An Evaluation of Ontology Exchange Languages for Bioinformatics

Robin McEntire[1], Peter Karp[2], Neil Abernethy[3,] David Benton[1], Gregg Helt[8], Matt DeJongh[6], Robert Kent[7], Anthony Kosky[9], Suzanna Lewis[8], Dan Hodnett[6], Eric Neumann[10], Frank Olken[4], Dhiraj Pathak[1], Peter Tarczy-Hornoch[5], Luca Toldo[11], Thodoros Topaloglou[9]

[1]SmithKline Beecham Pharmaceuticals, 709 Swedeland Rd, King of Prussia, PA 19406
{Robin_A_McEntire,W_David_Benton,Dhiraj_K_Pathak}@sbphrd.com
[2]SRI International, 333 Ravenswood Ave, Menlo Park, CA 94025, pkarp@ai.sri.com
[3]InGenuity, 500 Ellis St, Mountain View, CA 94043, nfs@ingsys.com
[4]Lawrence Berkeley Livermore Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, olken@lbl.gov
[5]University of Washington, Department of Pediatrics, Box 356320, Seattle, Washington 98195-6230, pth@u.washington.edu
[6]NetGenics, 500 W Wilson Bridge Rd, #100, Worthington, OH 43085, {mdejongh,dhodnett}@netgenics.com
[7]Ontologos, http://www.ontologos.org, rkent@ontologos.com
[8]University of California Berkeley, Berkeley Drosophila Genome Project, Rubin Lab, LSA Room 539, Berkeley, 94720-3200, {suzi,gregg}@fruitfly.berkeley.edu
[9]GeneLogic, 708 Quince Orchard Road, Gaithersburg, Maryland 20878, {anthony,thodoros}@genelogic.com
[10]3rd Millenium, Cambridge, Massachusetts, eneumann@3rdmill.com
[11]Merck KgaA, Frankfurter Street 250, Darmstadt, Germany, luca.toldo@merck.de

Abstract

Ontologies are specifications of the concepts in a given field, and of the relationships among those concepts. The development of ontologies for molecular-biology information and the sharing of those ontologies within the bioinformatics community are central problems in bioinformatics. If the bioinformatics community is to share ontologies effectively, ontologies must be exchanged in a form that uses standardized syntax and semantics. This paper reports on an effort among the authors to evaluate alternative ontology-exchange languages, and to recommend one or more languages for use within the larger [1]bioinformatics community. The study selected a set of candidate languages, and defined a set of capabilities that the ideal ontology-exchange language should satisfy. The study scored the languages according to the degree to which they satisfied each capability. In addition, the authors performed several ontology-exchange experiments with the two languages that received the highest scores: OML and Ontolingua. The result of those experiments, and the main conclusion of this study, was that the frame-based semantic model of Ontolingua is preferable to the conceptual graph model of OML, but that the XML-based syntax of OML is preferable to the Lisp-based syntax of Ontolingua.

Introduction

Ontologies, as specifications of the concepts in a given field, and of the relationships among those concepts, provide insight into the nature of information produced by that field and are an essential ingredient for any attempts to arrive at a shared understanding of concepts in a field. Thus the development of ontologies for molecular-biology information and the sharing of those ontologies within the bioinformatics community are central problems in bioinformatics.

If the bioinformatics community is to share ontologies effectively, the ontologies must be exchanged in some standardized form, such as using a file with a well-defined syntax and semantics. Exchange of bioinformatics ontologies will be simplified if the community can agree on a relatively small number of such exchange forms --- ideally, on one form.

This paper reports on an effort among the authors to evaluate a number of alternative ontology-exchange languages, and to recommend one or more languages for use within the larger bioinformatics community. The evaluation effort involved three separate meetings in 1998 and 1999 by the authors, as well as experiments with the proposed ontology languages. In phase I of the evaluation, the authors selected a set of candidate languages, and a set of capabilities that the ideal ontology-exchange language should satisfy.

The authors then scored the languages according to the degree to which they provided each capability. In phase II of the evaluation, the authors performed several ontology-exchange experiments with the two languages that rated the highest during phase I, which were OML and Ontolingua.

This paper describes the evaluation process and its results in more detail.

A web site maintained by the Bio-Ontologies Consortium can be found at http://www-smi.stanford.edu/projects/bio-ontology/.

Motivations

Ontology development is important because every biological database employs an ontology, either implicitly or explicitly, to model its data. The more *fine-grained* the ontology, the more precisely the database will be able to model the nuances of the data that it tries to capture. A *coarse-grained* ontology will model only superficial aspects of the data, and therefore may not capture data elements that are important for some problem-solving task. For example, a genome-sequence database that fails to record which genetic code is used to encode a given DNA sequence does not provide the information that users of the database will need to reliably translate each DNA sequence into the corresponding protein sequence.

A *semantically malformed* ontology is one that incorrectly models the semantics of its application domain, and therefore yields a database whose structure corrupts or restricts the information that it is intended to hold. For example, a metabolic database that defines a one-to-one relationship between enzymes and the reactions they catalyze cannot reliably model the fact that a bifunctional enzyme catalyzes two separate reactions.

Ontology sharing is important for several reasons. First, ontology development is time consuming. Different bioinformatics groups who wish to develop ontologies for the same types of biological information will often arrive at a solution faster by adopting an existing ontology than by developing a new ontology *de novo*. For example, a group that wishes to define an ontology for microarray gene-expression data will almost certainly accomplish this task more quickly by consulting one or more existing microarray ontologies.

Second, if different bioinformatics databases that cover the same types of data (e.g., protein sequences) employ the same ontology, they simplify the problem of database integration, that is, of processing queries across multiple biological databases. Different ontologies for the same types of data produce a semantic mismatch that complicates the multidatabase query problem.

