A Preliminary Analysis of Visualization Requirements in Knowledge Engineering Tools

Neil A. Ernst
Dr. Margaret-Anne D. Storey
Computer-Human Interaction and Software Engineering Lab
Computer Science Department, University of Victoria
Victoria, British Columbia, Canada
nernst@uvic.ca, mstorey@uvic.ca

***Abstract*— One of the central themes of our research is conceiving of ontologies as mechanisms which facilitate communication and aid interoperability in a number of domains. This report describes the design and results of a survey we conducted from November 2002 - January 2003 to further this theme. This survey, distributed to subscribers of the Protege-2000 discussion list and the rdf-interest list, elicited some background facts and visualization requirements of users of the Protege tool, as well as visualization in general. We discovered that respondents varied quite widely in domain, number of users, and experiences with visualization. As a result, we conclude that the best strategy for visualization tool developers is one emphasizing domain independence and customization.**

I. Introduction

Knowledge engineering is a systematic approach to building knowledge-aware applications. The term 'engineering' is here used in the sense first described in Mary Shaw's seminal paper on software engineering [1], that is, engineering is concerned with "creating cost-effective solutions... to practical problems... by applying scientific knowledge... to building things... in the service of mankind". Implicit in this definition is the notion that knowledge engineering follows a systematic and rational process. This process is complex and multi-faceted, with several different stages, including knowledge modeling, knowledge acquisition, and verification. The knowledge modeling phase requires building a set of formal descriptions of the concepts that exist in the domain and the relationships between these concepts. We call such a collection of descriptions an ontology [2], and note that such collections can take a number of forms, from highly formal, machine readable, symbolic representations, to less formal, human-centric text representations, such as those libraries use to classify books. In most types of ontology languages, the concepts can be mapped as vertices and the relationships as edges, permitting the ontology to be understood using graph-theoretic concepts such as trees; the large volume of research into graphs can be leveraged to support the ontology engineering process (such as efficient graph layouts).

A. Protégé-2000

One tool that is used to build these systems is called Protégé-2000 [3]. Originating as a system for modelling medical guidelines and protocols, this tool provides an interface to create and model ontologies, as well as to acquire knowledge based on that ontology. A popular example used to illustrate this process is that of an ontology of wines and associated meal courses [4]. An ontology Developer creates an ontology which defines what wine is (fermented grape beverage, for example), what food is, and how they relate to each other, among other things. When this model satisfies the requirements for the system (gathered at a preliminary stage), instances are collected/acquired from domain experts (vintners, oenophiles, etc.) to expand the knowledge model to include data that fits the model. The combination of data and knowledge model (ontology) can now be termed a knowledge base, and software tools can be used in conjunction to create knowledge-aware applications. As the application is used, the original ontology may be refined, either by the Developers or Guru users, to improve accuracy.

Ontological engineering has seen a surge in popularity recently, due to the increasing pervasiveness of Internet technology, and more recently, the Semantic Web initiative [5]. This vision sees the Internet leveraged to allow machines (software or hardware) to make use of data. For example, a software agent might visit a winery and request current prices for a particular type of wine, and then return that information to the User. Ontologies in this vision act as a way to add structure to the data – in the form of simple rules and axioms – and also to model information exchange, as a communication facilitator.

One of the emerging difficulties researchers and developers are encountering is that humans have an increasingly difficult time making sense of larger ontologies, due to innate cognitive limitations. One example, that our research group works with, models cancer research, and is composed of almost 30,000 classes and a further 30,000 instances, connected with 60+ relationships. In an ideal world, human involvement would be required only at the design stage. However, in our view humans will still be required to make sense of these ontologies, at least in the short term; we suggest three possible cases where human intervention may be required:

- When mapping ontology A to ontology B, at some point a higher semantic level will be required to identify the modelling decisions.
- Verifying that an ontology models what the user expects

it to model.
- When extracting parts of a larger ontology – machines currently are unable to make accurate decisions regarding which parts to remove.

In order to facilitate human understanding, we have relied on the techniques - some admittedly quite familiar - of information visualization.

B. Information Visualization

Information visualization represents information in a manner which aids in communication and facilitates understanding and exploration. While our tool, Jambalaya [6], does not present much in the way of novel information views, such as new layouts of large graphs, we contend that this fascination with novel visualization techniques is quite damaging, and certainly alienating from a central theme of information visualization: a means for communication. We argue that if your view requires the user to first understand what the view is attempting to show, then the tool has already failed in the larger goal of communication. Many innovative visualizations suffer from lack of industry acceptance. This lack of adoption restricts meaningful evaluation of the ideas in the tool.

