

The KXSLDbg Handbook

Keith Isdale



The KXSLDbg Handbook

Contents

1	Introduction	5
1.1	Features	5
2	Using KXSLDbg	6
2.1	Configuring a KXSLDbg Session	6
2.1.1	Getting Started	6
2.1.2	Working With Options	7
2.1.3	Working With Parameters	7
2.1.4	Apply Changes	7
2.2	Using the Main Window	8
2.2.1	Working With the Main Window	8
2.2.2	Working With KXSLDbg Output	8
2.3	Setting and Modifying Breakpoints	8
2.3.1	Adding a Breakpoint	9
2.3.2	Argument Details	9
2.3.3	Deleting a Breakpoint	9
2.3.4	Deleting All Breakpoints	9
2.3.5	Enabling or Disabling a Breakpoint	10
2.4	Working With Variables	10
2.5	Working With the Callstack	11
2.6	Working With Templates	11
2.7	Working With XSLT Source Files (Sources)	12
2.8	Working With XML Data Files (Entities)	13
2.9	Miscellaneous	13
2.9.1	Inspector Tool	13
2.9.2	Execute by Walking	14
2.9.3	Lookup XML Entities	14
3	Credits and License	15

Abstract

KXSLDbg provides a graphic user interface front-end to xsldbg, which supports debugging of XSLT scripts.

Chapter 1

Introduction

1.1 Features

KXSLDbg provides access to most of xsldbg's commands to

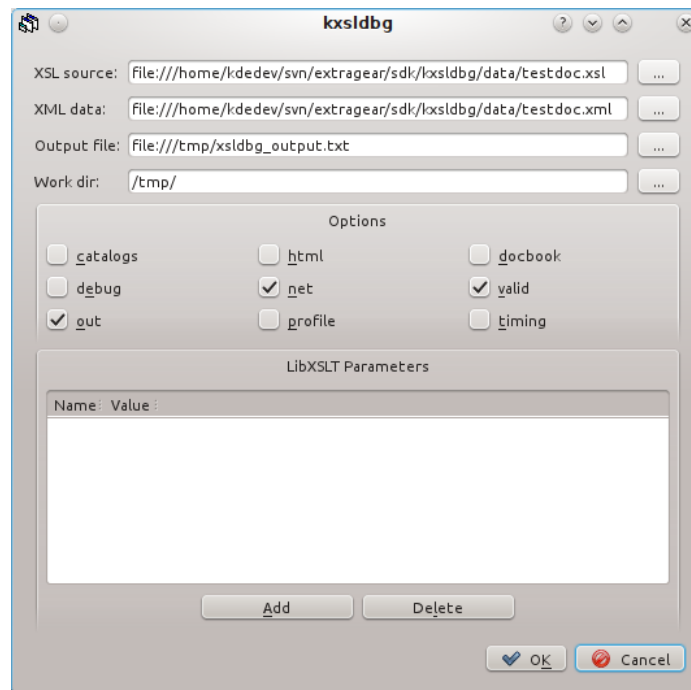
- Set and modify breakpoints
- Display value of XPath's
- Display information about the templates, variables, callstack entries, stylesheets and entities present
- Set and modify breakpoints and variables
- Move around XSL source and XML document via XPath's
- Lookup PUBLIC and SYSTEM ID's in the current XML catalog
- Set and modify variables
- Renders the text in the main window using the Kate libraries

Chapter 2

Using KXSLDbg

2.1 Configuring a KXSLDbg Session

You start configuration by clicking **Debug** → **Configure** in the Menubar.



The Configuration Dialog.

2.1.1 Getting Started

To be able to run a stylesheet you need to specify the:

- XSL source
- XML data
- Output file

The KXSLDbg Handbook

By using the ... button to choose file desired. The **XSL source** and **XML data** may refer to URI that contains a `http://`, `ftp://` or `file://`. The **Output file** must refer to a writable local file.

To follow along with the examples, select the following files in the example `<KDE_PREFIX>/share/apps/kxsldbg` folder

- XSL source: `testdoc.xsl`
- XML data: `testdoc.xml`
- Output file: `/tmp/xsldbg_output.txt`

2.1.2 Working With Options

You can select zero or more options from the **Options** section. Each option has a tooltip with a hint on what effect it has.

2.1.3 Working With Parameters

You can add zero or more parameters via the **LibXSLT Parameters** section of dialog. This allows you to provide parameter values to the stylesheet.

