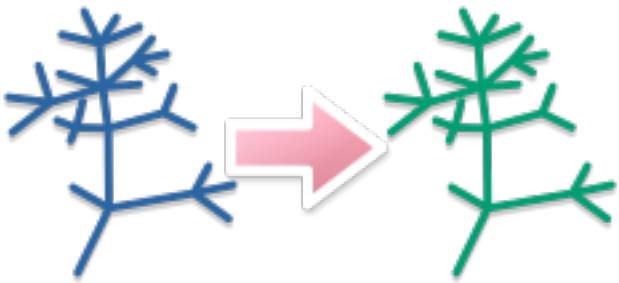
Parse tip states

Version 1.0.0, by Giorgio Bianchini



Description: Loads tip state data from an attachment.

Module type: FurtherTransformation

Module ID: `716b55a3-02d9-4007-a830-8326d407b24c`

This module can be used to parse attributes for the tips of the tree from a separate file loaded as an attachment.

Parameters

Data file

Control type: Attachment

Lines to skip

Control type: Number spin box

Default value: 0

Range: [0, +∞)

This parameter determines the lines to skip at the start of the file (useful e.g. if the data file contains header lines).

Separator

Control type: Text box

Default value: `\s`

This parameter contains the separator used to split the lines of the data file. If the Regex checkbox is checked, regex escape characters can be used. These include:

- `\t` matches a tabulation
- `\s` matches a whitespace character (e.g. a space or a tabulation)

Note that, since empty elements are discarded anyways, it is not a problem if multiple instances of the separator occur in sequence (e.g. `A  B` is parsed just as well as `A B`).

Regex

Control type: Check box

Default value: Checked

If this check box is checked, the separator is matched using a regular expression. This makes it possible e.g. to use escape characters or to perform advanced matching (for example, if this option is active, a separator of `[\s,;]` could be used to parse a file in which the states are separated by spaces, commas and/or semicolons).

Attribute(s)

Control type: Text box

Default value: `State`

This parameter determines the name of the attribute in which the parsed states are stored. If more than one attribute should be parsed from the file, the value of this parameter can be set to a string that will be split based on the same separator that is used for the data (e.g. if the separator is `;` and the attributes to be parsed are called `State1` and `State2`, a possible value for this parameter could be `State1;State2`).

Attribute type

Control type: Attribute type

Default value: String

Possible values:

- String
- Number

This parameter determines the type of the attribute that is parsed. If this is `String`, the attribute is stored as a string, even if the contents represent a number (e.g. the number `1` would be stored as the string `"1"`). Depending on how you intend to analyse the data, this may or may not be your intended behaviour - e.g. if the attribute represents a discrete character state, it is appropriate to parse it as a string; if it is instead a continuous character state, parsing it as a number may be more appropriate.

Apply

Control type: Button

This button applies the changes to the other parameters and signals to the downstream modules that the tree should be redrawn.

Further information

This module can be used to read "complex" attributes from a text file. If you just wish to load "presence-absence" data, the *Add attribute* module may also be suited to your needs.

This module reads each line of the text file and splits it using the selected separator. The first item of each line should be the taxon name on which the state is going to be applied. Other items represent the attributes that will be attached to that taxon.

More than one attribute can be parsed at once; the attributes that are loaded are specified by the Attribute(s) parameter.

For example, if the Separator is `\s` (i.e. whitespace) and the Attribute(s) are `State1 State2`, the following file would assign a `State1` of `A` and a `State2` of `5` to the taxon named `Nostoc`, and a `State1` of `B` and a `State2` of `3` to the taxon named `Synechococcus`:

```
Nostoc          A   5
Synechococcus   B   3
```