Using Process Requirements as the Basis for the Creation and Evaluation of Process Ontologies for Enterprise Modeling

Michael Gruninger[1], Craig Schlenoff[2], Amy Knutilla[2], Steven Ray[2]

1. Department of Mechanical and Industrial Engineering, 4 Taddle Creek Road, University of Toronto, Toronto, Ontario M5S 3G9.

2. NIST, Bldg 220 Rm A127, Gaithersburg, MD 20899.

1.0 Introduction

A wide variety of applications deal with the manipulation and representation of collections of activities. Each of the applications serves a specific audience and need, and focuses on particular aspects of a process[1]. Nevertheless, much could be gained by sharing information among these applications. One of the primary obstacles to such integration is the lack of a generic representation of what is really the common underlying concept of process. The National Institute of Standards and Technology (NIST) Process Specification Language (PSL)[2] project's goal is

[1] For the purpose of this paper, we will define process as one or more activities that act over time to change the attributes of an object. In general, all processes have at least three main characteristics: a duration, an action, and resources.

[2] This project is funded by NIST's Systems Integration for Manufacturing Applications (SIMA) Program. Initiated in 1994 under the federal government's High Performance Computing and Communications effort, SIMA is addressing manufacturing systems integration problems through applications of information technologies and development of standards-based solutions. With technical activities in all of the NIST's laboratories covering a broad spectrum of engineering and manufacturing domains, SIMA is making

to create a process specification language to capture this underling concept of process which will facilitate the complete and correct exchange of process information among manufacturing applications[3] . The project has recently completed its study of determining the information requirements necessary to represent manufacturing processes. Although these requirements were expected to serve as a basis for the process specification language, it could just as easily serve as a foundation for a process ontology. An ontology provides a sharable representation of knowledge that minimizes ambiguity and maximizes understanding and precision in communication. The goals of this paper are: (1) to give an overview of the requirements necessary to represent manufacturing process and discuss different ways of categorizing these requirements, (2) to discuss the concept of an ontology and describe its structure, and (3) to describe how the PSL process requirements can help provide the foundation for a single, high-level ontology that describes the process domain as well as provide a basis for comparing existing process ontologies specific to individual enterprises.

2.0 Information Requirements for Representing Process

To create a common process representation, one must understand the information requirements that must be captured to sufficiently represent a process. During the first phase of the NIST PSL project, a broad cross section of applications which use manufacturing process information were examined to determine what process-related requirements need to be represented in each application. Applications studied included scheduling, process planning, simulation, project management, workflow, business process reengineering, and product realization process modeling. These applications were reviewed to ensure that all aspects of process found in a manufacturing environment could be included in the specification language effort.

These requirements were then categorized to make them easier to understand and to facilitate the representation analysis phase (the second phase of the PSL project). As an initial approach, the requirements were categorized by their pervasiveness in these manufacturing applications.

information interpretable among systems and people within and across networked enterprises.

[3] For more information about this project the reader is referred to [Schlenoff 1996] or http://www.nist.gov/psl/.

Four categories emerged: Core, Outer Core, Extensions, and Application. Each are described below:

- Core: the most basic, essential requirements inherent to all processes. While all processes contain core requirements, the core requirements provide the basis for representing only the simplest of processes. (e.g. time, resource, activity)

- Outer Core: the pervasive, but not essential, requirements for describing processes common to most applications.(e.g. temporal constraints, resource grouping, alternative tasks)

- Extensions: the groupings of related requirements, common to some, but not all, applications that together provide an added functionality (e.g. process yield in the real-time/dynamic extension). Six extensions have been defined: administrative/business, planning/analysis, real-time/dynamic, process intent, aggregate processes/resources, and stochastic/statistics.

- Application Specific: the requirements only relevant within specific applications (e.g., dynamic rescheduling for a production scheduling application).

