CDAO Store: A New Vision for Data Integration

Brandon Chisham*, Trung Le*, Enrico Pontelli *, Tran Son *and Ben Wright *

Department of Computer Science, New Mexico State University, Las Cruces, New Mexico, USA

Email: Brandon Chisham*- bchisham@cs.nmsu.edu; Trung Le*- tle@cs.nmsu.edu; Enrico Pontelli *-epontell@cs.nmsu.edu; Tran Son *- tson@cs.nmsu.edu; Ben Wright *- bwright@cs.nmsu.edu;

*Corresponding author

Abstract

Background: The Comparative Data Analysis Ontology [1] is an ontology developed, as part of the EvoInfo[2] and EvoIO[3] groups supported by NESCent[4], to provide semantics to the descriptions of data and transformations commonly found in the domain of phylogenetic inference. The core concepts of the ontology enables the description of phylogenetic trees and associated character data matrices.

Results: Text for this section of the abstract ...

Conclusions: Text for this section of the abstract ...

Background

CDAO

CDAO, Comparative Data Analysis Ontology, provides a framework for describing phylogenies and their associated character state matrices. It was developed as part of the Evolutionary Informatics working group along with the NeXML file format, and the PhyloWS Webservice standard, forming what the group called the EvoIO stack. CDAO forms the base of this stack defining the semantics for the data represented

[1] http://www.evolutionaryontology.org
[2] https://www.nescent.org/wg_evoinfo/Main_Page
[3] http://evoio.org/wiki/Main_Page
[4] http://www.nescent.org/index.php

as NeXML files, or otherwise supplied by services implementing this set of standards.
CDAO is defined in terms of an OWL-DL ontology. It provides a general framework for talking about the relationships between taxa, characters, states, their matrices, and associated phylogenies. As a general framework it supplies general classes and relations between those classes, it is intended that for practical work these will be extended to for example talk about more specific types of characters or states. (e.g. Beak length might be defined as a specialization of CDAO's *Standard* character type).

NeXML

NeXML[5] is a file format for exchanging data containing character state data matrices and phylogenies. It is syntax is defined in terms of an XML schema, and the semantics of its elements are defined in terms of CDAO classes. Being defined in this way allows direct translation to CDAO class instances. This guarantee is also important to using it as a medium of exchange since it's semantics can be agreed upon by both the provider and recipient of a dataset.

PhlyoWS

PhyloWS is a standard for exposing phylogenetic data as a webservice, in such a way that particular data items, can be referenced by persistent HTTP URI's.

Implementation

CDAO-store builds on the EvoIO technology stack to provide a framework for supplying semantic services for phylogenetic data services. The platform is open-source and is available on source-forge, at http://sourceforge.net/projects/cdaotools/. It's divided into three main parts. A data importer/file translator, a database and web interface, and a gui visualization tool.
The file importer/translator is implemented in C++ and Python. In addition to its own set of parsers, the translator uses the NCL[6] library to read certain file formats. After reading, it maps data from these files on to an object model that mirrors CDAO classes, and then either converts to some specified format or to an RDF/XML serialization of the data. The import portion of this part of the system is written in Python and uses the RDFlib[7] module to store the RDF serializations produced by the translator into a database making it available to query on the web or by using the visual tools.

[5] http://www.nexml.org
[6] http://sourceforge.net/projects/ncl/
[7] http://www.rdflib.net/

The web and database portion of the application stores, and provides access to the data for the visual tools. This portion of the application is primarily implemented as a set of scripts in a variety of languages. The web interface is divided into two principal parts an HTML user interface, and a PhyloWS data provider. The HTML interface allows for online querying/exploration of datasets, while the PhyloWS interface supplies access to datasets for our visual tools or other third party programs. The database portion of the application is implemented as an RDFlib store running on a MySQL database.
The visual tools are implemented as a Java JNLP application called CDAO-Explorer. It uses a variety of frameworks to support its operation including Pellet[8] and Prefuse[9].
CDAO-Explorer provides a tree and matrix search windows which allow one to search for and load particular datasets, and visualizers for those data sets. It also allows one to make annotations about a dataset, or a general project space, a set of data sets of interest. These annotations can be from CDAO, Dublin-Core, or from a user-supplied source of annotation types.

Results

What should be described here is the functionality of the software together with data on how its performance and functionality compare with and improve on functionally similar existing software.
Things to compare functionality with: Nexplorer and PhiloWidget (about the same functionality with trees, but they dont display matrices and viewing edges as data)

Discussion

Intended use of the software and the benefits that are envisioned together, if possible, with an outline for the planned future development of new features.
Include conversation on MIAPA and OBI?

Conclusions

Text for this section ...

Availability and Requirements

Project name: Project home page: Operating system(s): Programming language: Other requirements: License: Any restriction to use by non-academics:

[8] http://pellet.owldl.com/
[9] http://prefuse.org/

Authors contributions

Text for this section ...

Acknowledgements

Text for this section ...

References

Figures

Figure 1 - Sample figure title

A short description of the figure content should go here.

Figure 2 - Sample figure title

Figure legend text.

Tables

Table 1 - Sample table title

Here is an example of a *small* table in LaTeX using `\tabular{...}`. This is where the description of the table should go.

My Table		
A1	B2	C3
A2	...	..
A3	..	.

Table 2 - Sample table title

Large tables are attached as separate files but should still be described here.

Additional Files

Additional file 1 — Sample additional file title

Additional file descriptions text (including details of how to view the file, if it is in a non-standard format or the file extension). This might refer to a multi-page table or a figure.

Additional file 2 — Sample additional file title

Additional file descriptions text.