Ontological Aspects in Representing Mathematical Knowledge for Reasoning and Presentation Purposes

Helmut Horacek

Universität des Saarlandes
FB 14 Informatik, Postfach 1150, D-66041 Saarbrücken, Germany
email: horacek@cs.uni-sb.de

Abstract. Workshops on ontologies have been organized frequently over the past decade in connection with AI conferences. This is an indication that the topic is of broad interest to the AI community and also to other research communities, and that pursuing this topic seems to be a long-range term enterprise. One of the most severe problems in the field is the design of ontologies for multiple purposes. We address this issue by discussing the design of the knowledge base of our proof development system ΩMEGA, which combines reasoning facilities for handling mathematical proofs with presentation capabilities in natural language. Supporting the latter requires several enhancements to the primary ontology that is oriented on reasoning purposes. Important extensions are additional conceptual descriptions and markers for roles of defined items. Interfacing reasoning and presentation skills is not only beneficial for illustrating the results of formal inference systems; it is especially valuable for progress in tutorial systems.

1 Introduction

Workshops on ontologies have been organized frequently over the past decade in connection with AI conferences. This is an indication that the topic is of broad interest to the AI community and also to other research communities, and that pursuing this topic seems to be a long-range term enterprise. One of the most severe problems in the field is the design of ontologies for multiple purposes, which in our view, has rarely been addressed explicitly in the literature so far.

We address this issue by discussing the design of the knowledge base of our proof development system ΩMEGA [1], which combines reasoning facilities for handling mathematical proofs with associated presentation capabilities. Supporting the latter requires several enhancements to the primary ontology that is oriented on reasoning purposes. Important extensions are additional conceptual descriptions and markers for roles of defined items, plus an intermediate representation for handling the composition of natural language expressions in a flexible and linguistically motivated way.

This paper is organized as follows. First we characterize requirements on a domain ontology, based on the purposes of the contributing subsystems (here, reasoning and presentation). Then we sketch the present state of our knowledge base from the perspective of its ontology. Thereafter we illustrate extensions that meet the demands of presentation capabilities better than the present ontology does.

2 Task Purposes and Ontological Demands

For our domain of application, high quality representations are required, so that they can only be produced in a hand-crafted manner. Apart from that quality aspect, building ontologies for the domain of mathematics can in some sense be considered less difficult than a similar task for another domains:

- Vagueness does not play a role at all.
- There is a high degree of agreement about the domain concepts; discrepancies merely concern alternative representation variants, formats, and conventions.
- Although the domain as a whole is very large, it can reasonably well be broken down into subdomains of managable size.

In our application, the underlying ontology (or, better, system of ontologies) has to serve multiple purposes, that is, support domain reasoning and presentations in natural language. Depending on the given task purpose (reasoning or presentation), there exist different demands on the underlying ontology. For purposes of reasoning, the main requirement is an efficient access to definitions and axioms, thereby avoiding redundancy in storing that information to ease maintenance. For meeting this demand, an inheritance network proves to be the best mechanism.

Purposes of presentation impose some additional demands that cannot be met adequately by the inheritance network alone:

- Definitions of items need to be organized in a taxonomy to enable references to categories; in the inheritance network, this information is represented only implicitly. Moreover, items preferred for referring expression need to be marked. For example, the terms 'group' and 'semigroup' are more common than 'monoid', which identifies a similar algebraic structure. The more common terms should be preferred in descriptions even though this may require extended descriptions.
- Conceptual equivalences must be made explicit to avoid redundancy in presentations. There are, for example, various possibilities to define a group in algebraic terms, and switching between definitions might easily result in a strange rephrasing of an obvious equivalence. For reasoning purposes, the equivalence is established by an inference step or, in more complicated cases, by a subproof.
- Additional, highly special conceptual definitions that have no relevance for reasoning purposes may improve the presentation capabilities significantly. Typically, axioms that are expressed as compound or nested rules are good sources for building such definitions, which relate to subexpressions that appear in that axiom and are likely to be described as intermediate results in proofs.

