

# Package ‘xslt’

May 17, 2020

**Version** 1.4.2

**Title** Extensible Style-Sheet Language Transformations

**Description** An extension for the 'xml2' package to transform XML documents by applying an 'xslt' style-sheet.

**Maintainer** Jeroen Ooms <jeroen@berkeley.edu>

**URL** <https://docs.ropensci.org/xslt> (website)  
<https://github.com/ropensci/xslt> (devel)

**BugReports** <https://github.com/ropensci/xslt/issues>

**Depends** xml2 (>= 1.3.0)

**Imports** Rcpp

**LinkingTo** Rcpp, xml2

**SystemRequirements** libxslt: libxslt1-dev (deb), libxslt-devel (rpm)

**License** GPL (>= 2)

**RoxygenNote** 6.0.1

**Suggests** testthat

**NeedsCompilation** yes

**Author** Jeroen Ooms [aut, cre] (<<https://orcid.org/0000-0002-4035-0289>>)

**Repository** CRAN

**Date/Publication** 2020-05-17 05:00:02 UTC

## R topics documented:

xslt . . . . .	2
<b>Index</b>	<b>3</b>

**Description**

Transform an XML document by applying an XSL stylesheet. Usually returns the transformed [xml\\_document](#), unless the stylesheet has `<xsl:output method="text">` in which case we return a text string.

**Usage**

```
xml_xslt(doc, stylesheet, params)
```

```
xslt_version()
```

**Arguments**

doc	xml document as returned by <a href="#">xml2::read_xml</a>
stylesheet	another xml document containing the XSL stylesheet
params	named list or vector with additional XSLT parameters

**Details**

This implementation supports XSLT 1.0 features plus most of the EXSLT set of processor-portable extensions functions. Unfortunately XSLT 2.0 or 3.0 features are only available in proprietary libraries and currently unsupported. However XSLT 2.0 is not widely adopted anyway because it is unavailable in most browsers.

**Examples**

```
doc <- read_xml(system.file("examples/cd_catalog.xml", package = "xslt"))
style <- read_xml(system.file("examples/cd_catalog.xsl", package = "xslt"))
html <- xml_xslt(doc, style)
cat(as.character(html))
```

# Index

`xml2::read_xml`, [2](#)  
`xml_document`, [2](#)  
`xml_xslt(xslt)`, [2](#)  
`xslt`, [2](#)  
`xslt_version(xslt)`, [2](#)