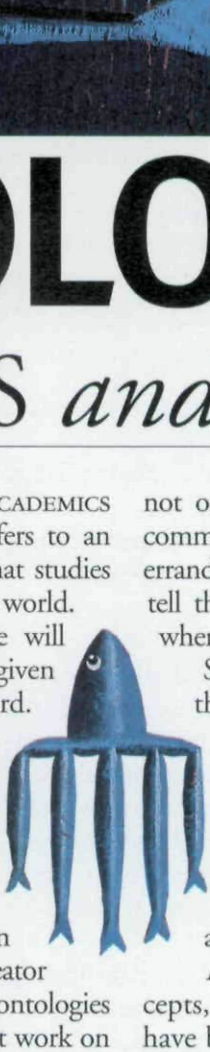
ONTOLOGY
APPLICATIONS *and* DESIGN

TWENTY YEARS AGO, MOST ACADEMICS would have said ontology refers to an esoteric field in philosophy that studies *being*—what there is in the world. Today, a Web search engine will return over 64,000 pages given "ontology" as the keyword. Among the sites appearing on the first few pages are phrases such as "enabling virtual business," "gene ontology consortium," and "enterprise ontology," sounding more practical than philosophical. In fact, Tim Berners-Lee—the creator of the Web—now considers ontologies to be a critical part of his latest work on a semantic Web, which he envisions will not only allow our software agents to communicate among themselves for our errands, but also enable our phones to tell the TV and stereo to quiet down when they ring [1].



ALL ILLUSTRATIONS BY MARTIN JARRIE

So what is an ontology and why this attention, not only from academic disciplines such as computer science, information science, and business schools, but also from industries as diverse as the high-tech, financial, medical, educational, and agricultural sectors?

As with most fashionable concepts, many definitions of ontology have been offered. A most commonly cited definition is one offered by Gru-

MICHAEL GRUNINGER AND JINTAE LEE

ber: "An ontology is a formal explicit specification of a shared conceptualization" [3]. A conceptualization, in this context, refers to an abstract model of how people think about things in the world, usually restricted to a particular subject area. An explicit specification means the concepts and relationships of the abstract model are given explicit terms and definitions.

It is not our intention in this special section to offer a universal definition for ontologies (even if such a consensus were possible); rather, we wish to focus on the uses of ontologies (see the table) and the emerging discipline of ontological engineering, which considers the entire ontology life cycle—the design, evaluation, validation, revision, and deployment of ontologies within intelligent systems.

Uses of Ontology (customized from the uses of ontology identified at the KRSL kickoff meeting 1994).
For communication
between implemented computational systems. between humans. between humans and implemented computational systems.
For computational inference
for internally representing and manipulating plans and planning information.
for analyzing the internal structures, algorithms, inputs and outputs of implemented systems in theoretical and conceptual terms.
For reuse (and organization) of knowledge
for structuring or organizing libraries or repositories of plans and planning and domain information.

Ontological Engineering

Ontological engineering has as its goal effective support of ontology development and use throughout its life cycle—design, evaluation, validation, maintenance, deployment, mapping, integration, sharing and reuse. The articles in this special section highlight several key challenges that motivate current research in this emerging discipline. For a more comprehensive discussion of the issues it needs to address, we refer to [2].

Building ontologies is difficult, time-consuming, and expensive, particularly if the goal is the design of an ontology that is formal enough to support automated inference. One reason for this is that ontologies require consensus across a community whose members may have radically different visions of the domain under consideration. In practice, the quest for consensus is dealt with in a variety of ways. At one extreme, small lightweight ontologies are developed by large numbers of people and then merged. At the other extreme, rigorous formal ontologies are developed by consortia and standards organizations. In the former case, there will be a greater need for ontology mapping and merging, while the latter case will require better support for collaborative design and analysis. Kim's article addresses this issue in the context of developing and deploying ontologies on a semantic Web. Holsapple and Joshi's article considers the challenge of collaborative ontology design.