Since reasoning is the primary purpose in the overall system, the knowledge base has been designed to meet the underlying demands best. Presentation capabilities are a typical 'adds-on' feature and not an absolutely required system facility, which constitutes a similar situation as in other systems. Hence, the knowledge base has to be enhanced appropriately to meet presentation demands, too.

3 The Domain Ontology

The enterprise of building an ontology is carried out within the system MBASE [5]. Apart from various measures to make storage and transactions highly efficient, MBASE offers access to a number of mathematical services through a system of mediators that take care of ontological differences between the mathematical service at hand and the view of MBASE (currently supported for three such services).

The statement of a mathematical theorem can depend on the availability of an eventually large set of definitions of mathematical concepts which, in turn, may themselves depend on other concepts. Moreover, previously proven theorems or lemmata may be reused within the context of a proof. Going beyond pure representation purposes, a formal reasoning system needs access to other forms of knowledge, including information about control knowledge for automated reasoners (theorem provers) and about (human-related) presentation knowledge.

In order to store and manipulate these kinds of information, MBASE distinguishes several categories of information objects, on which the structure of the underlying data base model is grounded:

- *Primary (mathematical) objects*, such as symbols, definitions, lemmata, theorems, and proofs.
- *Definitions* for associating meanings to symbols in terms of already defined ones.
- *Assertions*, which are logical sentences, including axioms, theorems, conjectures and lemmata, distinguished according to pragmatic or proof-theoretic status.
- *Human-oriented (technical) knowledge*, such as names for theorems, and specifications for notation and symbol handling.

Based on these categories, the data base model distinguishes the following primary data base objects:

- *Proofs*, as representations of evidence for the truth of assertions.
- *Proof objects*, encapsulating the actual proof objects in various formats, including formal, informal, and even natural language formats.
- *Examples*, due to their importance in mathematical practice.

In addition, the data base model contains some relations between objects of these kinds onto which inheritance is made. These include

- *Definition-entailmen*t, to relate defined symbols to others and, ultimately, to symbols marked as primitive.
- *Depends-on*/*Local-in*, which specify dependency and locality information for primary knowledge items. At the present stage of development, this relation is implemented for definitions and proofs, which make explicit the use of symbols/ lemmata in a definition or assertion, as well as for theories, which specifies the organization of mathematical subdomains and the associated inheritance of their properties.

For the task of presentation, most of these objects and relations are relevant, although we do not exploit the full potential at the current stage of development. At present, we make use of human-oriented objects to address technical presentation issues, and we combine definitions, assertions, and proofs along the relations among these objects, selected according to the requirements of a given presentation goal.

4 Methods for Increasing Presentation Capabilities

The techniques described so far mainly support the coordination of static knowledge originating from heterogeneous sources for problem solving purposes. For presentation purposes, there are two fundamental shortcomings of the present representation which severely limit presentation capablities:

- The connection between mathematical objects and natural language expressions is too simple, since it is restricted to one-to-one correspondences. This is meaningfully applicable to domain terms, but not to some relations and inference rules.
- The dynamic knowledge (e.,g., proofs, examples) is represented too coarse-grained and on a superficial level only, which limits variations for presenting it.

These shortcomings are as fundamental as they are deliberate, since the complexity of the design and development tasks for MBASE is high enough. For future extensions, these are good candidates for linking more linguistically oriented tools. Apart from these long-term issues, we intend to address the specific presentation problems with the currently available material, as outlined in section 2 when discussing the additional demands of presentation purposes, by some simpler measures.

In those places where knowledge about certain roles of definitions is required for presentation purposes, this is provided by some sort of marking and a disciplined domain model design. This measure applies to situation where the domain modeling part (e.g., for axioms) contains definitional equivalences (e.g., for two well-established representation versions of mathematical relations), which are connected by mere implications presently. Similarly, common and less common definitions are marked. Moreover, taxonomic links are introduced where needed.

