Project Summary for PhyloSoC: Interoperable exchange of gene tree reconciliation maps

by Daniel Packer for NESCent/GSOC 2011

Overview

The two goals of this project were to develop an inter-operable exchange of tree reconciliation data, and to implement that exchange in the iPlant Tree of Life TR database. Application of the new NeXML format in the TR viewer was also considered and investigated (see notes below).

We identified a way to standardize inter-operable gene tree reconciliation maps using the NeXML format. A detailed description of our application of NeXML has been documented on the NESCent wiki at:

https://pods.iplantcollaborative.org/wiki/display/iptol/Reconciled+Trees+in+XML

Using NeXML, a platform-agnostic format for phylogenetic data, we have a good foundation for the interchange of gene species tree maps. We worked out the details on how to to store mappings between trees and nodes in a way that is fully supported by existing standard libraries – namely Bio::Phylo and Bio::Perl.

A new library, Bio::TRParse, based on Rutger Vos's Bio::Phylo has been developed and is in early stages to implement this NeXML approach. The library can currently read in and parse a NeXML file and extract most of the data into an internal data structure. The next feature to be added would be generation of NeXML from the internal structure. The parser is object oriented and uses an interface based loosely on Bio::Phylo. The library does not directly support NHX/PRIME legacy formats, but can support it through iPToL's existing import software which has been used for TreeBest output. There is initial support for querying the TR database, but the data isn't utilized by the library at this point, and no ability to save TR data has been implemented.

In order for the Bio::TRParse to have practical applications, more work must yet be done on it (see next steps below). At present, it demonstrates a proof of concept for the beginnings of being able to translate between the new NeXML standard and existing data sources.

Inter-operable Data

This project centers around an application of NeXML, which was developed in cooperation with Rutger Vos, author of NeXML. James Estill wrote the initial draft which formed the basis for the approach and James and I worked with Rutger to figure out some issues around data organization and library support in Bio::Phylo.

In additional to NeXML, there is legacy NHX/PRIME format data from tools such as TreeBest, which was first used for iPToL, PrimeGSR which is currently being used, and Notung which is next on the list to be implemented. These tools use modified versions of NHX to support the annotations for reconciled trees not present in NHX format and so cannot be processed with existing NHX parsers. In order to import TreeBest data, James Estill developed a modified parser.

Existing iPToL data is for one species tree of six species and roughly 2500 gene families. The TR data was generated by TreeBest and imported with James Estill's script and modified BioPerl::TreeIO::NHX module which are located in iPToL SVN.

The Bio::TRParse library

The interface for Bio::TRParse is similar to that of Bio::Phylo, with a 'format' and source specified.

```
use Bio::TRParse;
my $TRobject = Bio::TRParse->new();

$TRobject->load(
     'source' => 'myfile.xml',
     'format' => 'nexml'
);
```

To load from the TR database the call would instead be the following. 'pg00892' refers to the internal ID of the gene family from the imported TreeBest output in the TR database and 'bowers_rosids' refers to the species tree (there is currently only one available).

```
$TRobject->load(
    'format' => 'iplant',
    'source' => ['bowers_rosids', 'pg00892']
);
```

The ultimate goal of Bio::TRParse is to support IO between NeXML, iPToL, NHX/PRIME, and other formats as needed. The skeleton of the internal data is there, as well as initial support for NeXML and iPToL input.

Calls to load() and store() would retrieve and save TR mappings respectively. One could call

load() on a NeXML document, and then store() on the TR database to save the reconciliation. This effectively implements inter-operable exchange between the two data sources. Currently, calls to load() on NeXML files populate an internal data structure. This load support is not yet complete. Support for loading from iPToL TR database is in preliminary stages with working queries implemented.

The data structure returned by the load() call has object oriented encapsulation of reconciliations mappings and trees and their nodes and supports methods for modifying those properties, but is at present not meant to be manipulated except internally. For further documentation, please see the perldoc documentation in my code at:

https://github.com/danielpacker/TreeRecXML/tree/master/parser/TRParse

Notes on the iPToL TR Viewer

The iPlant TR viewer is a Java web application that uses the Phyloviewer tree display component to display TR mappings from the TR database. The TR viewer and database are two separate applications, and the phyloviewer component is a separate project with different contributors. The TR database is comprised of a mysql schema and data set, and Perl modules and scripts to implement queries and JSON marshaling for the TR viewer client. James Estill is the primary author of the TR database.

Using the Bio::TRParse module, it was suggested that the TR viewer might be able to use NeXML, legacy, or other data sources directly, rather than having to read the TR database via JSON calls. Currently, the TR viewer operates exlusively via TR database JSON calls.

Github for TR viewer: https://github.com/iPlantCollaborativeOpenSource/tr-standalone

Notes: The TR viewer calls the TR database JSON services in classes located in this directory:

src/main/java/org/iplantc/tr/demo/client/panels/

Github for TR database: https://github.com/iPlantCollaborativeOpenSource/iplant-treerec

Notes: James Estill is the primary technical contact for the TR database and can be reached at jamesestill@gmail.com.

The TR database services the TR viewer JSON calls via the following perl module:

lib/IPlant/TreeRec.pm

Github for Phyloviewer: https://github.com/iPlantCollaborativeOpenSource/iPlant_phyloviewer

Notes: Kristopher Urie is the current primary technical contact for the Phyloviewer module and can be reached at kurie@fieldmuseum.org. The phyloviewer project has little overlap with the TR viewer project in terms of code, except that the TR Viewer embeds the phyloviewer component as a dumb display method.

Next Steps for Bio::TRParse

- Complete NeXML input, and implement NeXML output
- Complete iPToL TR database input and implement TR DB output
- Implement legacy (NHX/PRIME/etc) input and output support
- Hard code test data in NeXML, NHX/PRIME, and TR database based on contrived gene/species tree demo examples
    - Choose 2-3 gene families to focus on that exhibit various levels of complexity (i.e. one duplication, then two, three...)
    - This will allow testing various levels of complexity and multiple reconciled host trees.
- Write a tests to compare input from NeXML, iPToL TR database, and legacy files
    - We should at least compare saved NeXML imports to imported TreeBest reconciliations to make sure they are equivalent using the existing TreeBest data.
- Improve the documentation and distribute to the community for use