Bioinformatics, 33(18), 2017, 2946–2947
doi: 10.1093/bioinformatics/btx324
Advance Access Publication Date: 19 May 2017
Applications Note

Phylogenetics

PhyD3: a phylogenetic tree viewer with extended phyloXML support for functional genomics data visualization

Łukasz Kreft[1], Alexander Botzki[1], Frederik Coppens[2,3], Klaas Vandepoele[2,3,4,]* and Michiel Van Bel[2,3,]*

[1]VIB Bioinformatics Core, VIB, 9052 Ghent, Belgium, [2]Department of Plant Biotechnology and Bioinformatics, Ghent University, 9052 Ghent, Belgium, [3]VIB-UGent Center for Plant Systems Biology, 9052 Ghent, Belgium and [4]Bioinformatics Institute Ghent, Ghent University, 9052 Ghent, Belgium

*To whom correspondence should be addressed.

Associate Editor: Janet Kelso

Received on December 7, 2016; revised on April 5, 2017; editorial decision on May 15, 2017; accepted on May 18, 2017

Abstract

Motivation: Comparative and evolutionary studies utilize phylogenetic trees to analyze and visualize biological data. Recently, several web-based tools for the display, manipulation and annotation of phylogenetic trees, such as iTOL and Evolview, have released updates to be compatible with the latest web technologies. While those web tools operate an open server access model with a multitude of registered users, a feature-rich open source solution using current web technologies is not available.

Results: Here, we present an extension of the widely used PhyloXML standard with several new options to accommodate functional genomics or annotation datasets for advanced visualization. Furthermore, PhyD3 has been developed as a lightweight tool using the JavaScript library D3.js to achieve a state-of-the-art phylogenetic tree visualization in the web browser, with support for advanced annotations. The current implementation is open source, easily adaptable and easy to implement in third parties' web sites.

Availability and implementation: More information about PhyD3 itself, installation procedures and implementation links are available at http://phyd3.bits.vib.be and at http://github.com/vibbits/phyd3/.

Contact: klaas.vandepoele@ugent.vib.be or michiel.vanbel@ugent.vib.be

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The construction, visualization and interpretation of phylogenetic trees are instrumental for biological analysis in multiple fields, including evolutionary biology, genetics and comparative genomics. Recently published software tools allow for the use of interactive elements and comprehensive analytics in association with these trees by means of the latest web technologies (He *et al.*, 2016; Kuraku *et al.*, 2013; Letunic and Bork, 2016; Smits and Ouverney, 2010). The phyloXML standard (Han and Zmasek, 2009) is widely supported in those tools to visualize the evolutionary trees as well as associated data.

A comparison of the most popular phylogenetic tree viewers reveals that a feature rich open source tool that (i) supports extended charting, (ii) is easily adaptable and (iii) can be seamlessly implemented in third party websites, is currently not available (see Supplementary Data). To this end, we extended the phyloXML standard with several new elements in order to accommodate common annotation datasets and developed a lightweight phylogenetic tree viewer using the D3.js JavaScript library (http://d3js.org).

2 Materials and methods

The commonly used phyloXML standard has been extended with various elements of the complex type as permitted by the current phyloXML XSD scheme. The newly created elements <taxonomies> and <domains> are used to specify taxonomy and domain colours, descriptions and links. In order to reference between the <phylogeny> element of the current standard and the new elements, the <code> and <name> sub-elements are used for the <taxonomy> and the <domain> element, respectively. Colours are represented as HEX values (e.g. 0xRRGGBB). Furthermore, graphs like a pie, a binary and a multi-bar chart as well as a heat map or a boxplot can be displayed next to the leaf nodes by using the newly defined <graph> elements.

The pie chart and binary chart types can also be associated with internal nodes of the phylogenetic tree. Referencing of the chart values to the nodes is achieved by defining an <id> tag in the <clade> element corresponding to the '*for*' attribute in <value> elements.

A complete graph specification consists of the following child elements:

1. name: This element contains the graph name that will be displayed in a detailed node information popup.
2. legend: For each value series of your data a legend field has to be specified. This legend consists of the field name (which will be displayed to the user in the legend and info popup), the colour that will be used to draw this series (for all graph types besides heat map) and an additional symbol shape (e.g. circle) for the binary graph. For the heat map a gradient specification has to be applied according to the specifications of the ColorBrewer2 project (http://colorbrewer2.org).
3. data: Data values for the graph have to be defined for each clade id for which a graph is displayed. They have to be sorted according to how the series are defined in the respective legend fields. Existing tags or properties from the phylogeny element can be used by referencing them via the '*tag*' and '*ref*' attributes of the data element.

