Hands-on experiences in concurrent ontology building with Collaborative Protégé

Daniel Schober[1,2], James Malone[2], Robert Stevens[3]

[1]*Institute for Medical Biometry and Medical Informatics, University Clinic, Freiburg, Germany*

[2]*European Bioinformatics Institute, Cambridge, UK*

[3]*Manchester School of Computer Sciences, Manchester, UK*

Introduction

Although living in a world of greater international collaboration, the geographical distribution of developers still makes collaborative development approaches for ontology development difficult to realize. If collaboration is not built into tools, reaching widespread community consensus and encoding according to some shared plan is not easy to achieve. For this reason, tools are developed that allow not only for distributed collaborative ontology creation and modification, but for direct and topic-linked communication about all aspects of the engineering process as well. To investigate this process and corresponding capabilities of the new Collaborative Protégé 3 (CP) tool, the Ontology of Biomedical Investigations (OBI) was enriched in an experiment ran as part of an OntoGenesis network (website: http://ontogenesis.ontonet.org/moin/NetworkMeeting7) at the European Bioinformatics Institute (EBI).

We investigated the CPs plugins [1] ability to:

- Facilitate multiple concurrent edits of a single owl file from different computers
- Track annotations associated with specific representational units (RUs), e.g. on classes or properties
- Track annotations associated with actions of ontology change (deletions, axiom edits and annotation edits)
- Support for discussion threads and instant messaging communication between ontology developers (real time chat).

In this talk we present our observations and recommendations for CP based upon this experience.

Method

Our methodology involved the following set of tasks:

- Familiarization of users with Collaborative Protégé 3.4, its GUI and collaborative features.
- Ad hoc additions of attendee's own lists of devices (with possibility of duplication).
- Controlled additions of devices from a list as provided by the metabolomics standards initiative (MSI).

- Deployment of an 'Agent Provocateur' to assess the transparency of the changes occurring to others. These conflicting and deliberately incorrect edits were made during a specified period of time known only to the session organizer.
- Controlling/restricting communication channels (notes, discussion threads and chat) to evaluate CPs ability to facilitate communication in distributed, collaborative development.

The communication and interaction of the participants with each other, directly or through the tool, were tracked and analyzed. The Obi.owl file was populated with new ‘device’ classes from the domains of the OntoGenesis members and as taken from a list provided by the Metabolomics Standard Initiative (http://msi-ontology.sourceforge.net/). Detailed statistics on numbers and kinds of annotations made during the sessions with tables, diagrams and further discussions are available in a spreadsheet from the OntoGenesis website. Initially, development occurred in a single group but this was then divided into subgroups. Ad hoc additions were made which was followed by subgroups adding classes from the provided MSI term list. The results were then reviewed and commented by the other subgroups adding annotations/notes. Subsequently, other communication channels were tested. First, chat only, then by voice only and after that by chat and voice together. During the latter stages of this session, the Agent Provocateur user was deployed.

Results

A realistic collaborative ontology building session was set up to test CPs collaborative editing and communication features. Collaborative ontology building was relatively trouble free and the tool also copes with complicated setups and is flexible enough to allow for corresponding adjustments.

We highlight some unfulfilled requirements:

Editing functionalities:

The lack of a RU and module locking mechanism meant that others could alter classes that have a logical impact on the class under current definition by another user. A roll back function would aid in conflict resolution and would lead to safer editing. Subscription and Notification were requested, where users subscribe to certain areas of interest within the ontology and are then notified of any changes that occur in those areas.

Annotations on RUs with entity notes:

For minor and trivial annotations providing an annotation type, subject heading and value in an overly granular manner was perceived as overkill. Also the change track captured in the project-linked ChAO knowledge base is sometimes presented in an overly granular manner. Users would like the changes to be described in a high level abstraction, rather than at a detailed granular level.

Communication:

Chats were requested to be linked with specific RUs and axioms to aid a more immediate and direct conflict resolution. A closed 'retreat room' was desired as well as a filter function on user names to enable to see only the chats of certain people or on particular ontology modules. Integration of emoticons in text fields would increase transmittance of pragmatic aspects of communications.

Planning:

Integrated voting functionalities allow users to vote on change issues. A mechanism that changes the ontology based on vote outcomes would increase development time and could be implemented using ChAO information and formalized voting outcomes. Issue tracker functions were requested, i.e. a scratch pad or todo list that can be worked through and 'checked', e.g. indicating a proposed plan and what has been already realized at a certain time point.

Conclusion

Although some caveats persist, it became clear that the CP tool is now in an advanced state and can be used in practice with sufficient stability and much can be done with configuration to further optimize it. Our practice-driven requirement and fault analysis provoked much feedback to the tool developers, and will be valuable for the CP version of P4, which is in preparation. A paper collating all results has been accepted for the ICBO 2009 Conference and will be published in their Proceedings. We will continue to investigate CP in further ontogenesis meetings and hence will gain further insights into the process of software guided collaborative functionalities for ontology engineering, ensuring continuous feedback to the CP developers.

Acknowledgements

The authors thank Tania Tudorache, Timothy Redmond and Natasha Noy from the Protégé developers team and James Watson from the EBI teaching facilities. Further thanks go to the ontogenesis network participants, and to Melanie Courtot, Alan Ruttenberg and the OBI Consortium for providing the OBI.owl file. DS is funded by the DebugIT project of the EU 7th FP (ICT-2007.5.2-217139). He was supported by EBI's NET-project funds (www.ebi.ac.uk/net-projects) during some initial work leading to this paper. JM is funded by the EMERALD EU project number LSHG-CT-2006-037686. The OntoGenesis Network is funded by EPSRC grant EP/E021352/1.

References

[1] Noy N, Tudorache T, de Coronado S, Musen M, Developing biomedical ontologies collaboratively. AMIA Annu Symp Proc. 2008, p. 520-4.