Third, bioinformatics databases must make their schemas available to their user communities if the users are to have a full understanding of the semantics of these databases. However, relational schemas are inadequate for the representation and exchange of biological information.

Fourth, ontology sharing is important because ontologies themselves constitute a form of biological knowledge that is quite valuable when shared within the bioinformatics community. For example, the taxonomy of enzymatic reactions developed by the Enzyme Commission (Webb 1992) and the taxonomy of gene function developed by (Riley 1993) are valuable bioinformatics ontologies.

Fifth, differences between ontologies purporting to represent the same biological process may lead to important insights into ways of improving those representations, and/or new insights into the underlying biology.

Terminology

Ontologies are defined in the literature in various ways with varying degrees of formality. One prevailing definition of an ontology is a specification of a conceptualization that is designed for reuse across multiple applications. By conceptualization, we mean a set of concepts, relations, objects, and constraints that define some domain of interest.

One can argue at length about what is and is not an ontology (Gruber 1993)(Guarino 1995). Our view is that ontologies exist at several levels of complexity:

- A *controlled vocabulary* is an ontology that simply lists a set of terms.
- A *taxonomy* is a set of terms that are arranged into a generalization-specialization hierarchy. A taxonomy does not define attributes of these terms, nor does it define relationships between the terms.
- An *object-oriented database schema* defines a hierarchy of classes, and attributes and relationships of those classes.
- A *knowledge-representation system* based on first-order logic can express all of the preceding relationships, as well as negation and disjunction.

The GeneClinics experiment (see www.geneclinics.org) illustrates this range of complexity among different ontologies. One of the first steps of the experiment was to augment the object-oriented schema with a richer set of capabilities including disjunction, role restriction, and other constraints. In the GeneClinics object database much of this information was in fact represented in the Java software interacting with the database but was hidden from the end user.

Candidate Languages

Candidate ontology-exchange languages were evaluated by the authors. We discuss the reasons each language was selected for consideration as a bioinformatics ontology exchange language, we list the developers of each language and the design considerations for each language, and we provide references for each language.

Ontolingua. The Ontolingua language was developed by a group at Stanford University for the exchange of ontologies, and was originally funded by the DARPA Knowledge Sharing Effort. Ontolingua is one of the most significant efforts to come out of the knowledge representation community and is based on the Knowledge Interchange Format (KIF), a language specifically built for the sharing of knowledge among different knowledge representation systems. The authors believed that any evaluation of languages for the exchange of ontologies must include this project. The semantics of Ontolingua are based on the frame knowledge representation systems developed by knowledge-representation researchers (Fikes and Kehler 1985)(Karp 1992).

CycL. Cyc is perhaps the best-known of the knowledge representation systems and is significant in its scope and its longevity. Cyc was developed by Doug Lenat at MCC but has since spun off as a commercial entity, Cycorp. The underlying representation language for Cyc is called CycL, which derives from first-order predicate calculus but with extensions for additional expressivity. Currently, Cyc is one of the most significant commercial products, if not the most significant, in the marketplace. For this reason, as well as its significance within the knowledge representation community and its rich expressive abilities, it was selected for evaluation. (Lenat and Guha 1990)(CycL 2000)

OML/CKML. Ontology Markup Language/Conceptual Knowledge Markup Language (OML/CKML) is a relatively new effort, from Washington State University, that is attempting to base a system for the expression of ontologies on an XML-based syntax. The OML effort was begun in the 1990s and, though relatively young and untested, the authors believed it to have significant representational power. This representational power combined with the interoperable nature of an XML-based language was believed to be a combination worth investigating. In addition, since OML/CKML is currently under development there is a potential for co-development to allow the bioinformatics community to influence features and expressive power of the language. There is, though, a possible disadvantage in that the language may evolve in ways that are not to the advantage of the community or that it may not be stable or standardized. (Kent 1999)

OPM. OPM was interesting to the authors as a candidate language for exchange of ontologies because of the significance of the OPM system, a product from GeneLogic used by Pharmaceutical, BioTech, and academic organizations. OPM is an object-oriented data model used to describe single and multi-database schemas and queries. As a product, it is used for the rapid development of databases, database query interfaces and integration of multiple data sources. (Topaloglou, Kosky and Markowitz 1999)

XML/RDF. Extensible Markup Language/ Resource Description Format (XML/RDF) was developed by the World Wide Web Consortium (W3C). The current standard for the XML Schema Language is controlled by the XML Schema Working Group of the W3C. RDF is intended to encode metadata concerning web documents. XML/RDF was investigated as a part of the evaluation effort because of the significance of the web and web-based applications. It is clear that the web is rapidly becoming the primary method for the exchange of information and data, and that XML is currently the leading candidate for a generic language for the exchange of semistructured objects. XML/RDF as is, without a higher-level formalism that encompasses the expressivity present in frame-based languages, does not go far enough to allow the kind of modeling needed in the bioinformatics community. (St Laurent 1998)(W3C 2000)

UML. The Unified Modeling Language (UML) provides a set of notational conventions that can be used by software application designers/developers to model their software systems. UML was developed by Rational Software and is currently backed by Rational, Microsoft, and the Object Management Group (OMG). UML was selected for evaluation because it is another widely used system for the representation of objects and their relationships. (Rumbaugh, Jacobson and Booch 1998, Object Management Group 1997)

OKBC. Open Knowledge Base Connectivity (OKBC) is an API for accessing and modifying multiple, heterogeneous knowledge bases. OKBC is not actually an ontology exchange language – it is a programmatic API. This group considered it because its knowledge model was designed to capture ontologies. The OKBC effort began as a part of the recent DARPA High Performance Knowledge Base (HPKB) program, and is the successor of Generic Frame Protocol (GFP), a frame representation system developed at the Artificial Intelligence Center at SRI International. OKBC was created because it provides a uniform model that can be understood across knowledge representation systems. The work on OKBC is currently being overseen by a working group led by Richard Fikes at Stanford. Voting members in this group are ISI, Stanford KSL, SRI, Cycorp, SAIC, and Teknowledge. (Chaudhri, et al 1998) (OKBC 2000)

