NAME

xmllint − command line XML tool

SYNOPSIS

xmllint [**−−version** | **−−debug** | **−−shell** | **−−xpath** "*XPath_expression*" | **−−debugent** | **−−copy** | **−−recover** | **−−noent** | **−−noout** | **−−nonet** | **−−path** "*PATH(S)*" | **−−load−trace** | **−−htmlout** | **−−nowrap** | **−−valid** | **−−postvalid** | **−−dtdvalid** *URL* | **−−dtdvalidfpi** *FPI* | **−−timing** | **−−output** *FILE* | **−−repeat** | **−−insert** | **−−compress** | **−−html** | **−−xmlout** | **−−push** | **−−memory** | **−−maxmem** *NBBYTES* | **−−nowarning** | **−−noblanks** | **−−nocdata** | **−−format** | **−−encode** *ENCODING* | **−−dropdtd** | **−−nsclean** | **−−testIO** | **−−catalogs** | **−−nocatalogs** | **−−auto** | **−−xinclude** | **−−noxincludenode** | **−−loaddtd** | **−−dtdattr** | **−−stream** | **−−walker** | **−−pattern** *PATTERNVALUE* | **−−chkregister** | **−−relaxng** *SCHEMA* | **−−schema** *SCHEMA* | **−−c14n**] {*XML−FILE(S)...* | −}

xmllint −−help

DESCRIPTION

The **xmllint** program parses one or more XML files, specified on the command line as *XML−FILE* (or the standard input if the filename provided is **−**). It prints various types of output, depending upon the options selected. It is useful for detecting errors both in XML code and in the XML parser itself.

xmllint is included in **libxml**(3).

OPTIONS

xmllint accepts the following options (in alphabetical order):

−−auto
Generate a small document for testing purposes.

−−catalogs
Use the SGML catalog(s) from **SGML_CATALOG_FILES**. Otherwise XML catalogs starting from /etc/xml/catalog are used by default.

−−chkregister
Turn on node registration. Useful for developers testing **libxml**(3) node tracking code.

−−compress
Turn on **gzip**(1) compression of output.

−−copy
Test the internal copy implementation.

−−c14n
Use the W3C XML Canonicalisation (C14N) to serialize the result of parsing to stdout. It keeps comments in the result.

−−dtdvalid *URL*
Use the DTD specified by an *URL* for validation.

−−dtdvalidfpi *FPI*
Use the DTD specified by a Formal Public Identifier *FPI* for validation, note that this will require a catalog exporting that Formal Public Identifier to work.

−−debug
Parse a file and output an annotated tree of the in−memory version of the document.

−−debugent
Debug the entities defined in the document.

−−dropdtd
Remove DTD from output.

−−dtdattr
Fetch external DTD and populate the tree with inherited attributes.

−−encode *ENCODING*

Output in the given encoding. Note that this works for full document not fragments or result from XPath queries.

−−format

Reformat and reindent the output. The **XMLLINT_INDENT** environment variable controls the indentation. The default value is two spaces " ").

−−help

Print out a short usage summary for **xmllint**.

−−html

Use the HTML parser.

−−htmlout

Output results as an HTML file. This causes **xmllint** to output the necessary HTML tags surrounding the result tree output so the results can be displayed/viewed in a browser.

−−insert

Test for valid insertions.

−−loaddtd

Fetch an external DTD.

−−load−trace

Display all the documents loaded during the processing to stderr.

−−maxmem *NNBYTES*

Test the parser memory support. *NNBYTES* is the maximum number of bytes the library is allowed to allocate. This can also be used to make sure batch processing of XML files will not exhaust the virtual memory of the server running them.

−−memory

Parse from memory.

−−noblanks

Drop ignorable blank spaces.

−−nocatalogs

Do not use any catalogs.

−−nocdata

Substitute CDATA section by equivalent text nodes.

−−noent

Substitute entity values for entity references. By default, **xmllint** leaves entity references in place.

−−nonet

Do not use the Internet to fetch DTDs or entities.

−−noout

Suppress output. By default, **xmllint** outputs the result tree.

−−nowarning

Do not emit warnings from the parser and/or validator.

−−nowrap

Do not output HTML doc wrapper.

−−noxincludenode

Do XInclude processing but do not generate XInclude start and end nodes.

−−nsclean

Remove redundant namespace declarations.

−−output *FILE*

Define a file path where **xmllint** will save the result of parsing. Usually the programs build a tree and save it on stdout, with this option the result XML instance will be saved onto a file.