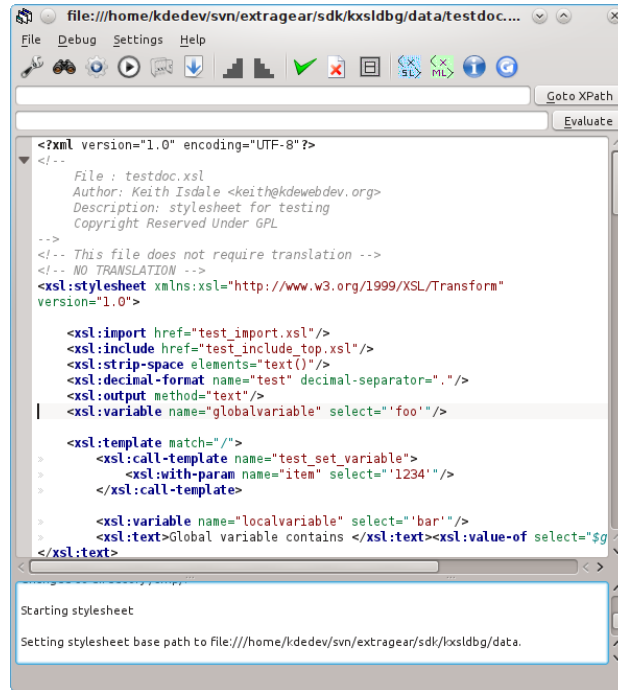
For example you could click the **Add** button and enter a parameter **Name** of `myparam` with a parameter **Value** of `'Hello World!'`. To update the name or value of an existing parameter select it with the left mouse button.

2.1.4 Apply Changes

For the changes you have made to take effect press the **OK** button and close the dialog. To ignore any changes press the **Cancel** button.

If you need to change the configuration just re-open the configuration dialog as shown before.

2.2 Using the Main Window



A text view of the current file being debugged.

2.2.1 Working With the Main Window

The state of a given breakpoint is indicated via the relevant text with a different background color. You can choose the color desired: see the **Settings** → **Configure Editor** dialog, on the **Colors** page.

You can set, disable or delete a breakpoint using keys, the **Debug** menu or the buttons on the tool bar.

You can move the cursor around the text using the following keys:

Arrow keys: **Left Arrow**, **Right Arrow**, **Up Arrow** or **Down Arrow**.

Page keys: **Page Up** or **Page Down**

2.2.2 Working With KXSLDbg Output

Most of the output from KXSLDbg is captured and presented either in the inspectors dialog or the KXSLDbg output window. The exceptions to this rule are:

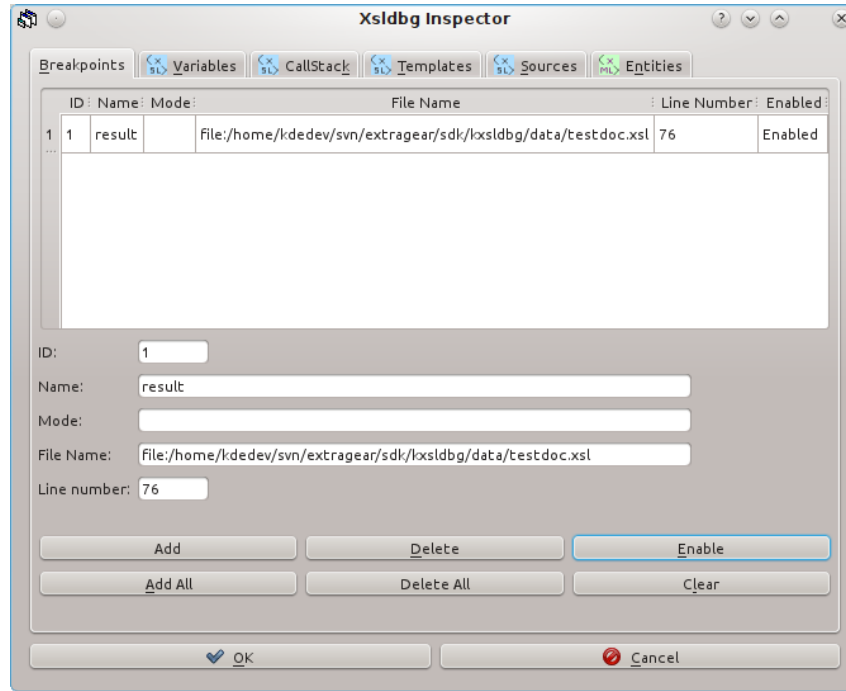
- An error message that comes from KXSLDbg is displayed inside a message dialog.
- The result of evaluating an expression is displayed in a message dialog.
- The output of search is sent to the file indicated in the KXSLDbg output window.