ASN.1. ASN.1 was included in this evaluation because of its historical significance as an early language for the exchange of datatypes and simple objects. The ASN.1 standard was developed as part of the OSI networking stack. It has been, and still is being used in bioinformatics applications from the National Center for Biotechnology Information. ASN.1 was also used in conjunction with the Unified Medical Language System (UMLS) project at the National Library of Medicine (NLM). However, production of ASN.1 encodings of the UMLS has been discontinued because of low demand for ASN.1 by UMLS users. (Larmouth 1999)

ODL. The Object Definition Language (ODL) is a relatively new standard from the Object Database Management Group (ODMG) and was developed in the early 1990s. ODL was selected for evaluation because it is a de facto standard for a common representation of objects for object-oriented databases and programming languages, and so has the potential to be supported throughout the industry. The ODMG member companies include almost all organizations in the ODBMS/ODM industry. The

ODMG is very closely aligned with the OMG. (Cattell, et al 2000)

Evaluation

Initial Evaluation

The evaluation process began with the selection of known languages for expressing ontologies. Our selection process relied on an informal review of current literature and prior knowledge of participants, but, we believe, covers the most viable candidate languages for the exchange of ontologies. The languages, once selected, were then divided among the authors for evaluation.

To evaluate the languages in a consistent fashion, the authors arrived at a set of questions for which each candidate language would be evaluated. The full set of questions distributed to members of the working group can be found in Appendix A. The questions were divided into the following five major categories:

1. Language Support and Standardization: general questions about the depth of support for the language, including technical support and relationship with standards efforts
2. Data model/capabilities: richness of the expressive capabilities of the language
3. Performance: rather than expressiveness of the language, some notion of what might be expected in terms of performance if a given language were used
4. Other Issues: pragmatics, such as current use of the language and representation of, or connectivity to, non-ontology sources

The final judgement of the authors for the initial evaluation phase was guided by a matrix of the aspects of an exchange language that were considered key to its use by members of the Bio-Ontology Consortium (http://www-smi.stanford.edu/projects/bio-ontology/) and other groups who may want to build ontologies in the area of molecular biology. Tables 1 and 2 show results of the evaluation of candidate languages. In addition, Table 3, below, was used by the authors to evaluate the initial candidate languages after evaluation of each question was complete. This table show the desired attributes of an exchange language, and how each language was rated along those aspects.

The authors decided that no single language stood out as the only appropriate candidate for recommendation as a language for the exchange of molecular biology ontologies. It was clear that representational expressiveness was not adequate in some languages, and so they were eliminated from consideration. For example, some languages were

Property	**ASN.1**	**ODL**	**Ontolingua**	**OML/ CKML**	**OPM**	**XML/ RDF**	**UML**
Formal Syntax?	Yes	No	Yes	Yes	Yes	Yes	Yes
Translators	No	Yes	Loom, IDL, KIF, CLIPS, etc	No	Relational, ASN.1, XML, HTML, ER	No	No
Software Tools	Parsers	Parsers	WWW browsers, editors,comparison tools	No	Yes	XML toolkits	Rational Rose
Support		yes	WWW docs, FAQs, tutorial,support staff	WWW grammars, WWW examples	Docs, training, tutorials	WWW sites, mailing lists, books	Formal courses, books, tutorials
Controlling Org	ISO	ODMG	Stanford	WSU	GeneLogic Inc	W3C	OMG
Stability	Stable	Stable	Stable	Evolving	Stable	Evolving	Stable
Users	Yes	OO Vendors	WWW users	Intel apps	Yes, Bix and others	WWW developers	many parts of industry
Bioinfo Users	NCBI	Yes	SB, Stanford RiboWeb	Yes	GDB, MaizeDB, SB, PE Biosystems, other Pharma, Biotech	No	SB, (probably other pharmas)
Developers		OO Vendors	Stanford	WSU	GeneLogic	many, many	Rational Rose

Table 1: Evaluation Matrix 1, answers to general questions.

Property	**ASN.1**	**ODL**	**Onto**	**OML/ CKML**	**OPM**	**XML/ RDF**	**UML**
Negation	No	No	Yes	Yes	No	No	No
Conjunction	No	No	Yes	Yes	Yes	No	No
Disjunction	No	No	Yes	Yes	Yes	No	No
Relations	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Multiple Inheritance	No	Yes	Yes	Yes	Yes	Yes	No
Inverses	No	Yes	Yes	Yes	Yes	No	No
Multi-valued slots	Yes	Yes	Yes	Yes	Yes	Yes	No
Multiple collection types	Yes	Yes	No	No	Yes	Yes	No
Number restrictions	No	No	Yes	Yes	Yes	No	Yes
Slot hierarchies	No	No	Yes	Yes	No	No	No
Facets	No	No	Yes	Yes	Yes	No	No
Default Values	No	No	No	Yes	Yes	Yes	No
Other slot constraints	No	No	Yes	Yes	No	No	No
Primitive Datatypes	Standard	Standard	Standard	Standard	Standard	None	N/A
Data Model	Object w/o inheritance	Object	Object and Logic	Object and Logic	Object	SemiStructured data	Object
Instances and classes	No	No	Yes	Yes	No	Yes	No

Table 2: Evaluation Matrix 2, representational expressiveness. See Appendix B for explanation of properties.

unable to encode ground facts (instance objects). Also, some languages were in part or in whole proprietary, or had a significant cost associated with them. This was considered prohibitive to the successful adoption and use of the languages, and so these languages were also eliminated . It was decided that two languages, Ontolingua and OML/CKML, provided enough expressivity to warrant a more in-depth evaluation.