With this concept of simplicity in mind, we made conscious and deliberate efforts to focus on creating a tool which will focus on the key goals of communication and understanding, leveraging the techniques that best facilitate those goals. As such, we typically describe Jambalaya as a visualization toolkit, with numerous different techniques and methods for making sense of an ontological information space. In conjunction with other studies we have done, particularly in the area of software understanding, our group has developed a customization framework for developing software products which will support end-user customization. In section I-A, we introduced three user types, namely the User, the Guru/Designer, and the Developer. This typology was introduced in Michaud and Storey [7], who first developed this framework for customization. In the knowledge engineering community, Users are people who use a knowledge-based system; Gurus are users who design and enhance these systems, and typically have greater technical skill with the toolsets; finally, the Developer is someone who constructs and maintains the ontology itself.

II. Survey Goals

The Jambalaya project has been ongoing since January 2001. Its ancestor was a software understanding tool called SHriMP that applied information visualization techniques to the problem of understanding large software systems [8]. This approach seemed highly adaptable to the problem of navigating and understanding large knowledge-based systems, and thus an effort was made to integrate the SHriMP tool with Protege-2000 [9]. Our initial efforts were targeted towards technical success – goals were defined in terms of how well our software integration efforts worked. This focus was maintained until we realized we had little idea who the users of the tool were, and how they might use it. It was at this point that we realized that our software development approach was somewhat backward, since we began with an idea, implemented it and *then* sought to validate the approach with the users. However, this methodology has had some rewards. For example, we were able to use a sophisticated tool as a prototype with potential users, to gauge the level of interest as well as seek out specific challenges our tool was not meeting.

This survey was conducted to provide an initial and preliminary picture of who the users were, both of Jambalaya and ontology visualization tools in general, and to seek some goals these users had for such tools. These questions were unanswered in previous research on the area of visualization in knowledge engineering, and therefore provides an early attempt to elicit a broad picture of the user community for this growing field (for examples, see [10], [11], [12], [13], [14], to cite a few).

III. Methodology

We decided to release the survey electronically. We designed a survey according to principles described in ([15], p. 133-134), using a combination of scalar and open-ended questions to cover our topics of interest. The survey (see Appendix) focused on some major areas in which we were interested in gaining respondent information:

1) User information - familiarity with ontologies, with Protege, with visualization techniques
2) Domain of use
3) Ontology usage tasks - editing, browsing, etc.
4) Visualization requirements

To target a large population of potential respondents, we decided to announce the survey on two large mailing lists, automated discussion services that collect email messages from subscribers and redistribute them to the other members. These lists were the Protege-2000 discussion list, with approximately 5300 members with an interest in the Protege tool, and a list concerning the W3C specification for the Resource Description Framework [16], *rdf-interest*. Finally, we posted a link on our web site, `http://shrimp.cs.uvic.ca`, which receives approximately 30-40 visits a day. We received 44 responses between the dates of November 2002 - January 2003, when the survey was taken offline.

IV. Results

The raw results of the survey are available by request. This section summarizes the results by question and presents some simple graphics to present the salient points. Where there were similar long-answer responses, such as to 'other:' questions, we have collected similar responses. The number in parentheses, e.g. **(4)**, indicates the number of respondents. Multiple responses may mean not all categories tally to 44.

0. Approximate number of Protégé-2000 users in your organization (see Fig. 1):

- 1 user: **(20)**
- 2-5: **(13)**
- 6-10: **(4)**
- 10+: **(4)**

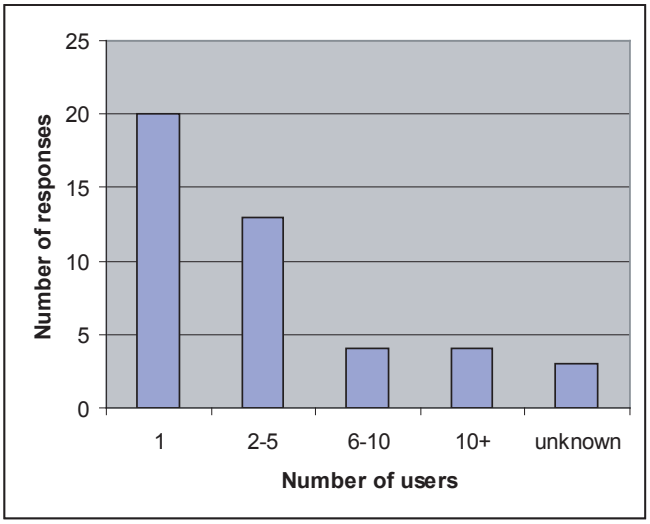

Fig. 1. The number of users of Protege-2000 in respondents' organization.