2.3 Setting and Modifying Breakpoints

The primary way to work with breakpoints is via the main window. See Section 2.2

The KXSLDbg Handbook

Once you have started the style sheet, you can use the **Debug** → **Inspect...** menu item. Then click on the Breakpoints tab. See below for an example.



2.3.1 Adding a Breakpoint

You can add a breakpoint by supplying any of:

1. a file and line number
2. a template name
3. a template name and a mode name
4. a mode name

And then pressing the **Add** button.

2.3.2 Argument Details

A file name may be absolute path to a local file. Or partial file (e.g. `xsl.doc.xml`).

A template or mode name may be fully Qualified Name where the non-local part is optional e.g. `xsl:mytemplate` is matched by `mytemplate`

2.3.3 Deleting a Breakpoint

Firstly left mouse click the breakpoint you want to delete in the list of current breakpoints. Then click the **Delete** button.

2.3.4 Deleting All Breakpoints

Click the **Delete All** button.

2.3.5 Enabling or Disabling a Breakpoint

Firstly left mouse button click the breakpoint you want to delete in the list of current breakpoints. Then click the **Enable** button.

2.4 Working With Variables

If the inspector dialog is not showing use the **Debug** → **Inspect...** menu item.

Local and global variables are show in a tab on the inspector dialog. The following example shows a XSLT code segment that declares a global and a local variable

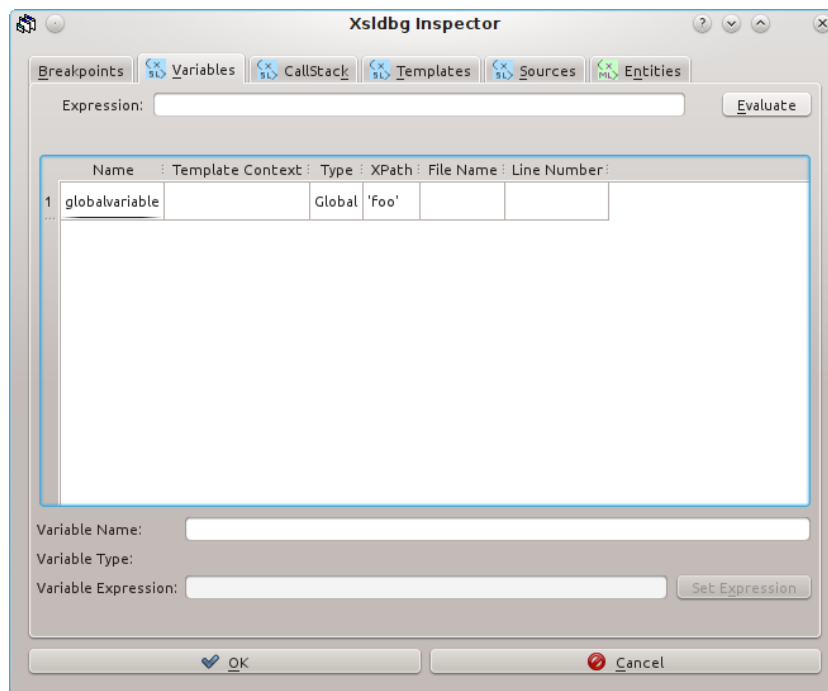
```
<xsl:variable name="globalvariable" select="'foo'"/>

<xsl:template match="/" />
<xsl:param name="localvariable" select="'bar'"/>
</xsl:template match="/" />
```

Clicking with the mouse on a variable in the list will cause summary information to be displayed in the bottom of the dialog. If a variable has a select expression, for example

```
<xsl:variable name="changeable" select="'oldValue' " />
```

then a new XPath can be chosen by entering a new value for **Variable Expression** then clicking the **Set Expression** button.