Evaluation Part II: OML and Ontolingua

The second phase of the evaluation process focused on the two candidate languages that were deemed most interesting from the initial evaluation: Ontolingua and OML/CKML.

The authors decided that it would be useful to create a small model in each language in order to judge utility and representational richness. A set of experiments was developed to perform this detailed evaluation. The three experiments are outlined below. Details of these experiments and their results can be found at the web site for the Bio-Ontologies Consortium.

Experiment 1: OML Representation of the EcoCyc Gene Ontology. Dr. Peter Karp's group at Pangea Systems performed an experiment to better understand the OML language by translating the EcoCyc gene ontology into OML. The gene ontology, a taxonomy of 150 classes that classify microbial genes according to their functions, was developed by Dr. Monica Riley as part of the EcoCyc project (Riley 1993)(Karp et al 1999). The ontology is relatively simple in terms of the representational constructs required to encode it.

Within EcoCyc, the ontology can be accessed at http://ecocyc.pangeasystems.com:1555/class-subs?object=Genes. The OML encoding of the ontology can be accessed at http://www.ai.sri.com/~pkarp/xol/omlgenes.txt.

Results: Our findings were that OML was able to capture most aspects of the gene ontology. However, we identified what we consider to be limitations of OML during the course of this experiment.

1. Several aspects of the terminology used in the tags in OML files are not intuitive, and are not consistent with the terminology used in the more mainstream ontology community. This terminology will interfere with the acceptance and understanding of the language in the

bioinformatics community. We suggested that OML could allow several alternatives for each tag so that the language would be accepted by different communities that use different terminology.

	Ontolingua	XML/RDF	OML	OKBC	OPM	CycL	UML/XMI
classes & instances	+	+	+	+	-	+	+
multiple inheritance	+	+	+	+	+	+	+
constraints	++	-	++	+	+	+	+
defaults	+	+	+	+	+	+	
expressive power	+++	+	+++	++	++	+++	+
tools available*	lisp *(AF)*	Java		lisp, Java, C	Java, C++	lisp, Java, C *(AF)*	
stability	+	-	+	+	+	+	-
support	+	++	+	+	+	+	-
translators	++	+	?	+	+	KIF. Loom	-
many applications	+	+	+	+	+	+	-
open language	+	+	+	+	+	+	+
simplicity: human	good	low	low		good	good	low
simplicity: formal	good	good	good		good	good	
open to collaboration	+	++	++	+			
STATUS		out		out	out	out	out

Table 3: Final Results of Evaluation (A plus sign, "+", indicates a positive. More than one plus sign indicates more significant positives. The minus sign, "-", indicates a negative evaluation of a criterion. AF indicates that the language/product is free to academic organizations.)

2. The OML definitions are not modular in the sense that the OML definition of a given Class is spread out into several parts of the file, making OML files less human readable.
3. OML has limitations in expressive power:
   a) It cannot express facets directly (attributes of attributes), but R. Kent suggested that N-ary relations can be used to express facets.
   b) It cannot express annotations.
   c) It cannot handle multiple collection types -- sets only.
   d) It cannot express cardinality or numeric-range constraints.

Experiment 2: Ontolingua Representation of the EcoCyc Gene Ontology. Dr. Karp's group represented the same gene ontology using Ontolingua.

Expressing the gene ontology in Ontolingua was straightforward. The Ontolingua encoding of the ontology can be found at http://www.ai.sri.com/~pkarp/xol/ontogenes.txt.

Experiment 3: Representation of GeneClinics Data Model as an Ontology. Peter Tarczy-Hornoch in collaboration with Luca Toldo and Robert Kent performed an experiment with the general goal of using the existing GeneClinics OODB model as the basis for an ontology to assess OML/CKML and Ontolingua for ontology creation/exchange. The specific goal was to develop a small representative ontology in both Ontolingua and OML/CKML that represents key clinical and molecular entities and their linkages.

1. Peter Tarczy-Hornoch in collaboration with Luca Toldo and Robert Kent performed an experiment with the general goal of using the existing GeneClinics OODB model as the basis for an ontology to assess OML/CKML and Ontolingua for ontology creation/exchange. The specific goal was to develop a small representative ontology in both Ontolingua and OML/CKML that represents key clinical and molecular entities and their linkages. The design of the experiment was as follows.
2. The experiment was conducted using E-mail among the three investigators.
3. The GeneClinics investigator developed a 5-page document outlining a subset of the high-level (coarse-grain) GeneClinics OODB model. The scope of this model was to represent key clinical entities (clinical diagnoses, tests), key molecular entities (genes, loci, products, alleles, mutations), and their inter-relationships (causality maps to diagnoses, clinical tests for molecular entities).
4. The whole group clarified points including disjunctions, restrictions, and other constraints not in the OODB model.
5. The developer of OML/CKML (one of the three participants) implemented and refined the OML/CKML ontology.
6. A specific instance (Charcot Marie Tooth type 1A) was represented.

7. The same ontology was represented in parallel in Ontolingua.
8. A specific instance (CMT 1A) was represented in Ontolingua.
9. The OML/CKML and Ontolingua experiences were compared and contrasted.
10. A few very granular elements were implemented (chosen to "stress" each language and compare robustness).

Results.