- unknown: **(3)**

1. I would characterize my ontology experience as:

- None or little **(2)**
- I am interested in using ontologies. **(12)**
- I already use ontologies **(8)**
- I develop or do research with ontologies (and possibly use them as well). **(21)**
- Other **(1)**: Doing Knowledge Engineering and as such connect with ontology management and possibly existing ontologies.

2. Select which phrase best describes your usage of ontologies (check all that apply).

- I author and edit ontologies directly in Protégé-2000 for other users. **(13)**
- I manage projects (including research) which deal with ontologies. **(22)**
- I deal with different ontologies and encounter new ones often. **(13)**
- I primarily work with one ontology. **(9)**
- Other **(2)**: I'm implementing an ontology-based application for enterprise level use and enterprise integration; research into ontology theory & applications; teaching ontology courses

3. What prompted you or might prompt you to select Protégé-2000 as your development environment (check all that apply)?

- Widely accepted **(23)**
- Relatively simple to understand and begin to use **(27)**
- Frame-based knowledge representation **(19)**
- Extensible platform for custom applications **(27)**
- Other **(11)**: freely available; it provides different representation formats (RDF, JDBC, later on DAML); collaboration with Stanford Medical Informatics; recommended by my advisor; Most ontologically complete; just preparing to download, not knowing Protege; Connect-ability with my Knowledge Engineering environment;

4. What domain will/do(es) your ontology project(s) describe? (examples: clinical guideline modeling, e-business communication). Where no number is specified, only one person answered that way.

- MEDICAL
  - Clinical guideline modeling (4)
  - Organ transplant management
  - Endocrinology
  - Interventions to change Health Behavior
  - Drug therapy optimization
  - Clinical trials for promoting healthy behaviors
  - Psychological research
  - Disease/syndromic surveillance
- COMPUTER/SOFTWARE
  - PIM-like
  - Software development
  - Semantic image retrieval
  - e-business communication
  - Municipal information systems
  - Software architecture
  - Program comprehension
  - e-business and enterprise architectures management
- OTHER
  - Domain independent or various (3)
  - Multiple (e.g., computer science; information ontology; ...)
  - People skills within Manufacturing
  - Geology–geologic Units, Earth materials, Structure
  - Mineral deposit modelling; earth science, environmental and land-use mapping
  - Various legal domains
  - Architectural, geographical, historical, economics, social, politics
  - Biodiversity data and metadata
  - E-financing, telecommunication
  - Publication
  - Injection molds
  - Biological and chemical warfare defence
  - DOD Data Dictionary Analysis for C4ISR
  - Art and culture, company dictionary, website structures
  - Generalized endeavor (i.e., enterprise) intelligence and operations
  - Complex labour taxonomic informations (more than 200 labour contracts)
  - Cultural Heritage
  - Divorce legal help
  - Product configuration
  - Military application

5. What size of ontology do you commonly work with or anticipate working with?

- Fewer than 100 frames (classes + instances) **(10)**
- Between 101 and 1000 frames **(19)**
- Between 1001 and 10,000 frames **(8)**
- Between 10,001 and 100,000 frames **(7)**

- Greater than 100,001 frames **(0)**

6. What tasks do you (might you) carry out on your ontology?

a. I try to capture a higher level of understanding of the ontology through browsing.

always	often	sometimes	rarely	never
11	13	14	3	1

b. I usually work on a small subset of the ontology.

always	often	sometimes	rarely	never
3	14	15	6	2

c. I find myself switching back and forth between certain subsets.

always	often	sometimes	rarely	never
7	19	9	2	2

d. I work on different aspects of the ontology, so I often don't know where I will be in the ontology structure.

always	often	sometimes	rarely	never
0	0	0	0	0

7. In what ways would you like a graph visualization tool to aid your work?

- Assist with navigating information space **(33)**
- Show hidden relationships **(31)**
- Show areas of interest **(21)**
- Check for inconsistencies or errors **(34)**
- Present reports to others **(24)**
- Help with understanding new ontologies **(28)**
- Other **(3)**: to describe relationships with instances; diagram direct and indirect relationships to a given subject

