontology in a formal language is to help achieve higher levels of clarity, rigor, and accuracy in ontological theories and models and to ensure that the content of the ontology is easily communicated to other people and perhaps more easily translated into programming code. Collaboration between philosophers and informaticians with a background in logic may assist in the creation of better formalizations.

A further benefit that can be realized from the combined efforts of philosophy and informatics is that each field will have more manpower working on its problems. The more there are working on a given problem, the better the chances it will get solved. Subjecting a body of work to more minds and different perspectives is also an effective way to expose previously hidden difficulties. Examples in which informatics has benefited in this respect can be found in [23] and [24].

The cooperation between fields has also created openings in knowledge engineering and consultancy that can be filled by philosophers with a background in ontology and logic. Companies like *Ontology Works*, *Cycorp*, *Kanisa*, and *Language and Computing* are among those that have put philosophers to work. Working in informatics has provided some philosophers with supplemental training that will undoubtedly be of value to them in the future.

These collaborative ontological efforts promise indirect benefits to those outside philosophy and informatics as well. If the collaboration produces improvements in information systems design and functionality, then there are subsequent benefits passed on to everyone who is served directly or indirectly by information systems. Clients, customers, and patients, for example, would certainly benefit from improved efficiency and accuracy in handling their needs. Businesses would improve their ability to better serve the client and thereby benefit financially from a stable or perhaps even growing clientele.

Drawbacks

In comparison to the actual and potential benefits attributable to the collaboration between philosophy and informatics, the drawbacks are few. It is difficult to conceive of

any actual harm that could come from the two fields working together, but there are foreseeable difficulties and drawbacks.

An obvious difficulty arises with the attempt to assimilate concepts and approaches with which one is not entirely familiar. A lack of mutual knowledge and experience in the respective fields may result in the individuals talking past each other, rather than effectively communicating their ideas. The depth of professional knowledge possessed by a worker or researcher in one field may not easily transfer to someone in another, especially if the two fields are quite different. At the same time, years of training and practice in concepts and methods specific to a profession are hard to set aside, and we tend to think and communicate our ideas in terms of what we know, rather than in a context-free manner. Philosophy and informatics are sufficiently different for this difficulty to surface in collaborations between the two.

However, by providing the right environment for knowledge exchange this difficulty can be largely overcome. A learning environment and centers dedicated to philosophy and informatics research would help tremendously. (Such centers and special interest groups are being established in Buffalo, Leipzig, Rome, Trento, and Turin.) Along with sharing knowledge, supplemental training can help in overcoming this difficulty.

The difficulty with conceptual and methodological differences between the two fields points to a potential drawback concerning the time limits and predefined design guidelines imposed upon many informatics projects. Most philosophers do primarily theoretical work, and they are not under heavy time constraints. The work done in informatics is very often geared toward client-specific practical applications, and tasks must normally be completed within a relatively short time. An information system is often the backbone of operations; it is something a business, for instance, needs in order to do what it does. In other words, the people who need information systems cannot wait around while researchers figure out what the absolute best ontological design should be. The specific needs of the client must be taken into account, and the designers must develop the best system they can within the time limit set by the client.

There is a potential drawback for clients in that they must settle for what the systems designers can build for them in the time they are allotted. Put philosophers and informaticians to work on a project like this one, and we immediately see the drawbacks from their different perspectives. The philosopher wants to make sure that the ontology behind the system accurately reflects the world as it is and that any logical theory behind the design is sound and complete. The informatician may very well want the same thing, but realize that perfection must be sacrificed in the interest of finishing the job on time.

The immediate solution to this drawback is for the philosophically minded to learn how to work within predefined design guidelines and time limits. The ultimate solution is, again, to establish centers where basic research in philosophical and informatics ontology research can be carried out under conditions where time limits are dramatically relaxed. Research would be directed at developing better design guidelines in ways which could ultimately benefit everyone related to information systems.