1. The underlying paradigms (data models) of Ontolingua and OML/CKML are subtly different – frames based vs. conceptual graph based (formal concept analysis, information flow theory). Those not familiar with either paradigm will need to learn it.
2. Ontolingua concepts are mapped more closely to object databases and object oriented programming paradigms, and thus might be easier for the typical bioinformaticist to learn.
3. The two languages have a minor difference in namespaces -- Ontolingua requires the object name to be a unique identifier.
4. OML/CKML's XML syntax makes it easier to learn than Ontolingua with its LISP syntax.
5. Neither language has the type of documentation of its syntax and semantics that would be needed for a tutorial for a bioinformaticist. Ideally, the tutorial/documentation would include a formal representation of syntax with modified BNF format, as well as selected examples drawn from biology, building in complexity. Examples: representation of a biological entity like a protein, representing the concept of a sequence of DNA codes for that protein, expressing that proteins have one or more of a following list of functions, and so forth.
6. Both languages are very expressive – Ontolingua's expressivity is easier to see in both LISP and in the Ontolingua ontology-development tool because it is exposed even in simple examples. The expressivity of OML/CKML is rich but harder to determine since (a) it is not apparent in simpler examples, (b) things like local theories and other concepts are powerful but harder to understand (the documentation is in the conceptual graph paradigm), and (c) the documentation and specification are both evolving. In principle the OML/CKML conceptual graph model may be richer and more expressive than the frame model; an exact comparison of the two models would be useful.
7. Both languages were able to handle the needs of the GeneClinics sample ontology (not a complex ontology).
8. The conceptual-graph paradigm is dense but very powerful (see document Designator-Facet.doc for examples).
9. Though not per se an attribute of the languages themselves, it is important to note that software tools and applications, such as editors, browsers, parsers, translators, and query systems, exist for Ontolingua but not for OML/CKML, making Ontolingua a more accessible language for ontology development, as opposed to ontology exchange. That is, OML/CKML is "an uninstantiated formalism."
10. The availability of the developer of OML/CKML (R. Kent) for collaboration on this project was immensely helpful.

Conclusions.

1. The expressive power of the two languages is similar and more than adequate for the purposes of expressing a part of the GeneClinics data model as an ontology. OML/CKML is, however, theoretically more powerful because it is based on the conceptual-graph methodology. For a specific example of the expressive capabilities of each language, please review the examples in Appendix C.
2. The Ontolingua frames semantics/paradigm may be easier to learn since it is less of a leap from the object database and object programming paradigms. However, the LISP syntax of Ontolingua could present a challenge to many bioinformaticians and the XML syntax of OML/CKML is likely to be more intuitive. Ideally, an ontology exchange language would have an easy-to-learn basic semantics and syntax (like XML) but be very expressive (like OML/CKML and Ontolingua). Neither language as it stands quite achieves this ideal, though a more frame-based version of OML/CKML or an XML encoding of Ontolingua might come closer.
3. For the general bioinformatics community (not versed in ontology representation), it might be helpful to create documentation and tutorials that use biological examples.

Evaluation Part 3; Recommendations

At its last meeting, the BioOntology Core Group reached the following conclusions and recommendations.

The group reached two major decisions for the selection of a language for the exchange of ontologies for molecular biology:

1. A traditional frame-based approach for representation of biological entities is sufficient for current needs since many databases of biological information are in relational or flat file format. Frame-based systems provide natural mappings onto relational schemas. In addition, frame-based systems have been in use for a significant period of time and are, in general, stable representation systems. Among frame-based systems, Ontolingua is clearly one of the most prominent and has had extensive use for many years.
2. XML has tremendous momentum with significant interest from commercial organizations and a serious standardization effort. We anticipate that XML-based tools and web servers supporting XML will be available soon.

The belief of the group was that the language that the bioinformatics community needs for the exchange of ontologies should have frame-based semantics with an XML expression. However, the group also believed that we do not yet have such a language because Ontolingua is frame-based but without an XML expression and OML has an XML expression but is based on conceptual graphs instead of frames.

At the meeting Peter Karp presented preliminary work that he and Vinay Chaudhri, from SRI, had done on producing an XML expression based on the OKBC knowledge model, which in turn is very closely related to Ontolingua (the Ontolingua developers were also involved in the development of OKBC). This new language is the XML Ontology Language (XOL) (Karp and Chaurdhri 1999).

The consensus of the group was that we recommend the use of a frame-based language with an XML syntax for the exchange of ontologies, and, to that end, the group requested that Karp and Chaudhri complete their work on the XML expression of Ontolingua, so that the group could complete its evaluation of exchange languages.

Summary

Over the last two decades, the knowledge representation and object-oriented database communities have developed languages that may be used for the expression of semantic database models. These languages share many elements in common, and are exemplified by the frame-knowledge representation systems used in the knowledge representation community. Frame systems have been used in many different bioinformatics projects, and the authors believe that frame systems provide the necessary representational constructs to model ontologies for molecular biology. Furthermore, frame systems have a significant history of use, and provide a stable representational paradigm.

The authors also believe that the explosion of the web and the languages associated with it simply cannot be ignored. Acceptance of an exchange language that is expressed in a Lisp syntax will be limited within the bioinformatics community, even though the underlying representational system may be identical to that expressed in a web-based language. For this reason the authors believe that an XML-based syntax must be used for a bioinformatics ontology exchange language to increase the likelihood that the language will see widespread acceptance.

Future Directions

The results of this evaluation suggest two directions for future work: development of an XML expression for the Ontolingua model, or adapting OML/CKML to include a frame-based semantic model.

The authors support the use of a frame-based exchange language using an XML syntax. Several researchers on the evaluation team are currently developing a specification of XML expression of Ontolingua using OKBC. Other researchers on the team are pursuing a frame-based version of OML.

The exchange language evaluation team will meet again to consider the question of whether either, or both, of these efforts provides an acceptable exchange language meeting the group's requirements.

Appendix A: Evaluation Criteria

The following questions were asked about each candidate language during the Phase I evaluation process.

Language Support and Standardization

Is a formal specification of the syntax of the ontology language available? How complex is its syntax? Please present that formal specification of the language at the meeting.

What parsers are available for the language? What translators are available to convert between language L and other ontology-description languages? How complete are those translators?

What other software is available that operates on the language, such as for web-based publishing of ontologies or browsing/editing of ontologies?

What support (documentation, training, tutorials, e-mail) is available for the language?