Although ontologies were originally motivated by the need for sharable and reusable knowledge bases, the reuse and sharing of ontologies themselves is still limited because the ontology users (and other designers) do not always share the same assumptions as the original designers. It is difficult for users to identify what the implicit assumptions were and to understand the key distinctions within the ontology. Some disagreements among people are superficial. For instance, one ontology may represent the color red as a relation while another represents it as a value. It is difficult to determine which disagreements reflect fundamentally different ontological commitments.

The article by Welty and Guarino provides guidelines on how to identify the hidden assumptions held by ontology designers and potential ontology users, and hence prevent the confusion that arises from inappropriate modeling choices.

Each of the disciplines within computer science considers a variety of theoretical and empirical techniques to evaluate potential solutions to research problems. However, there are currently few widely used techniques to evaluate and compare different ontologies. Within systems engineering, benchmark problems are used to compare hardware systems (VLSI chips) and software systems (protocols for load balancing networks); there is no such notion of benchmarks within ontological engineering. Within complexity theory, decision problems are classified into P and NP; algorithms are characterized by the order of their complexity. Within ontological engineering, there is no classification of problem domains and no characterization of ontologies to

Although ontologies were originally motivated by the need for sharable and reusable knowledge bases, the **reuse and sharing of ontologies themselves is still limited** because the ontology users (and other designers) do not always share the same assumptions as the original designers.

evaluate and compare their adequacy or performance.

Theoretical analysis of ontologies is possible. Given the specification of an ontology as a set of axioms in a logical language, we can characterize the models of the ontology's axioms and compare them to the intended models of the designer or user. In Welty and Guarino's article, we see the application of philosophical analysis to the evaluation of ontologies; they provide additional criteria to evaluate the commitments made by an ontology and to make the consequences of these commitments more explicit.

On the other hand, there are some issues that can only be resolved empirically. An ontology may have wonderful formal properties, but if it does not capture the intended semantics of the user's terminology, then the ontology has little practical utility. The article by Everett et al. considers this problem in the context of knowledge management. The ontology they develop must be adequate to correctly identify conceptually similar documents within a technical knowledge sharing system.

There are also needs for empirical research addressing the application of techniques to industrial problems. Do current design methodologies and techniques scale up for realistic problems? What is the tradeoff between generic ontologies and domain-specific ontologies that may be more immediately applicable to industrial problems, but may consequently be less reusable? Unfortunately, there is no notion of experiment or testing within ontological engineering. We would like to be able to use applications of ontologies as experimental test beds to compare and evaluate ontologies, and to serve as guides to frame the critical questions that must be posed. Ideally, an ontology test bed would include applications that require large-scale ontologies assembled from smaller, independently designed ontologies; new frontiers in ontology applications, such as knowledge management (see Holsapple and Joshi, Everett et al.), and the semantic Web (see Kim) would be excellent candidates.

The articles in this special section illustrate the interdisciplinary approaches within the ontology research community. Everett et al.'s study uses linguistic techniques to address the problems of knowledge management. Welty and Guarino's study uses the concepts and analytic methods from philosophy to evaluate an ontology, in the manner that normalization principles ensure the modularity of the database. Holsapple and Joshi's study illustrates the applicability of the management technique in addressing the challenge of consensus in ontology design. Kim's study uses history to draw predictions for the applicable areas of formal ontologies. This diversity and the academic tradition of the disciplines from which ontological engineering draws seem to testify to the conceptual complexity of ontology design. **C**

References

1. Berners-Lee, T., Hendler, J., and Lassila, O. The semantic Web. *Scientific American* 284, 5 (2001), 34–43.
2. Fernandez, M., Gomez-Perez, A., and Juristo, N. Methontology: From ontological art toward ontological engineering. In *Spring Symposium Series*, Stanford University. Stanford, CA., 1997.
3. Gruber, T. A translation approach to portable ontologies. *Knowledge 7*.

Michael Gruninger (gruning@cme.nist.gov) is a research scientist in the Institute for Systems Research at the University of Maryland, College Park.

Jintae Lee (jintae@colorado.edu) is an associate professor in the Systems Division of the Leeds School of Business at the University of Colorado, Boulder.