Next, additional conceptual definitions are provided for meeting presentation demands, which may be required to express uses of complex axioms in inferences. For example, the terms 'meat eater' and 'plant eater' may be used in an elegant way to refer to partial results in involved inferencing about consequences of the eating habits of animals (as in the Steamroller problem [6]). For pure reasoning purposes, there is no need to introduce such definitions – they do not even help in representing intermediate result in the inference process more compactly, since the use of these presentation-oriented concepts for producing text requires some abstraction from the referential forms appearing in the intermediate inference results. The central axiom is

$$\forall x\,(\mathrm{ANIMAL}(x) \rightarrow (\forall y\,(\mathrm{PLANT}(y) \rightarrow \mathrm{EATS}(x,y)) \vee \forall y\,\exists z\,((\mathrm{PLANT}(z) \wedge \mathrm{ANIMAL}(y) \wedge \mathrm{EATS}(y,z) \wedge (y < x)) \rightarrow \mathrm{EATS}(x,y))))$$

on the basis of which additional definitions are given (as described in detail in [4]):

$\mathrm{PLANT\text{-}EATER}(x) \quad ::= \quad \forall y\,(\mathrm{PLANT}(y) \rightarrow \mathrm{EATS}(x,y))$ — "x is a plant eater"

$\mathrm{MEAT\text{-}EATER}(x) \quad ::= \quad \forall y\,\exists z\,((\mathrm{PLANT}(z) \wedge \mathrm{ANIMAL}(y) \wedge \mathrm{EATS}(y,z) \wedge (y < x)) \rightarrow \mathrm{EATS}(x,y)))$ — "x is a meat eater"

Finally, in order to increase the flexibility in which uses of mathematical concepts and associated inferences can be presented by natural language text, we intend to make use of bi-directional mappings between natural language and conceptual representations. The mapping operations provide a considerable degree of paraprasing capabilities. *Mapping schemata* express local correspondences across representation levels. They define equivalences of the information content associated with individual elements of the target representations – natural language and domain model – and corresponding constructs of the intermediate representation, which may consist of a chunk of elements for target representation elements associated with rich semantics. These operations have been originally designed for NL generation, enabling handling multiple languages [2], and their usefulness for building domain models and exploring alternative versions has been demonstrated in [3]. A methodological description and technical details can be found in [2].

5 Expected Benefits

There are several benefits of having a common database for a highly standardized domain such as mathematics:

- Primarily, inference services, such as the capabilities of provers can be strengthened and made better usable.
- Moreover, improved presentation capabilities contribute to the development of didactic tools. We will soon start a project on a dialog-oriented tutorial system that teaches proof skills, which heavily relies on data from MBASE, be it directly or indirectly through the logical reasoning systems incorporated.

References

1. Christoph Benzmüller, Lassaad Cheikhrouhou, Detlef Fehrer, Armin Fiedler, Xiaorong Huang, Manfred Kerber, Michael Kohlhase, Karsten Konrad, Erica Melis, Andreas Meier, Wolf Schaarschmidt, Jörg Siekmann, Volker Sorge: ΩMEGA: Towards a Mathematical Assistant. In Proc. of the 14th International Conference on Automated Deduction (CADE-97), pp. 252-255, Townsville, Australia, 1997.
2. Helmut Horacek: On Expressing Metonymic Relations in Multiple Languages. *Machine Translation* 11, pp. 109-158, 1996.
3. Helmut Horacek: An Approach to Building Domain Models Interactively. In Proc. of *NLDB-2001*, pp. 7-16, Madrid, Spain, 2001.
4. Helmut Horacek: Expressing References to Rules in Proof Presentations. In Proc. of *IJCAR-2001*, short paper session, Siena, Italy, 2001.
5. Michael Kohlhase, Andreas Franke: MBASE: Representing Knowledge and Context for the Integration of Mathematical Software Systems. Journal Symbolic Computation 11, pp. 1-37, 2000.
6. Mark Stickel: Schubert's Steamroller Problem: Formulations and Solutions. In *Journal of Automated Reasoning* 2(1), 1986.