Does it have any development/usage standards? Who controls this standard?

Does a stable release of the language exist (i.e., one that will not fundamentally change in 6 months)?

Data Model/Capabilities

What assumptions does the language make about the ontology to be represented?

Which of the following does the language support:

negation
conjunction
disjunction
recursion
relations
multiple inheritance
multi-valued slots
number restrictions on roles
role hierarchies
transitive roles
axioms
template/default values
method slots (calculated values?)
constraints

If the language supports constraints, how rich is the constraint language? Is the constraint language formally defined?

What are the primitive data types in the language?

What database data model(s) does the language support?

Does the language encode instances as well as classes (data as well as schema)?

Querying

[These questions are more about ontology tools (editors, viewers, ...) than language.]

What tools exist for querying an ontology expressed in this language?

How are queries expressed?

Which of the following queries can be expressed in the query language:

What are the parents of concept C?

What are the children of concept C?

What could I say about concept C (e.g., what roles are legally applicable to C)?

Is concept C satisfiable?

What role-fillers can a role have for a concept C?

What English expression does C have?

Is C a kind of D?

What is the least common parent of C and D?

What is the greatest common child of C and D?

Are C and D equivalent?

Can queries be translated/compiled into a standard programming/query language?

Performance

[These questions are more about ontology tools (editors, viewers, ...) than language.]

Are there any limits (or the limits of available translators/parsers) in the size of the ontology, the length of names/values, etc. (theoretical or practical).

What is the overhead (bytes) for a language parser? interpreter?

For resources which depend on an information service for support (such as Ontolingua), does the service have the capacity to support all of the users of the technology?

Other Issues

What example applications exist which utilize the language? How many of these are from or representative of the bioinformatics domain?

[The two questions below are asking about the ability to express non-domain relevant information in the ontology, so that, for example, one could include user model information (preferences for viewers, etc.) or database access information (for access to persistent instance-level information) in the domain model.]

Can the ontology be partitioned, for example, into biology and bioinformatics (e.g., a protein has an accession number)?

Can the core ontology be extended to include other information, e.g., mappings to functions in databases, control information for showing the ontology through interfaces.

Appendix B: Property Definitions

This appendix provides explanations of the properties (column 1) in Evaluation Matrix 2, which provides a comparison of the expressive power of the ontology-exchange languages. Properties are:

Negation: Does the language allow the assertion that a relation does not hold between x and y?

Conjunction: Does the language allow the assertion that a relation holds both between (x, y) and between (x, z)?

Disjunction: Does the language allow the assertion that a relation holds both between (x, y) or between (x, z), but not both?

Relations: Does the language allow the mapping of the elements of a set A to the elements of a set B?

Multiple inheritance: Can the language describe inheritance of a child class from multiple parent classes?

Inverses: Can the language encode that slot X and slot Y are inverses of one another?

Multi-valued slots: Can the language encode slots that may have multiple values?

Multiple collection types: Can the language encode slots with different collection types such as bags, sets, and sequences?

Number restrictions on slots: Can the language encode constraints on the number of values a slot may have?

Slot hierarchies: Can the language encode taxonomic hierarchies of slots?

Facets: Can the language encode facets (facets encode properties of slots)?

Default values: Can the language encode default slot values?

Other slot constraints: Can the language encode other types of constraints on slot values, such as numeric ranges?

Primitive datatypes: What primitive datatypes does the language support? “Standard” indicates standard datatypes such as numbers, strings, Boolean.

Data model: What database data model does the language support?

Instances and classes: Can the language encode information about instance objects as well as class objects?

Appendix C: Ontology Example

The example representations below contain an encoding of the class Genes from the EcoCyc ontology in the OML and Ontolingua languages. The Genes class represents the concept of a procaryotic coding region. In both languages, the definitions in each example define the class itself, and then define the slots (attributes and relations) associated with that class.