This initial categorization does not necessarily imply that the final specification language will follow this organization. Preliminary thought has been given to alternative categorizations, which may or may not be mutually exclusive, including:

1. grouping of requirements depending on their relationship to resources, tasks, time (the three basic attributes of a process), or some combination of them

2. grouping of requirements as primitive concepts, their characteristics, and relationships

3. grouping of requirements with respect to their level of granularity as a function of the manufacturing life cycle. Some requirements may only be necessary later in the manufacturing life cycle (when detailed information is required) while others may only be relevant earlier in the life cycle.

Different types of categorization may have benefits to the different phases of the project. For example, the original categorization (Core, Outer Core, etc.) might be used for the introduction of the PSL language into the standardization process by specifying an application's level of conformance to a standard. The second grouping alternative (primitive concepts, etc.) can be used to rate existing process representations to determine how well they capture these requirements. Although many existing representations include constructs to represent the primitive concepts, far fewer have constructs to model concept characteristics or relationships between them.

3.0 Axiomatization of Ontologies

An ontology can be thought of as a sharable representation of knowledge that: (1) provides a shared terminology that various applications can jointly understand and use; (2) defines the meaning of each term (a.k.a. semantics) in a precise and unambiguous manner; and (3) implements the semantics in a set of axioms that allows one to automatically deduce the answer to many "common sense" questions. The process requirements discussed in Section 2 would serve as a strong foundation for the shared terminology and definitions for an ontology that describes the process domain.

An ontology is specified by a set of axioms in some formal language. However, this is not an arbitrary set of sentences - some axioms are conservative definitions, some are axiom schemas expressible using KIF ([Genesereth and Fikes 92]), and some are specifications of object classes and relations. The architecture in Figure 1 makes this structure explicit within the axioms in the specification of an ontology and integrates the object libraries of an ontology with the theories used to provide the semantics for the terminology in the object libraries[4]. The generic ontologies in the figure are examples of ontologies that have been defined in the TOVE ([Gruninger 96], [Gruninger and Fox 95], [Fox and Gruninger 94]) project.

Foundational Theories

A foundational theory is a set of distinguished predicates and functions together with some axiomatization. Distinguished predicates are those for which there are no definitions; the intended interpretations of these predicates is defined using the axioms in the foundational theories. Any terminology that does not have a definition is axiomatized in some foundational theory.

A major challenge facing ontology design is that the semantics for many ontologies are in people's heads; we need some framework for making it explicit. Any ideas that are implicit are a possible source of ambiguity and confusion. The foundational theories provide the means for formally specifying the semantics for the terminology of an ontology. Once we have specified a set of axioms in a foundational theory, they can be given an interpretation. Different interpretations can be given, but one of these will be the intended interpretation that guides the development of the axioms. The axiomatization allows a characterization of these interpretations. We can reason about the semantics of the terminology of the ontologies using the models of the axioms in the foundational theories.

Ontology Building Blocks

Once we have specified the axioms of the foundational theories and characterized these axioms, we can define classes of theories using the predicates and functions in the foundational theories. We call these classes of theories ontology building blocks. We explicitly identify building blocks to assist axiomatization; they will be used to provide definitions for the classes and relations of the generic ontologies. All sentences in the generic ontologies belong to the classes of theories defined in the ontology building blocks.

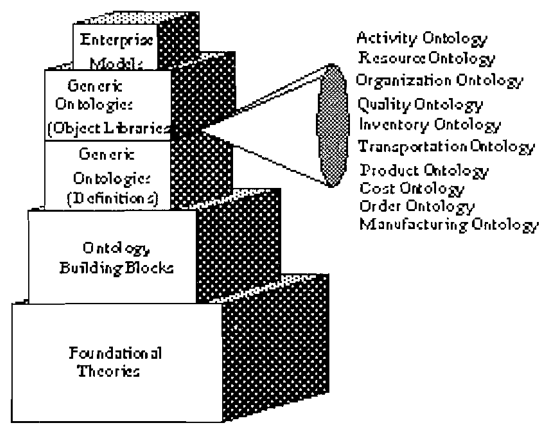

Figure 1 - The structure of ontologies

Generic Ontologies

At the next level up in the architecture, we have the object classes and relations of the ontology structured as a set of object libraries. All of these classes and relations have definitions expressed using the axiomatization of some set of underlying foundational logical theories. These definitions are conservative with respect to the foundational theories; that is, every sentence that we can prove using the definitions of the ontology and axioms of the foundational theories we can prove using the axioms of the foundational theories alone. Typically, the object classes in the libraries are used to assist in modeling, while the definitions are used by an underlying reasoning system.