3 Features

Using the D3.js JavaScript library, PhyD3 is implemented as a flexible and lightweight tool allowing for the display of interactive and complex phylogenetic trees in a web-based environment without security-based limitations and without the need for external plugins. The implementation is fast and responsive to user interaction with the tree elements' display parameters being easily changed through user-friendly access controls (Fig. 1).

PhyD3 offers a number of key features with respect to tree nodes and graphs visualization. Basic information (like node and taxonomy names, branch length and support values) can be displayed next to the respective nodes with detailed information (including link outs to external URLs) being shown in an info box per user request. The display of the information is flexible according to the users' preferences (e.g. all content, or only internal or external node annotations). To support a clearly annotated data presentation and better readability, leaf nodes can be lined up to match the furthest node and text can be scaled.

Furthermore, nodes can be annotated with structural and numerical information, displayed in the form of various graphs: protein domain information or read-count data can for example easily be visualized next to each node. PhyD3 provides controls to scale and filter this information to support easy analysis. Numerical annotations are presented using multiple bar charts, pie charts, binary charts or boxplot charts. Users can easily change the size and scale of graphs to match their needs.

Additionally, a number of convenience features were introduced in PhyD3: tree nodes are automatically hidden when they do not fit in the space between nodes to prevent overlapping of text and graphs; tree colours can be inverted for greater readability; node texts can be coloured according to node taxonomy. Extra display options include the capabilities to swap tree nodes within a sub-tree as well colouring options for these sub-trees.

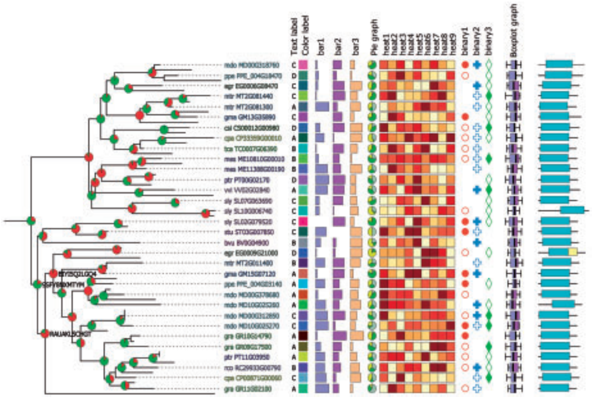

Fig. 1. PhyD3 visualization of artificial data using inner node pie charts, taxonomy colorization, leaf node labels, and leaf charts (multi-bar, pie, heatmap, binary, and boxplot) and domain architecture

Lastly, PhyD3 provides import and export tools to facilitate greater interoperability. Using the import tool users can supply trees in Newick and phyloXML formats, with optional numerical data, which can be easily converted to the extended phyloXML format with graph annotations. Export capabilities provide the conversion of the current tree visualization into vector graphics (SVG format) and bitmaps (PNG format), as well as the extended phyloXML data itself. A comparative overview of the main features of most popular phylogenetic tree viewers is given in Supplementary Table S1. A full feature list is described in Supplementary Table S2 (Fig. 1).

4 Conclusions and further information

We have developed an extension of the phyloXML standard to include graph specifications and datasets as extensions of the XSD scheme as well as a phylogenetic tree viewer PhyD3, using the JavaScript D3.js library. Thanks to the phyloXML extensions, the PhyD3 viewer efficiently visualizes extensively annotated phylogenetic tree data using modern web browsers. The software, full documentation and demo material are available at http://phyd3.bits.vib.be and http://github.com/vibbits/phyd3.

Conflict of Interest: none declared.

References

Han,M.V. and Zmasek,C.M. (2009) phyloXML: XML for evolutionary biology and comparative genomics. *BMC Bioinformatics*, **10**, 356.

He,Z. *et al*. (2016) Evolview v2: an online visualization and management tool for customized and annotated phylogenetic trees. *Nucleic Acids Res*., **44**, W236–W241.

Kuraku,S. *et al*. (2013) aLeaves facilitates on-demand exploration of metazoan gene family trees on MAFFT sequence alignment server with enhanced interactivity. *Nucleic Acids Res*., **41**, W22–W28.

Letunic,I. and Bork,P. (2016) Interactive tree of life (iTOL) v3: an online tool for the display and annotation of phylogenetic and other trees. *Nucleic Acids Res*., **44**, W242–W245.

Smits,S.A. *et al*. (2010) jsPhyloSVG: a javascript library for visualizing interactive and vector-based phylogenetic trees on the web. *Plos One*, **5**, e12267.