OML Representation

```
<CKML>
  <Ontology id="Riley's Gene Classes" version="1.0">
  <comment> This OML ontology defines an encoding of the gene classification
system developed by Monica Riley.
  </comment>
  <extends ontology="http://www.ckml.org/ontology/" prefix="CKML"/>
  <Object type="Genes">
    <comment> The class of all genes is divided into several subclasses.  Genes
whose function is unknown or known only approximately are grouped
into the classes ORFs and Unclassified-Genes, respectively.  Genes
of known function have been classified using two orthogonal classification
schemes developed by Monica Riley.  One scheme classifies genes according
to the physiological role of their product class (Physiological-Roles); the other
scheme classifies genes according to the function of their product, such
as enzymes and transport proteins (Product-Types).
    </comment>
  </Object>

  <Function type="LEFT-END-POSITION" srcType="Genes" tgtType="data.Real"/>
  <Function type="INTERRUPTED?" srcType="Genes" tgtType="data.Boolean">
    <comment> The value of this slot is T for genes that are interrupted,
i.e., those that have an early stop codon inserted.
    </comment>
  </Function>
  <BinaryRelation type="HISTORY" srcType="CKML#Object" tgtType="data.String">
    <comment> Contains a textual history of changes made to this frame.  Each item
is either a string or a note frame. </comment>
  </BinaryRelation>
  <Theory genus="Evidence">
    <Object type="EXPERIMENT"/>
    <Object type="SEQUENCE-ANALYSIS"/>
  </Theory>
  <BinaryRelation type="EVIDENCE" srcType="Genes" tgtType="Evidence">
    <comment> Describes evidence for the defined function of this object.
Currently we distinguish between function that is determined
experimentally, and function that is determined through
computational sequence analysis.
    </comment>
  </BinaryRelation>
  <Function type="CENTISOME-POSITION" srcType="Genes" tgtType="data.Real">
    <comment> This slot lists the map position of this gene on the chromosome
in centisome units. </comment>
  </Function>
  <BinaryRelation type="CITATIONS" srcType="CKML#Object" tgtType="data.String">
    <comment> This slot lists general citations pertaining to the object containing
the slot.  Each value of the slot is a citation of the form
[reference-id]. </comment>
  </BinaryRelation>
  <BinaryRelation type="COMMENT" srcType="CKML#Object" tgtType="data.String">
    <comment> The Comment slot stores a general comment about the object that
contains the slot. </comment>
  </BinaryRelation>
  <Function type="COMMON-NAME" srcType="CKML#Object" tgtType="data.String">
    <comment> The primary name by which an object is known to
scientists -- a widely used and familiar name (in some cases
arbitrary choices must be made). </comment>
  </Function>
  <Theory genus="Transcription-Direction">
    <Object type="+"/>
    <Object type="-"/>
  </Theory>
  <Function type="TRANSCRIPTION-DIRECTION" srcType="Genes"
tgtType="Transcription-Direction">
    <comment> This slot specifies the direction along the chromosome in which
this gene is transcribed; allowable values are + or -. </comment>
  </Function>
  <BinaryRelation type="PRODUCT" srcType="Genes" tgtType="Polypeptides"/>
  <BinaryRelation type="SYNONYMS" srcType="CKML#Object" tgtType="data.String">
    <comment> One or more secondary names for an object -- names
that a scientist might attempt to use to retrieve the object.
The Synonyms should include any name a user might use to
try to retrieve an object. </comment>
  </BinaryRelation>
  <BinaryRelation type="PRODUCT-STRING" srcType="Genes" tgtType="data.String">
    <comment> This slot holds a text string that describes the product of this
gene;
this slot is only used when EcoCyc does not describe the gene product
as a frame (such as a polypeptide frame). </comment>
  </BinaryRelation>
  <Theory genus="Product-Types">
    <Object type="ENZYME"/>
    <Object type="REGULATOR"/>
    <Object type="LEADER"/>
    <Object type="MEMBRANE"/>
    <Object type="TRANSPORT"/>
    <Object type="STRUCTURAL"/>
    <Object type="RNA"/>
    <Object type="PHENOTYPE"/>
    <Object type="FACTOR"/>
    <Object type="CARRIER"/>
  </Theory>
  <BinaryRelation type="PRODUCT-TYPES"
srcType="Genes" tgtType="Product-Types">
    <comment> Describes the type of the gene product,
e.g., is it an enzyme, an
RNA, etc. </comment>
  </BinaryRelation>
  <Function type="RIGHT-END-POSITION"
srcType="Genes" tgtType="data.Real"/>

  <Collection.Object>
    <Genes id="EG10707" text="pheA">
      <LEFT-END-POSITION tgt="2735765"/>
      <CENTISOME-POSITION tgt="58.97035d0"/>
      <TRANSCRIPTION-DIRECTION tgt="+"/>
      <RIGHT-END-POSITION tgt="2736925"/>
    </Genes>
  </Collection.Object>
  <Collection.BinaryRelation>
    <EVIDENCE src="EG10707" tgt="EXPERIMENT"/>
    <NAMES src="EG10707" tgt="pheA"/>
    <NAMES src="EG10707" tgt="b2599"/>
    <PRODUCT src="EG10707"
tgt="CHORISMUTPREPHENDEHYDRAT-
MONOMER"/>
    <PRODUCT-STRING src="EG10707"
tgt="chorismate mutase-P and prephenate
dehydratase"/>
  </Collection.BinaryRelation>
```