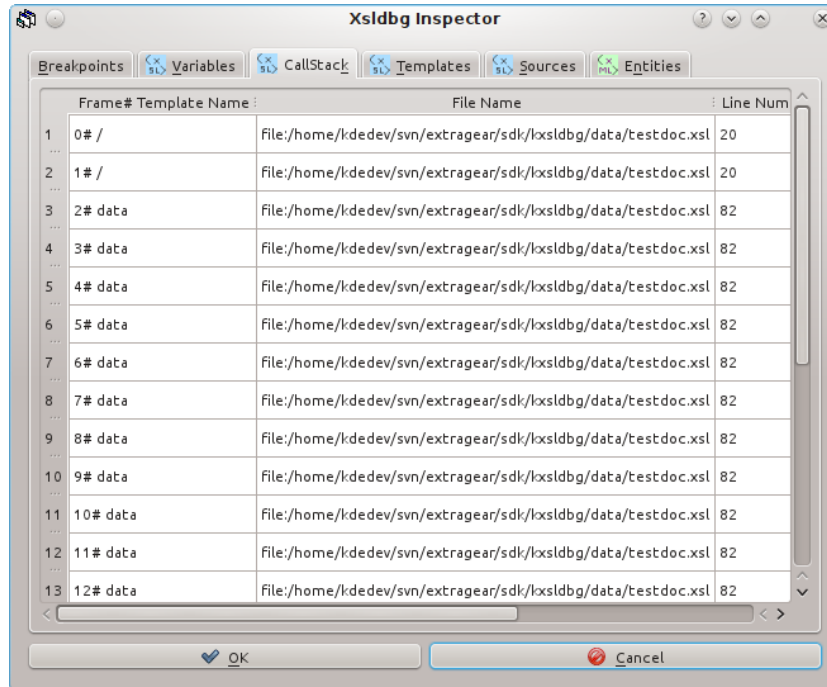


The Variables Tab

Clicking on a variable entry in the list shown will cause the cursor in the main window to move to the file and line number indicated.

2.5 Working With the Callstack

All call stack items found are listed here. The older the callstack entry the lower the frame number it will have. See below for an example.



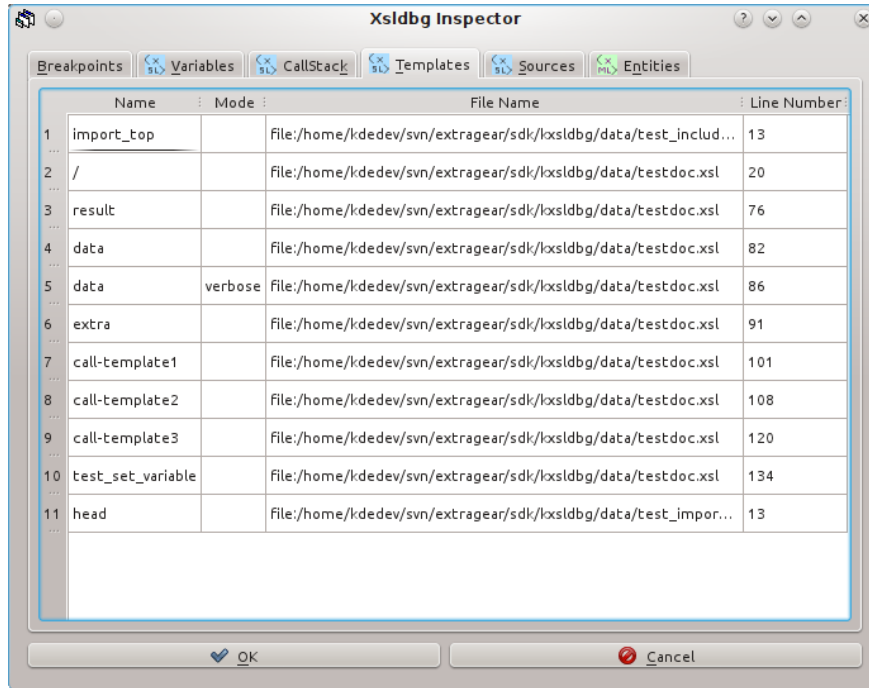
Clicking on a callstack entry in the list shown will cause the cursor in the main window to move to the file and line number indicated.

2.6 Working With Templates

If the inspector dialog is not showing use the **Debug** → **Inspect...** menu item. To work with templates click on the templates tab of dialog shown.

All templates found are listed here. Please note that the export rules of XSLT apply. So only there may be more than one template with the same match and mode details.

