**JPhyloIO** — A Java library for event-based reading and writing of different alignment and tree formats through one common interface

**Aims and concept**

To date many bioinformatic software tools support only a single format. Applications based on JPhyloIO need to implement just one single reader and writer to support all formats without needing detailed knowledge of these. This should increase interoperability and foster the usage of more recently proposed powerful formats, such as NeXML.

Our library allows to access nine phylogenetic file formats through one common interface, providing access to all features of each format (including complex metadata of NeXML and PhyloXML). Documents are translated to a stream of events (see figure 1 and 2), allowing memory efficient processing independent of the application business model.

**Supported formats**

- NeXML (Vos et al., 2012)
- Nexus (Maddison et al., 1997)
- PhyloXML (Han & Zmasek, 2009)
- FASTA
- Newick tree format
- Phylib and extended Phylip (also sequential)
- MEGA (Kumar et al., 2016)
- PDE used by the alignment editor PhyDE
- XTG used by TreeGraph 2 (Stöver & Müller, 2010)

**Availability**

JPhyloIO is distributed under GNU General Public License Version 3 at the BioinfWeb Software portal: [http://bioinfweb.info/JPhyloIO](http://bioinfweb.info/JPhyloIO)

**Event based document reading**

In JPhyloIO documents are represented as sequences of events that model the contained elements. Figure 1 shows the grammar that defines how documents can be represented as event sequences, while figure 2 shows an example document and its translation into events. All elements can carry metadata.

**Writing events using data adapters**

Since different formats require the data in different orders, a simple event stream is not efficient for writing. A number of data adapters have been defined instead, each of them providing a subsequence of the event stream.

---

**Citations:**


**Acknowledgements**

The funding of parts of the development of JPhyloIO with grant BI 287c/1 to KPM by the DFG (German research foundation) is highly appreciated. BCS wants to thank the European Conference on Computational Biology (ECCB) and the International Society for Computational Biology (ISCB) for partly financing the poster at the ECCB 2016. Furthermore the authors are very thankful to the developers of the other open source projects (PhyloXML uses Apache commons, CML, API, Jxml, Jexpress).

---

**Figure 1** A EBNF grammar describing JPhyloIO event sequences. The terminal symbols (in green) represent the types of events, each of which either has a single SOLE or a START and END version, depending on whether additional data can be tested or not.

**Figure 2** The shown example document contains an OTU list and an alignment, which references this list. The shown sequence of events is generated from it according to the grammar in figure 1, where each box represents one event object. Each object has an ID in order to be referenced by subsequent events, as exemplarily shown on the OTU list and OTU start events, which are referenced by the according alignment and sequence start events.

**Figure 3** The data adapters of (PhyloXML) are implemented by an application to provide access to its business model.

**Figure 4** An UML class diagram showing the inheritance and relations between the different data adapter interfaces of (PhyloXML). DataAdapter is the start point providing access to the others.