Ontolingua Representation

```
(DEFINE-CLASS |Genes| (?X)
 "The class of all genes is divided into several subclasses. Genes
whose function is unknown or known only approximately are grouped
into the classes ORFs and Unclassified-Genes, respectively.  Genes
of known function have been classified using two orthogonal classification
schemes developed by Monica Riley.  One scheme classifies genes according
to the physiological role of their product class (Physiological-Roles); the other
scheme classifies genes according to the function of their product, such
as enzymes and transport proteins (Product-Types).
" :DEF (AND (|DNA-Segments| ?X)))
     ?VALUE)))

(DEFINE-FUNCTION CENTISOME-POSITION
(?FRAME) :-> ?VALUE
 "This slot lists the map position of this gene on the chromosome
in centisome units." :DEF (AND (|Genes| ?FRAME)
(NUMBER ?VALUE)))

(DEFINE-RELATION CITATIONS (?FRAME ?VALUE)
 "This slot lists general citations pertaining to the object containing
the slot.  Each value of the slot is a citation of the form
[reference-id]." :DEF (AND (|Organisms| ?FRAME)
(STRING ?VALUE)))


(DEFINE-RELATION COMMENT (?FRAME ?VALUE)
 "The Comment slot stores a general comment about the object that
contains the slot." :DEF (AND (:THING ?FRAME)
(STRING ?VALUE)))

(DEFINE-FUNCTION COMMON-NAME (?FRAME) :->
?VALUE
 "The primary name by which an object is known to
scientists -- a widely used and familiar name (in some cases
arbitrary choices must be made)." :DEF (AND (|Organisms|
?FRAME) (STRING ?VALUE)))

(DEFINE-RELATION EVIDENCE (?FRAME ?VALUE)
 "Describes evidence for the defined function of this object.
Currently we distinguish between function that is determined
experimentally, and function that is determined through
computational sequence analysis.
" :DEF (AND (|Genes| ?FRAME) ((:ONE-OF
:EXPERIMENT :SEQUENCE-ANALYSIS) ?VALUE)))

(DEFINE-RELATION HISTORY (?FRAME ?VALUE)
 "Contains a textual history of changes made to this frame.
Each item is either a
string or a note frame."
 :DEF (AND (:THING ?FRAME) ((:OR :STRING |Notes|)
?VALUE)))

(DEFINE-FUNCTION INTERRUPTED? (?FRAME) :->
?VALUE
 "The value of this slot is T for genes that are interrupted,
i.e., those that have an early stop codon inserted.
" :DEF (AND (|Genes| ?FRAME) (BOOLEAN
?VALUE)))

(DEFINE-FUNCTION LEFT-END-POSITION
(?FRAME) :-> ?VALUE "" :DEF
 (AND (|DNA-Segments| ?FRAME) (NUMBER
?VALUE)))

(DEFINE-RELATION PRODUCT (?FRAME ?VALUE)
 "This slot lists the product of a gene, which could be a
polypeptide or a tRNA.
```

```
Multiple products will be recorded in the case that several chemically
modified forms of the protein product exist.
" :DEF (AND (|Genes| ?FRAME) ((:OR |Polypeptides| RNA) ?VALUE)))

(DEFINE-RELATION PRODUCT-STRING (?FRAME ?VALUE)
 "This slot holds a text string that describes the product of this gene;
this slot is only used when EcoCyc does not describe the gene product
as a frame (such as a polypeptide frame)." :DEF
 (AND (|Genes| ?FRAME) (STRING ?VALUE)))

(DEFINE-RELATION PRODUCT-TYPES (?FRAME ?VALUE)
 "Describes the type of the gene product, e.g., is it an enzyme, an RNA, etc." :DEF
 (AND (|Genes| ?FRAME)
     ((:ONE-OF :ENZYME :REGULATOR :LEADER :MEMBRANE :TRANSPORT :STRUCTURAL :RNA
        :PHENOTYPE :FACTOR :CARRIER)
       ?VALUE)))

(DEFINE-FUNCTION RIGHT-END-POSITION (?FRAME) :-> ?VALUE "" :DEF
 (AND (|DNA-Segments| ?FRAME) (NUMBER ?VALUE)))

(DEFINE-RELATION SYNONYMS (?FRAME ?VALUE)
 "One or more secondary names for an object -- names
that a scientist might attempt to use to retrieve the object.
The Synonyms should include any name a user might use to try to retrieve an object." :DEF
 (AND (|Generalized-Reactions| ?FRAME) (STRING ?VALUE)))

(DEFINE-FUNCTION TRANSCRIPTION-DIRECTION (?FRAME) :-> ?VALUE
 "This slot specifies the direction along the chromosome in which
this gene is transcribed; allowable values are + or -." :DEF
 (AND (DNA ?FRAME) ((:ONE-OF "+" "-") ?VALUE)))
```

References

Chaudhri, V.K., Farquhar, A., Fikes, R., Karp, P.D., Rice, J.P.. 1998. OKBC: A Programmatic Foundation for Knowledge Base Interoperability. *Proceedings of the AAAI-98*.

Cyc, *Features of CycL*, http://www.cyc.com/cycl.html.

Webb, Edwin C. 1992. Enzyme Nomenclature: Recommendations of the nomenclature committee of the International Union of Biochemistry and Molecular Biology on the nomenclature and classification of enzymes. *Eur. J. Biochem*.

Karp, P., Riley, M., Paley, S., Pellegrini-Toole, A., Krummenacker, M. 1999. EcoCyc: Electronic Encyclopedia of *E. coli* Genes and Metabolism. *Nuc. Acids Res*., Vol 27 No 1, pp 55-58.

Fikes, R., Kehler, T. 1985. The Role of Frame-Based Representation in Reasoning *Communications of the Association for Computing Machinery*, 28(9):904-920.

Gruber, T.R. 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199-220.

Guarino, N. and Giaretta, P. 1995. Ontologies and knowledge bases towards a terminological clarification. *Towards very large knowledge bases*, pp25-32.

Kent, Robert. 1999. Conceptual Knowledge Markup Language: The Central Core. *Knowledge Acquisition Workshop 1999*, http://sern.ucalgary.ca/ksi/kaw/kaw99/papers/Kent1/CKML.pdf (also Power Point slides) http://sern.ucalgary.ca/ksi/kaw/kaw99/papers/Kent1/CKML.ppt.

Karp, P.D. 1992. The design space of frame knowledge representation systems, SRI International AI Center, #520, URL ftp://www.ai.sri.com/pub/papers/ karp-freview.ps.Z.

Karp, P.D., Chaudhri, V, *XOL Specification*, http://www.ai.sri.com/pkarp/xol/.

Kosky, A.S., Chen, I.A., Markowitz, V.M., Szeto, E. 1998. Exploring Heterogeneous Biological Databases: Tools and Applications, In *Proceedings of the 6th International Conference on Extending Database Technology*, 499-513, Lecture Notes in Computer Science Vol. 1377, Springer-Verlag.

Larmouth, J., 1999. *ASN.1 Complete*, Morgan Kaufmann Publishers, October

Lenat, D.B., Guha, R.V. 1990. *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Reading, Mass, Addison-Wesley. (http://www.cyc.com)

Cattell, R.G.G., et al, 2000. *The Object Data Standard: ODMG 3.0*, Morgan Kaufmann Publishers

Rice, James, *OKBC Specification*, http://www.ai.sri.com/~okbc/spec.html.

Object Management Group, 1997. *UML Semantics, Version 1.1*, ftp://ftp.omg.org/pub/docs/ad/97-08-04.pdf.

Riley, M., 1993. Functions of the gene products of Escherichia coli, *Microbiological Reviews*, 57:862-952.

Rumbaugh, J., Jacobson, I., Booch, G., 1998. *The Unified Modeling Language Reference Manual*, Addison-Wesley.

St. Laurent, Simon. 1998. *XML: A Primer*, IDG Books Worldwide.

Topaloglou, T., Kosky, A., Markowitz, V., 1999. Seamless Integration of Biological Applications within a Database Framework, In *Proceedings of ISMB '99*.

W3C, http://www.w3c.org/XML.