NAME

    xmltree – DOM-like facilities for gawk-xml. Its status is **experimental**. May change in the future.

SYNOPSIS

```
@include "xmltree"

XmlPrintElementStart(index)
XmlPrintElementEnd(index)
XmlPrintNodeText(index)

XmlPrintNodeTree(index)

n = XmlGetNodes(rootnode, path, nodeset)
value = XmlGetValue(rootnode, path)
```

DESCRIPTION

    The *xmltree* awk library adds DOM-like facilities to the *gawk-xml* extension.

Automatic storage of the element tree

    The *xmlbase* library contains rules that automatically store the document's element tree in memory. The tree contains a node for each:

- Element

- Attribute

- Text content fragment

Each node in the tree can be referenced by an integer node index. The root element node has an index of 1. Nodes are stored in lexicographical order.

Processing the tree in the END clause

    The stored tree is not fully available until the end of the input file. The intended way of using the tree is to put all the processing code in the **END** clause.

Printing tree fragments

    XmlPrintElementStart(index)

        Prints the element's start tag, including the attributes. The index argument must point to an element node.

    XmlPrintElementEnd(index)

        Prints the element's end tag. The index argument must point to an element node.

    XmlPrintNodeText(index)

        Prints the text content of the node. The index argument must point to an attribute or text fragment node.

Selecting tree fragments

    The *xmltree* library provides an XPath-like facility for querying or navigating the document tree.

    n = XmlGetNodes(rootnode, path, nodeset)

        Populates de *nodeset* integer array argument with the indexes of the nodes selected from the starting *rootnode* by the given *path* pattern. Returns the number of selected nodes.

    value = XmlGetValue(rootnode, path)

        Returns the text content of the set of nodes selected from the starting *rootnode* by the given *path* pattern. The content depends on the node kind:

        Attribute node

            The content is the attribute value.

        Text fragment node

            The content is the text fragment.

Element node
> Concatenates the content of the descendant element and text fragment nodes. Attributes are excluded from the result.

The path expression language

path   A relative path from one node to one of its descendants is denoted by a sequence of slash separated labels. The label of a child element is the element name. The label of an attribute node is the attribute name prefixed by the "@" sign. The label of a text content node is the string "#text". The path from one node to itself is an empty path. Examples: `book/title`, `recipe/ingredient/@calories`, `book/author/#text`.

path pattern
> A sequence of selection steps `selector!condition!selector!condition....` Each step is a pair of contiguous "!" delimited fields of the expression.

selector
> Regular expression that will be matched against relative paths between nodes.

condition
> Like selectors, and may also have a trailing "/?" prefixed value pattern, also given as a regular expression.

selection step
> A selection step selects descendant-or-self nodes whose relative path matches the selector, and in turn have some descendant-or-self node whose relative path and text content match the condition.

Examples:

`book!` --> selects all books.
`book!author` --> selects all books that have an author.
`book!author/?Kipling` --> selects all books written by Kipling.
`book!@onloan` --> selects all books that are loaned.
`book!@onloan!title!` --> selects the titles of all books that are loaned.

NOTES
The *xmltree* library includes both the *xmlbase* and the *xmlwrite* libraries. Their functionality is implicitly available.

BUGS
Currently only one XML input document is supported. And the stored node tree should not be modified.

The selection facility can only be used for descendants of a root node. Selectors for ascendant or sibling nodes are not supported.

SEE ALSO
XML Processing With gawk, *xmlbase*(3am), *xmlcopy*(3am), *xmlsimple*(3am), *xmlwrite*(3am).

AUTHOR
Manuel Collado, **m-collado@users.sourceforge.net**.

COPYING PERMISSIONS