The KXSLDbg Handbook

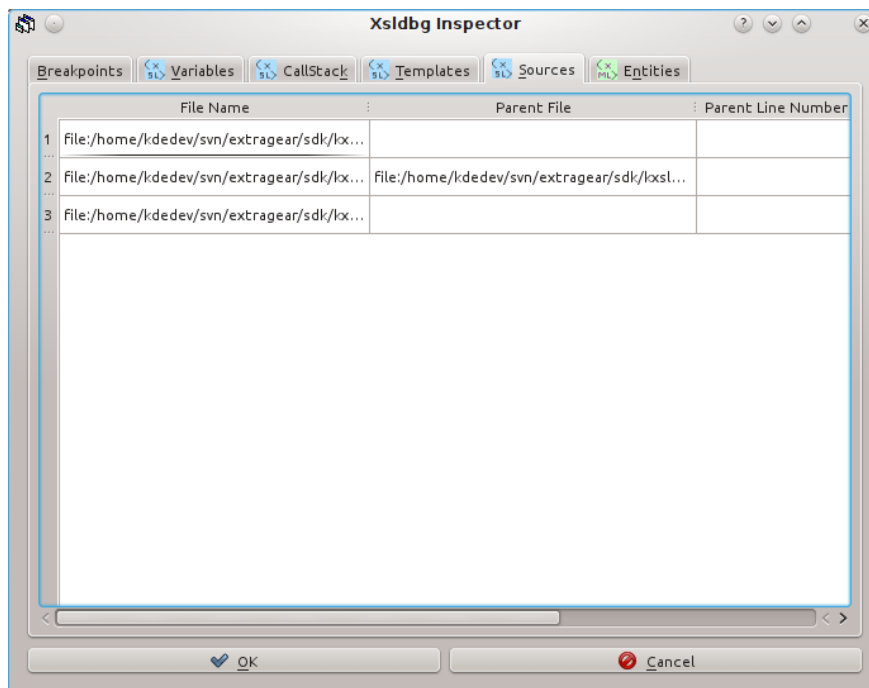


Clicking on a template entry in the list shown will cause the cursor in the main window to move to the file and line number indicated.

2.7 Working With XSLT Source Files (Sources)

If the inspector dialog is not showing use the **Debug** → **Inspect...** menu item. To work with sources click on the sources tab of dialog shown.

All XSLT source files that are included by the XSLT file or one of its siblings are listed here.

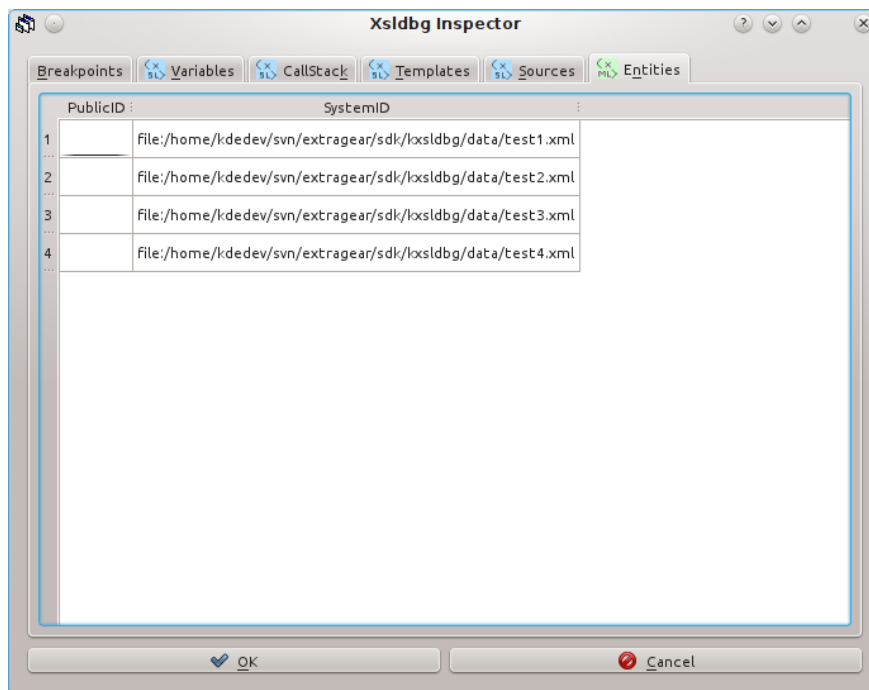


Clicking on a source entry in the list shown will cause the cursor in the main window to move to the start of file indicated.