−−path "*PATH(S)***"**
Use the (space− or colon−separated) list of filesystem paths specified by *PATHS* to load DTDs or entities. Enclose space−separated lists by quotation marks.

−−pattern *PATTERNVALUE*
Used to exercise the pattern recognition engine, which can be used with the reader interface to the parser. It allows to select some nodes in the document based on an XPath (subset) expression. Used for debugging.

−−postvalid
Validate after parsing has completed.

−−push
Use the push mode of the parser.

−−recover
Output any parsable portions of an invalid document.

−−relaxng *SCHEMA*
Use RelaxNG file named *SCHEMA* for validation.

−−repeat
Repeat 100 times, for timing or profiling.

−−schema *SCHEMA*
Use a W3C XML Schema file named *SCHEMA* for validation.

−−shell
Run a navigating shell. Details on available commands in shell mode are below (see the section called “SHELL COMMANDS”).

−−xpath "*XPath_expression***"**
Run an XPath expression given as argument and print the result. In case of a nodeset result, each node in the node set is serialized in full in the output. In case of an empty node set the "XPath set is empty" result will be shown and an error exit code will be returned.

−−stream
Use streaming API − useful when used in combination with **−−relaxng** or **−−valid** options for validation of files that are too large to be held in memory.

−−testIO
Test user input/output support.

−−timing
Output information about the time it takes **xmllint** to perform the various steps.

−−valid
Determine if the document is a valid instance of the included Document Type Definition (DTD). A DTD to be validated against also can be specified at the command line using the **−−dtdvalid** option. By default, **xmllint** also checks to determine if the document is well−formed.

−−version
Display the version of **libxml**(3) used.

−−walker
Test the walker module, which is a reader interface but for a document tree, instead of using the reader API on an unparsed document it works on an existing in−memory tree. Used for debugging.

−−xinclude
Do XInclude processing.

−−xmlout

Used in conjunction with **−−html**. Usually when HTML is parsed the document is saved with the HTML serializer. But with this option the resulting document is saved with the XML serializer. This is primarily used to generate XHTML from HTML input.

SHELL COMMANDS

xmllint offers an interactive shell mode invoked with the **−−shell** command. Available commands in shell mode include (in alphabetical order):

base
Display XML base of the node.

bye
Leave the shell.

cat *NODE*
Display the given node or the current one.

cd *PATH*
Change the current node to the given path (if unique) or root if no argument is given.

dir *PATH*
Dumps information about the node (namespace, attributes, content).

du *PATH*
Show the structure of the subtree under the given path or the current node.

exit
Leave the shell.

help
Show this help.

free
Display memory usage.

load *FILENAME*
Load a new document with the given filename.

ls *PATH*
List contents of the given path or the current directory.

pwd
Display the path to the current node.

quit
Leave the shell.

save *FILENAME*
Save the current document to the given filename or to the original name.

validate
Check the document for errors.

write *FILENAME*
Write the current node to the given filename.

ENVIRONMENT

SGML_CATALOG_FILES
SGML catalog behavior can be changed by redirecting queries to the user's own set of catalogs. This can be done by setting the **SGML_CATALOG_FILES** environment variable to a list of catalogs. An empty one should deactivate loading the default /etc/sgml/catalog catalog.

XML_CATALOG_FILES
XML catalog behavior can be changed by redirecting queries to the user's own set of catalogs. This can be done by setting the **XML_CATALOG_FILES** environment variable to a space−separated list of catalogs. Use percent−encoding to escape spaces or other characters. An empty variable should

deactivate loading the default /etc/xml/catalog catalog.

XML_DEBUG_CATALOG
Setting the environment variable **XML_DEBUG_CATALOG** to *non−zero* using the **export** command outputs debugging information related to catalog operations.

XMLLINT_INDENT
Setting the environment variable **XMLLINT_INDENT** controls the indentation. The default value is two spaces " ".

DIAGNOSTICS

xmllint return codes provide information that can be used when calling it from scripts.

0
No error

1
Unclassified

2
Error in DTD

3
Validation error

4
Validation error

5
Error in schema compilation

6
Error writing output

7
Error in pattern (generated when **−−pattern** option is used)

8
Error in Reader registration (generated when **−−chkregister** option is used)

9
Out of memory error

10
XPath evaluation error

SEE ALSO

libxml(3)

More information can be found at

- **libxml**(3) web page **https://gitlab.gnome.org/GNOME/libxml2**

AUTHORS

John Fleck <jfleck@inkstain.net>
Author.

Ziying Sherwin <sherwin@nlm.nih.gov>
Author.

Heiko Rupp <hwr@pilhuhn.de>
Author.

COPYRIGHT