8. What ontology visualization tools have you used?

- Jambalaya **(14)**
- OntoViz **(16)**
- Ozone **(0)**
- IsaViz **(5)**
- Kaon **(4)**
- Other **(7)**:
  - Webode
  - Ontoweb
  - Ontosaurus
  - Custom-built tool since 1988, using RDBMS with my object-metaschema
  - IODE
  - OilEd
  - The diagram widget in Protégé
  - Custom-built, based on ATT's GraphViz
  - RDFAuthor
  - Various hacks
  - Protege Knowledge Explorer - shows multiple-instance slots in a tree view
  - UML & ER

9. Other comments:

- Different parts of an ontology often have different visualization requirements. I don't think one metaphor will suffice. For example, a part of ontology may be mostly a classification of concepts; another part deals with structure of some entities.
- We are doing research on image-based ontology
- Graphical presentation of ontologies to our community of scientists and managers concerned with biodiversity issues is quite important.
- Nice displays. I had some trouble initially figuring out how to get the tree display to work.
- I really like Jambalaya a lot, but domain experts working here say they are intimidated by it, but I haven't figured out why.

V. Discussion

For a survey that did not delve into specifics, such as preferred layout styles or favorite initial view, we nevertheless believe we have obtained some interesting results. We frame our discussion using the four central themes of interest:

1) User information - familiarity with ontologies, Protege, visualizations
2) Domain of use
3) Ontology usage tasks - editing, browsing, etc.
4) Visualization requirements

A. User information

The respondents to the survey were quite varied. The majority of respondents (45%) were the sole user of the tool, yet a number (18%) were serving user communities of 6 or more. This shows that ontologies are increasingly being used by multi-user teams. The implications of this variation are that visualization support may be best targeted at the 'Guru' users, who have a greater familiarity with the tool and presumably, a greater ability to manipulate it as needed.

Over half the users developed ontologies using Protege, and we therefore place them in the Developer category of user – that is, those users who create applications or ontologies for others to use. Furthermore, respondents were likely to be managing projects which deal with ontologies as well. Most ontologies were between 100 and 1000 frames in size (a frame was defined as a class or an instance). Some users had ontologies up 100,000 frames, and no respondents had an ontology greater than 100,000 frames, although these systems certainly exist. This is positive for our efforts at making our tool scalable. We've gone from comfortably handling ontologies of fewer than 10,000 frames to some that are nearing 100,000 frames.

B. Domains

Domains were, as expected, quite varied. In keeping with Protege's origins as a medical informatics tool, a large number of domains were medical-related. A further group was centered on computer-based applications – again, not surprising. The other domains were quite assorted and defy categorization; this demonstrates the futility and impracticality of attempting

to design tools targeted to a specific domain. Indeed, in combination with section V-A above, we are convinced that any visualization solution to knowledge engineering problems must attempt to be domain independent. While this goal is extremely complex, we believe that a general domain-independent approach, coupled with a customization framework to support domain-specific configuration, is a solution that will address many of the issues. This approach will hopefully address the issue raised in the last question of the survey, where one respondent mentioned that "domain experts working here say they are intimidated by it"; a customizable solution should allow the respondent to configure the tool so the experts are no longer intimidated.

C. Ontology tasks

In keeping with the large number of domains, a number of tasks were common to the respondents. Most respondents replied that they often use tools to capture a higher understanding of the ontology (57%). However, respondents also often worked on a small subset of the ontology, as opposed to browsing the entire set of concepts. We suggest that this duality comes from the distinction between developing one's own ontologies, and integrating others. Understanding this duality can help us to target what views our tool should focus on. Finally, a lot of people (62%) switched between small subsets of their ontology when working on it. We discovered that few of the respondents actually browsed the ontology at a high-level (as we often do when presented with an unfamiliar ontology), likely because our respondents tend to be ontology engineering Gurus or Developers.

D. Visualization requirements

Our last two questions addressed visualization in ontology editing tools. We were attempting to discover a) whether people thought this idea was valid; and b) if so, what role should it play? The responses to these questions (7 and 8 above) were quite widespread, and it is difficult to show any one area as a focus. Certainly, users were interested in navigation and error-checking, primarily, but other roles, such as understanding hidden relationships, were not significantly less desirable. The conclusion we draw from this is positive; we feel this breadth of interest suggests that there are many improvements visualization can provide. This seems clear when we examine the different tools used for visualization (question 8). Certainly Jambalaya is highly used, given the nature of the questionnaire, as well as its tight integration with Protege, but there seems to be a wide variety of other tools that are used, from custom solutions to simple visualizations which come with other ontology editors (Ontoweb, WebOde, OilEd). This further supports our contention that the interest in visualization is increasing, if only because for so long, tool support for common tasks like understanding and communication have been so poor.