2.8 Working With XML Data Files (Entities)

If the inspector dialog is not showing use the **Debug** → **Inspect...** menu item. To work with entities click on the **Entities** tab of dialog shown.

All external XML entities included via the DATA file or one of its siblings are listed here. For this example I have run KXSLDbg on `testdoc.xsl` with `testdoc.xml` (found in the `<KDE_PREFIX>/share/apps/kxsldbg` folder so that you can see some entities.



Clicking on a entity entry in the list shown will cause the cursor in the main window to move to the start of the file indicated.

2.9 Miscellaneous

Several tools are available via the **Debug** menu. The main tool is the inspector tool.

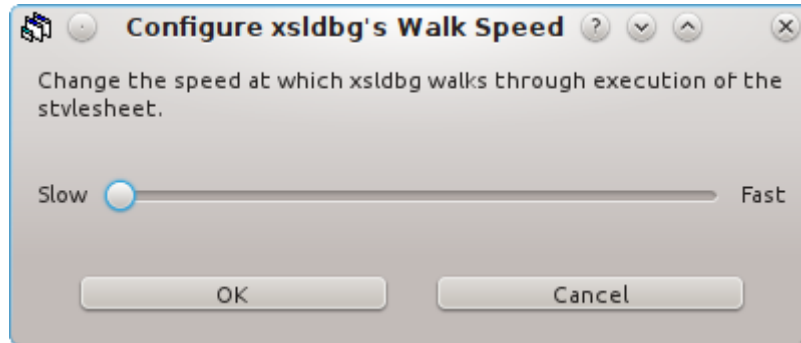
2.9.1 Inspector Tool

The inspector tool contains all the individual dialogs for working with:

- Breakpoints
- Templates
- Variables
- Callstack entries
- XSL source files
- XML Entries

2.9.2 Execute by Walking

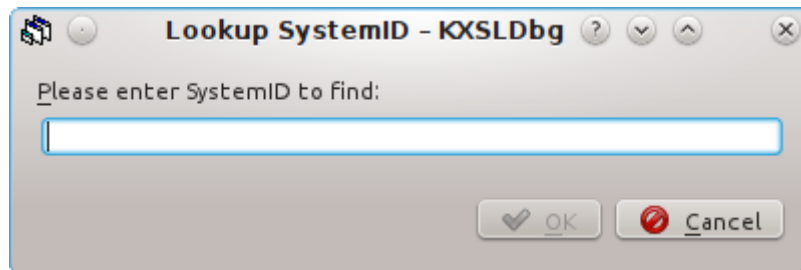
By clicking on **Walk Through Stylesheet...** in the **Debug** menu a dialog is shown to allow the walk speed to be chosen.



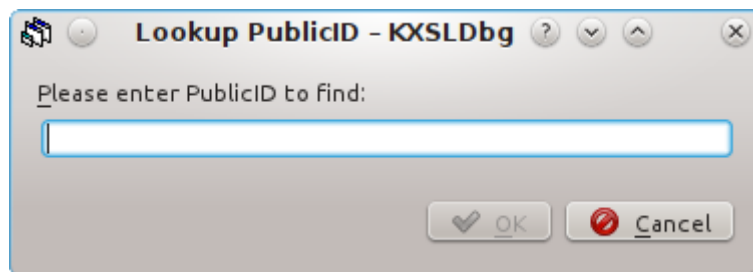
To stop walking either use the **K** key or select the **Stop Walking Through Stylesheet** menu item.

2.9.3 Lookup XML Entities

To lookup a System ID in the current XML catalog use the **Lookup SystemID...** menu then enter the value to find the dialog that displays.



To lookup a PUBLIC ID use the **Lookup PublicID...** menu entry then enter the value to find the dialog that displays.



Chapter 3

Credits and License

KXSLDbg (c) 2004 Keith Isdale

Documentation (c) 2004 Keith Isdale

THANKS TO:

- The writers the libxml and libxslt.
- Robert Jacolin for feedback on earlier version of KXSLDbg.
- Igor Zlatkovic for creating WIN32 binaries of libxml/xslt and xsldb.

This documentation is licensed under the terms of the [GNU Free Documentation License](#).

This program is licensed under the terms of the [GNU General Public License](#).