Parse node states

Version 2.1.3, by Giorgio Bianchini

Description: Loads node 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 nodes of the tree from a separate file loaded as an attachment.

Parameters

Data file

Control type: Attachment

This parameter is used to select the attachment that contains the data file to parse.

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). Note that if you want to use the header row to specify the column headers, that row must NOT be skipped.

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).

Match column

Control type: Number spin box

Default value: 1

Range: [1, +∞)

This parameter determines the column that contains the values that are matched against the match attribute.

Match attribute

Control type: Attribute selector

Default value: Name

This parameter determines the attribute that is matched against the values in the match column.

Match attribute type

Control type: Attribute type

Default value: String

Possible values:

- String
- Number

This parameter determines the type of the match attribute.

Use first row as header

Control type: Check box

Default value: Unchecked

If this check box is checked, the first row in the data file is assumed to be a header row, containing the names of the attribute(s) where the parsed states are stored. If this check box is unchecked, the column headers need to be specified manually in the column header(s) parameter.

Column header(s)

Control type: Text box

Default value: `State`

This parameter should contain the headers for the columns of data in the file. These will also be used as the names of the attribute(s) where the parsed states are stored. The names of different columns should be separated using 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`).

If the match column is `1`, the header for the first column can be omitted. Otherwise, headers should be provided for all columns; columns in the data without a corresponding header will be ignored.

Attribute type

Control type: Drop-down list

Default value: Auto

Possible values:

- String
- Number
- Auto

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.

If the selected value is `String` or `Number`, this will be applied to all the columns in the data. Instead, if the selected value is `Auto`, each attribute will be assessed independently to determine whether it can be represented as a String or a Number.

Preview/Apply

Control type: Buttons

Buttons:

- Preview
- Apply

The `Preview` button shows a preview of the data that will be parsed from the selected attachment file. The `Apply` 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. You can choose which column of the file contains the value that will be matched against the specified attribute of the nodes. Other columns represent the attributes that will be attached to the matched taxon.

More than one attribute can be parsed at once; if a header row is present, this can be used to specify the names of the attributes that are loaded. Alternatively, the attributes that are loaded can be specified by the column header(s) parameter. These should correspond to the headers of the columns in the file. If the match column is `1`, the header for the first column (which is matched against the match attribute) can be skipped. Otherwise, a header must be provided for the match column, even though it is not used.

For example, assume that:

- the separator is `\s` (i.e. whitespace)
- the match column is `1`
- the match attribute is `Name` (and the match attribute type is `String`)
- the column header(s) are `State1 State2`

In this case, 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
```

Instead, consider the case in which:

- the separator is `,` (i.e. a comma)
- the match column is `2`
- the match attribute is `Name` (and the match attribute type is `String`)
- the column header(s) are `State1,Genus,State2`

Here, 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`:

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

The `Preview` button can be used to display a preview of how the data in the file will be parsed based on the current settings.