VI. Conclusions

This survey has shown that there are a wide variety of users and domains to which ontology engineering is being applied, and further, that visualization is a desired feature. The lesson for those working with tools which manipulate and create ontologies is that this diversity must be supported. We believe that the wide-ranging degree of domains is a sign of the future, and that tools that operate at a meta level to assist users to understand the modeling decisions, such as Jambalaya, will be increasingly important in maintaining clear communication and understanding. It is important to keep in mind that a survey such as this cannot address all the possible uses or reasons something occurs. We use such a survey as a pointer to areas where more detailed investigation might be useful. For example, we have been conducting in-depth studies, using contextual inquiry techniques, to identify what requirements users have. This methodology has shown itself to be quite powerful at identifying some important aspects of the ontology editing process and tools.

Acknowledgements

We would like to acknowledge the respondents for their insightful comments and participation; our fellow researchers, who have provided useful feedback, especially in the design phase of the survey; and finally, the Protege-2000 team, for their ongoing assistance and friendship. The Protege resource is supported, in part, by grant P41 LM007885 from the National Library of Medicine.

References

[1] M. Shaw, "Towards an engineering discipline of software," *IEEE Software*, pp. 15–24, November 1990.

[2] T. Gruber, "A translation approach to portable ontology specification," *Knowledge Acquisition*, vol. 5, pp. 199–220, June 1993.

[3] W. Grosso, H. Eriksson, R. W. Fergerson, J. Gennari, S. Tu, and M. A. Musen, "Knowledge modeling at the millennium (the design and evolution of protege-2000)," Stanford Medical Informatics, Technical Report SMI-1999-0801, 1999.

[4] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Stanford Medical Informatics, Technical Report SMI-2001-0880, 2001.

[5] *The Semantic Web*, May 17 2001.

[6] M.-A. D. Storey, M. A. Musen, J. Silva, C. Best, N. Ernst, R. Fergerson, and N. F. Noy, "Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in protege," in *Workshop on Interactive Tools for Knowledge Capture, K-CAP-2001*, Victoria, B.C. Canada, 2001.

[7] J. Michaud and M.-A. D. Storey, "The role of knowledge in software customization," in *15th Int. Conf. on Software Engineering and Knowledge Engineering (SEKE03)*, San Francisco Bay, CA, 2003.

[8] M.-A. Storey, F. Fracchia, and H. A. Mueller, "Cognitive design elements to support the construction of a mental model during software exploration," *Journal of Software Systems: special issue on Program Comprehension*, vol. 44, pp. 171–185, 1999.

[9] C. Best, "Designing a component-based framework for a domain independent visualization tool," M.Sc., University of Victoria, 2002.

[10] E. Pietriga. (2003) Isaviz: A visual authoring tool for rdf. World Wide Web Consortium. [Online]. Available: http://www.w3.org/2001/11/IsaViz/

[11] M. Sintek. (2003) Ontoviz tab: Visualizing protege ontologies. [Online]. Available: http://protege.stanford.edu/plugins/ontoviz/ontoviz.html

[12] P. Eklund, N. Roberts, and S. P. Green, "Ontorama: Browsing an rdf ontology using a hyperbolic-like browser," in *First International Symposium on CyberWorlds (CW2002)*. IEEE, 2002, pp. 405–411.

[13] G. K.-C. Ng, "Interactive visualization techniques for ontology development," Ph.D., University of Manchester, 2000.

[14] H. Alani, "Tgviztab: An ontology visualisation extension for protege," in *Knowledge Capture 03 - Workshop on Visualizing Information in Knowledge Engineering*. Sanibel Island, FL: ACM, 2003.

[15] B. Shneiderman, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 3rd ed. Reading, MA: Addison-Wesley, 1998.
[16] F. Manola and E. Miller. (2002) Rdf primer. World Wide Web Consortium. [Online]. Available: http://www.w3.org/TR/rdf-primer

VII. Appendix

The survey form is available at `http://shrimp.cs.uvic.ca/jambalaya/user-survey.html`.