---

[4] Implicit in the architecture is the fact that there must be some formal language, such as first-order logic, which is used to express the axioms and definitions of the ontology.

4.0 A Methodology for the Creation and Evaluation of Ontologies

There are two ways in which the previous concepts (process requirements and ontologies) can be integrated. First, the complete set of requirements necessary for representing process (referenced in Section 2) could serve as a strong foundation for the creation of a proposed process ontology. The goal of this would be the construction of a single standardized process ontology satisfying the PSL requirements which could serve as a basis for future process ontologies. Although the categorizations of the process requirements mentioned in Section 2 were originally done for other purposes, they could easily be used to facilitate the creation of the ontology by providing the basis for the structuring the information.

Second, the process requirements can serve as a mechanism to evaluate existing process ontologies for completeness. We can use the architecture of Figure 1 to support the evaluation of an ontology with respect to a set of requirements. A formal evaluation of an ontology can only be done using the models of the foundational theories of the ontology. A requirement is formally expressed in terms of some set of intended interpretations. Intuitively, a requirement is satisfied by an ontology if the models of the axioms and definitions in the ontology are equivalent to the intended interpretations of the requirements. In this way, the design and evaluation of an ontology against a set of requirements proceeds in three directions:

- **Adopting Object Libraries:** In this case, the definitions of relations, functions, and classes in the ontology satisfy the requirements.

- **Extending Object Libraries:** If there do not exist relations, functions, or classes in the ontology which satisfy the requirements, then we can provide definitions for these terms using the existing foundational theories of the ontology.

- **Extending Foundational Theories:** The foundational theories may be too weak to provide definitions for terms in the requirements. Semantically, this means that the intended interpretation of the requirement cannot be captured using the models of the foundational theories. New distinguished predicates must therefore be proposed, and an axiomatization of these relations must be given and evaluated. Once the new foundational theories have been defined, it may be necessary to provide new definitions for the ontology in order to satisfy the requirements.

5.0 Conclusion

Within the PSL project, a study has been completed to determine the information requirements necessary to represent manufacturing process. These requirements will help to ensure the completeness and robustness of an ensuing process specification language. Although not originally gathered for this purpose, these requirements would have applicability as the foundation for the creation and evaluation of process ontologies. By having a comprehensive set of process requirements, existing ontologies can be evaluated for completeness by determining which requirements each supports. These requirements can also serve as a foundation for the creation of a new, standardized process ontology on which all future process ontologies would be built. Additionally, the various requirements categorizations discussed above can help to define the structure of the ensuing ontology.

6.0 Bibliography

Fox, M.S. and Gruninger, M., (1994), "Ontologies for Enterprise Integration", Proceedings of the 2nd Conference on Cooperative Information Systems, Toronto, Ontario.

Gensereth M., Fikes R. 1992. Knowledge Interchange Format. Stanford Logic Report Logic-92-1, Stanford Univ. http://logic.stanford.edu/kif/kif.html

Gruninger, M (1996), "Designing and Evaluating Generic Ontologies", Workshop on Ontological Engineering, ECAI-95, Budapest.

Gruninger, M., and Fox, M.S. (1995), "Methodology for the Design and Evaluation of Ontologies", Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal.

Schlenoff, C., Knutilla, A., Ray, S., "Unified Process Specification Language: Requirements for Modeling Process," NIST Interagency Report 5910, Gaithersburg, MD, September 1996.