OntoCAT — an integrated programming toolkit for common ontology application tasks

Tomasz Adamusiak*[1], Natalja Kurbatova[1], Morris A. Swertz[1,2], and Helen Parkinson[1]

[1]European Bioinformatics Institute, Wellcome Trust Genome Campus, Cambridge, CB10 1SD, United Kingdom
[2]Genomics Coordination Center, Department of Genetics, University Medical Center Groningen and Groningen Bioinformatics Center, University of Groningen, P.O. Box 30001, 9700 RB, Groningen, The Netherlands

Availability

Website: www.ontocat.org
Source: www.ontocat.org/svn
License: LGPLv.3

1 Introduction

Ontologies are essential to data integration, query expansion, and modelling biological knowledge in life sciences. Two major public ontology repositories provide programmatic access: the EBI Ontology Lookup Service (OLS) [1] and the NCBO BioPortal [5]. Many users also develop local ontologies, so it is important to integrate queries to local files. However, it is relatively difficult to connect to each of them, in particular because these resources are still evolving or require considerable experience with ontologies themselves.

Therefore, we developed OntoCAT, a software toolkit that provides high level abstraction for interacting with ontology resources including local files in standard OWL and OBO formats (via OWL API [2]), and public ontology repositories: EBI OLS and NCBO BioPortal. The requirements for these were based on our own use cases of Experimental Factor Ontology (EFO) development, ArrayExpress and MOLGENIS data annotation and analysis, and on user feedback. Since its inception in 2010 only the Java package has seen 22 releases. Most recent progress includes the implementation of reasoning for querying of relations other than subsumption (e.g. partonomy). This is enabled for local ontologies via HermiT reasoner [4], which supports knowledge bases expressed in $\mathcal{SROIQ}^{(\mathcal{D})}$ – the description logic underpinning OWL2 (see also www.ontocat.org/wiki/Reasoning) and OLS, which provides a dedicated web service.

2 Implementation

The library is implemented in Java6 and is available under the permissive LGPLv3 license. OntoCAT can also be used via other interfaces including a web-based ontology database and browser, scriptable REST service, and Google App application.

OntoCAT was designed to support simple use cases in an easy to implement way, while still enabling the implementation of advanced algorithms. Many of such common tasks are demonstrated in code examples available at www.ontocat.org. A complete list of available ontology, term, and hierarchy methods named in a self-describing manner includes: *getOntologies()*, *getOntology()*, *searchAll()*, *searchOntology()*, *getTerm()*, *getAllTerms()*, *getAnnotations()*, *getSynonyms()*, *getDefinitions()*, *getRootTerms()*, *getTermPath()*, *getChildren()*, *getParents()*, *getAllChildren()*, *getAllParents()*, *getRelations()*.

OntoCAT follows the convention over configuration design approach, i.e., requiring minimal configuration where possible. *FileOntologyService*, *OlsOntologyService*, and *BioportalOntologyService* are the core objects for working with: OWL and OBO ontologies, EBI OLS and NCBO BioPortal respectively. Because each ontology service implements the same *OntologyService* interface, these core services can then be combined or extended to provide additional behaviour by adding a wrapper (decorator), e.g.: combination of multiple ontology resources into one service (*CompositeServiceDecorator*), limiting and ranking of search results (*SortedSubsetDecorator*), translating one ontology namespace to another (*TranslatedOntologyService*), Ehcache-based enterprise-grade caching (*CachedServiceDecorator*), or enabling reasoner support (*ReasonedFileOntologyService*).

The current repertoire of supported ontology resources could easily be extended for other resources such as DAML, Protégé-OWL API, ONKI API, or OntoSelect. Such services would only need to implement the *OntologyService* interface to immediately become aligned with pre-existing resources and allow for their seamless interchangeability.

3 Applications

OntoCAT is being used by the ontocat Bioconductor/R package [3] and the concept recognition tool Zooma (zooma.sf.net).

Acknowledgements

This work was supported by the European Community's Seventh Framework Programmes GEN2PHEN [grant number 200754], SLING [grant number 226073], and SYBARIS [grant number 242220]; The European Molecular Biology Laboratory; the Netherlands Organisation for Scientific Research [NWO/Rubicon grant number 825.09.008]; and the Netherlands Bioinformatics Centre [BioAssist/Biobanking platform and BioRange grant SP1.2.3].

Bibliography

[1] Ct, R. G., Jones, P., Martens, L., Apweiler, R., and Hermjakob, H. (2008). The Ontology Lookup Service: more data and better tools for controlled vocabulary queries. *Nucleic Acids Res*, **36**(Web Server issue), W372–W376.

[2] Horridge, M. and Bechhofer, S. (2009). The OWL API: A Java API for Working with OWL 2 Ontologies. In *OWLED 2009, 6th OWL Experienced and Directions Workshop, Chantilly, Virginia*.

[3] Kurbatova, N., Adamusiak, T., Kurnosov, P., Swertz, M. A., and Kapushesky, M. (in press). ontoCAT: an R package for ontology traversal and search. *Bioinformatics*.

[4] Motik, B., Shearer, R., and Horrocks, I. (2009). Hypertableau Reasoning for Description Logics. *Journal of Artificial Intelligence Research*, **36**, 165–228.

[5] Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A., Chute, C. G., and Musen, M. A. (2009). BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Res*, **37**(Web Server issue), W170–W173.

*To whom correspondence should be addressed: tomasz@ebi.ac.uk