Concluding Remarks

Collaboration between philosophy and informatics is a reality from which a number of benefits have been reaped and positive results produced. With the formation of more special interest groups, institutes, and centers dedicated to bringing philosophical and informatics ontology researchers together to learn from one another, it is reasonable to assume that more advances in both fields are on the way.

Acknowledgements

This work was supported by the Alexander von Humboldt Foundation under the auspices of its Wolfgang Paul Program.

References

1. Mealy, G. H.: Another Look at Data. In: AFIPS Conference Proceedings, Volume 31. Thompson Books, Washington, DC (1967) 525–534
2. Aristotle: Categories $1^{b}25$. In: Barnes, J. (ed.): The Complete Works of Aristotle. Princeton University Press, Princeton (1984)
3. Lesniewski, S.: Foundations of the General Theory of Sets. In: Surma, S. J., Srzednicki, J. T. J., Barnett, D. I., Rickey, V. F. (eds.): Stanislaw Lesniewski: Collected Works, Volumes I and II. Kluwer, Dordrecht (1991)
4. Simons, P. M.: Parts: A Study in Ontology. Clarendon Press, Oxford (1987)
5. Smith, B.: Basic Formal Ontology. http://ontology.buffalo.edu/bfo (2002)
6. Guarino, N.: Formal Ontology and Information Systems. In: Guarino, N. (ed.), Formal Ontology in Information Systems: Proceedings of First International Conference, ACM Press, New York (1998) 3–15
7. Systematized Nomenclature of Medicine (SNOMED). http://www.snomed.org/
8. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., (eds.): The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, Cambridge, MA (2003)
9. Unified Modeling Language™ (UML™). http://www.uml.org/
10. Smith, B.: Ontology and Information Systems. http://ontology.buffalo.edu/smith/articles/ontologies.htm (2001)
11. Hobbes, T.: De Corpora. In: Molesworth, W. (ed.). The Collected English Works of Thomas Hobbes, Volume 1: London, Taylor and Francis Books Ltd. (1997)
12. Whitehead, A. N.: Science and the Modern World. Free Press, New York (1925)
13. Ingarden, R.: Time and Modes of Being. Michejda, H. R. (trans.), Charles C. Thomas, Springfield, Illinois (1964)
14. Chisholm, R. M.: A Realistic Theory of Categories: An Essay on Ontology. Cambridge University Press, Cambridge (1996)
15. Language and Computing. LinKSuite®. http://www.landcglobal.com/pages/linksuite.php
16. Institute for Formal Ontology and Medical Information Science (IFOMIS), Leipzig, Germany: Basic Formal Ontology (BFO) project. http://ontology.buffalo.edu/bfo/
17. Laboratory for Applied Ontology (LAO), Trento and Rome, Italy. http://www.loa-cnr.it/
18. Foundational Ontology. http://www.comp.leeds.ac.uk/brandon/ontology/

19. Unified Medical Language System® (UMLS®). http://www.nlm.nih.gov/research/umls/
20. Baral, C., Gelfond, M.: Reasoning Agents in Dynamic Domains. In Minker, J. (ed.): Logic-Based Artificial Intelligence, Kluwer, Dordrecht (2000) 257–279
21. Searle, J. R.: The Construction of Social Reality. Free Press, New York (1995)
22. Bittner, T., Smith, B.: A theory of granular partitions. In: Duckham, M., Goodchild, M. F., Worboys, M. F. (eds.): Foundations of Geographic Information Science, Taylor & Francis Books, London (2003) 117–150
23. Smith, B., Ceusters, W.: The First Industrial Philosophy: How Analytical Ontology Can Be Useful to Medical Informatics. Interdisciplinary Science Reviews, 28 (2003) 106–111
24. Kumar, A, Smith, B.: The Universal Medical Language System and the Gene Ontology: Some Critical Reflections. Proceedings of the 26th German Conference on Artificial Intelligence, Hamburg, 2003 (2003)