

How to set up a local TINA code browser

P.A. Bromiley

Last updated
1 / 12 / 2004



Imaging Science and Biomedical Engineering Division,
Medical School, University of Manchester,
Stopford Building, Oxford Road,
Manchester, M13 9PT.

How to set up a local TINA code browser

P.A. Bromiley.
Dept. of Medical Biophysics
Imaging Science and Biomedical Engineering Division
Medical School, University of Manchester
Manchester, M13 9PT, UK
`paul.bromiley@man.ac.uk`

1 Introduction

This document gives instructions on how to set up the Tina on-line code browser (see <http://www.tina-vision.net/software.php>) locally. There are two main scenarios where this could be useful: for laptop owners who wish to work away from their network, and for people on dial-up internet access. I started from a default "install all software" installation of Suse 9.0, and most of these instructions are the result of following the installation instructions for LXR. The version numbers of the major packages involved are:

- Suse 8.0
- Apache 1.3.23
- PHP 4.1.0
- LXR 0.3
- Glimpse 4.15
- TINA 5 (for Tina4, replace `/usr/local/Tina5` with `/usr/local/Tina/src` below)

The specific paths listed below assume that Apache and Perl have been installed in the default locations by Suse, and that TINA has been installed in `/usr/local/Tina`. You may need to be logged on as root to perform most of the steps below.

2 LXR Codebrowser

LXR indexes the source code for a software project, cross-referencing function names, structure name etc., and produces an html page that allows you to browse over these links. It is therefore allows you to navigate through the code, following the execution process. The searching facilities allow you to look up functions, variables or structures quickly if you need to know the arguments or return type

- Check the Apache installation. By default the document root is `/srv/www/htdocs`, and the configuration files are in `/etc/httpd`. The apache daemon is not started by default: go into the directory `/etc/rc.d` and type

```
./apache start
```

to start the daemon.

Start a web browser and point it at `http://127.0.0.1` to show the webserver information page. On most distributions this would be referred to as `http://localhost`, but it may instead be referred to as `http://linux.local`

- OLDER SUSE: I found that with a SUSE 8.0 installation the PHP module required an updated version of the LDAP client. I got the error message

```
Syntax error on line 8 of /etc/httpd/suse\_loadmodule.conf:  
Cannot load /usr/apache/libphp4.so into server: libdap.so.2:  
cannot open shared object file: no such file or directory
```

In order to fix this, use the KDE package manager to remove the package `openldap-client-1.2.13-133.i386.rpm`. This may also require you to remove an `openldap-devel-` package. Then install the package `openldap2-client-2.0.23-53` from directory `n1` on the first CD.

- (Optional) In order to get the freetext searching facilities you need to download Glimpse, from

`http://webglimpse.net`

This requires registration but is free: passwords are mailed automatically.

- The code browser itself is powered by lxr: download this from

`http://lxr.linux.no/`

- (Optional) Install `glimpse`. Put the tarball somewhere convenient (e.g. `/usr/local/glimpse`) and extract the files (using `gunzip` and then `tar -xvf`). Then follow the installation instructions: follow the usual procedure of

```
sh configure
make
make install
```

- Install `lxr`. Put the tarball somewhere convenient (e.g. `/usr/local/lxr`) and extract the files (as above). Then follow the installation instructions. Set the following variables in the Makefile:

```
PERLBIN=/usr/bin/perl
INSTALLPREFIX=/srv/www/htdocs/lxr
```

Then run

```
make install
```

The `lxr` files will be installed under `/srv/www/htdocs/lxr`.

- Set the following variables in the `lxr.conf` file (in `/srv/www/htdocs/lxr/http`). Note that the slash at the end of the path on the `baseurl` line is very important:

```
baseurl: http://127.0.0.1/lxr/http/
sourceroot: /usr/local/Tina5
srcrootname: TINA5
htmlhead: /srv/www/htdocs/lxr/http/template-head
htmltail: /srv/www/htdocs/lxr/http/template-tail
htmldir: /srv/www/htdocs/lxr/http/template-dir
dbdir: /srv/www/htdocs/lxr/source
glimpsebin: /usr/local/bin/glimpse
```

- Create a hidden file `.htaccess` in the same directory and insert the lines

```
<Files ~ (search|source|ident|diff|find)$>
SetHandler cgi-script
</Files>
```

- Insert the following lines exist in `/etc/httpd/access.conf`

```
<Directory /srv/www/htdocs/lxr>
Options All
AllowOverride All
</Directory>
```

- Create the identifier database using:

```
cd /srv/www/htdocs/lxr/source
/srv/www/htdocs/lxr/bin/genxref /usr/local/Tina5
```

- Create the glimpse database using:

```
cd /srv/www/htdocs/lxr/source
/usr/local/bin/glimpseindex -H . /usr/local/Tina5
```

- Restart the webserver: in the directory /etc/rc.d type

```
./apache stop
./apache start
```

- Start a web browser and go to the address

```
http://127.0.0.1/lxr/http/source
```

If all of the above has worked, you should now be looking at the Tina code browser. Note that you can change the installation location in the lxr Makefile and install additional versions of lxr, so you could have separate codebrowsers for the tina-libs and tina-tools directories, or a separate codebrowser for Tina 4 on the same machine.

- Troubleshooting:

- If after the final step you are looking at a page of text, then double check the .htaccess file. This ensures that the perl scripts are run, rather than viewed as text.
- If the front page looks correct, but none of the links work, check the permissions on the files in

```
/srv/www/htdocs/lxr/source
```

They should be globally readable. This also applies to the hidden files: remember that

```
chmod 755 *
```

does not change permissions on hidden files, so use

```
chmod 755 .*
```

aswell.

- If the front page looks correct, but none of the links work, and you have followed the last step correctly, then check that you included the trailing slash on the baseurl line in lxr.conf.
- Failing the above, read the lxr documentation. Most errors will be the result of problems with file permissions. If the machine has never been set up as a web server, it might also be a good idea to read the Apache documentation re. security.

3 Doxygen Codebrowser

Doxygen is a system for documenting code through comments contained within the source. The addition of these comments to Tina is a work-in-progress, but some areas of the code are extensively documented. At this point probably the most useful function of the doxygen codebrowser for Tina 5 is the dependency graphing, which allows you to see at an instant all of the other source code files that a given file is dependent on.

Setting up the doxygen codebrowser is trivial:

- Both doxygen (and the dot tool for producing the dependency graphs) should be installed by default on a full installation of Suse 9.0 or later, as

```
/usr/bin/doxygen
/usr/bin/dot
```

If one is missing, use Yast to install it.

- Create a directory called `/srv/www/htdocs/doxygen` and `cd` into it
- Type

```
doxygen -g config_file
```

This will create the config-file, which contains the configuration of the doxygen page. Edit this to set up up the doxygen page for Tina. The file contains extensive notes on the function of each tag, so the values needed are fairly obvious. See

```
/usr/share/doc/packages/doxygen/doxygen_manual.pdf
```

for further instructions.

- Type

```
doxygen config-file
```

Note that this can take some time to complete...

- Point your browser at

```
http://localhost/doxygen/html/index.html
```

You should now be looking at the doxygen page.