

Linux Tips and Tricks

Chris Karakas

Linux Tips and Tricks

by Chris Karakas

This is a collection of various tips and tricks around Linux. Since they don't fit anywhere else, they are presented here in a more or less loose form. We start on how to get an attractive [vi](#) by using syntax highlighting (even for ABAP/4[®]), [multiple search](#) and the [smartcase](#), [incsearch](#), [scrolloff](#), [wildmode](#) options in the [vimrc](#) file. We continue on various configuration subjects regarding Netscape[®], like positioning and sizing of windows, capturing a whole HTML page in an image, or roaming profiles. Further on, I present a CSS file for HTML documents that were created automatically from DocBook SGML. Also a chapter on transparent proxying with [Squid](#) and one on [LyX](#), covering mass insertion of cross-references and automatic index generation.

Revision History

Revision 0.09 18.02..2004 Revised by: CK
Automatic Index generation with LyX
Revision 0.08 22.01..2004 Revised by: CK
Chapter on LyX, cross-references, Licence section, Appendix
Revision 0.07 25.12..2003 Revised by: CK
Discussion of the CSS for DocBook code, Credits
Revision 0.06 01.08..2003 Revised by: CK
HTML to PNG. ABAP/4 syntax highlighting
Revision 0.05 01.07..2003 Revised by: CK
More on roaming profiles.
Revision 0.04 04.06..2003 Revised by: CK
Added chapter on transparent proxying. Added Index. Alt text and captions for images
Revision 0.03 29.05.2003 Revised by: CK
Added chapters on Netscape and CSS for DocBook.
Revision 0.02 18.12.2002 Revised by: CK
first version

Table of Contents

| | |
|--|-----------|
| 1. Introduction | 1 |
| 1.1. Disclaimer | 1 |
| 1.2. Formats | 1 |
| 1.3. Licence | 2 |
| 1.4. Availability of sources and support..... | 2 |
| 1.5. Credits | 2 |
| 2. Sexy vi | 3 |
| 3. Netscape | 5 |
| 3.1. Positioning and sizing of windows for Netscape Communicator | 5 |
| 3.2. Netscape roaming profiles | 6 |
| 3.3. How to save a HTML page as an image: from HTML to PNG | 8 |
| 3.3.1. Enter html2ps..... | 8 |
| 3.3.2. ...and ps2png! | 9 |
| 3.4. Restore HTML sessions | 9 |
| 4. CSS for DocBook SGML | 11 |
| 5. Transparent proxying with Squid | 22 |
| 6. LyX | 24 |
| 6.1. Mass insertion of cross-references in LyX | 24 |
| 6.2. Automatic Index generation | 26 |
| A. Appendix | 31 |
| A.1. The GNU Free Documentation Licence | 31 |
| A.1.1. PREAMBLE | 31 |
| A.1.2. APPLICABILITY AND DEFINITIONS | 31 |
| A.1.3. VERBATIM COPYING | 32 |
| A.1.4. COPYING IN QUANTITY | 33 |
| A.1.5. MODIFICATIONS | 33 |
| A.1.6. COMBINING DOCUMENTS | 35 |
| A.1.7. COLLECTIONS OF DOCUMENTS | 35 |
| A.1.8. AGGREGATION WITH INDEPENDENT WORKS | 35 |
| A.1.9. TRANSLATION..... | 36 |
| A.1.10. TERMINATION..... | 36 |
| A.1.11. FUTURE REVISIONS OF THIS LICENSE..... | 36 |
| A.1.12. ADDENDUM: How to use this License for your documents..... | 37 |
| Index | 38 |

List of Figures

| | |
|-------------------------------|----|
| 2-1. vi vs. emacs | 3 |
| 3-1. Netscape..... | 5 |
| 3-2. Netscape..... | 6 |
| 4-1. Valid CSS..... | 11 |
| 4-2. CSS page area model..... | 17 |
| 5-1. Squid..... | 22 |
| 6-1. LyX..... | 24 |

Chapter 1. Introduction

1.1. Disclaimer

No liability for the contents of this documents can be accepted. Use the concepts, examples and other content at your own risk. As this is a new edition of this document, there may be errors and inaccuracies, that may of course be damaging to your system. Proceed with caution, and although this is highly unlikely, the author does not take any responsibility for that. If it breaks, you keep both pieces.

All copyrights are held by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark.

Naming of particular products or brands should not be seen as endorsements.

1.2. Formats

This document is available in the following formats:

- [HTML \(HyperText Markup Language\)](#), for viewing with [any browser](#)
- [HTML \(HyperText Markup Language\)](#), one big HTML file
- [TXT \(ASCII Text\)](#), for any text editor
- [RTF \(Rich Text Format\)](#), for Word
- [PDF \(Portable Document Format\)](#), for printing (and viewing with Acrobat Reader)
- [PS.GZ \(Compressed Postscript\)](#), for printing (and viewing with Ghostview)
- [SGML \(Standard Generalized Markup Language\)](#) (with the [Appendix](#)), for creating all the above formats yourself
- [LYX \(LaTeX frontend LyX\)](#) (with the [Appendix](#)), for typesetting it with LaTeX



IMPORTANT: Downloads for offline reading!

If you want to download the HTML or RTF formats for offline reading, you will need to download the images as well - PNG for HTML and BMP for RTF, including the callouts! To save you the hassle, I have compiled the following zipped tar archives for offline reading:

- [TAR.GZ \(Compressed TAR Archive\)](#), many HTML files with images
- [TAR.GZ \(Compressed TAR Archive\)](#), one big HTML file with images

- [TAR.GZ \(Compressed TAR Archive\)](#), SGML file with images
- [TAR.GZ \(Compressed TAR Archive\)](#), RTF file with images

A tarball containing all the above is also available:

- [TAR.GZ \(Compressed TAR Archive\)](#), All files

1.3. Licence

Copyright © 2003, 2004 [Chris Karakas](#). Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts. A copy of the license can be found in [Section A.1](#), as well as at the [GNU Free Documentation License GNU0](#) .

1.4. Availability of sources and support

See [Section 1.2](#) for the modifiable sources of this document.



Pose your questions in my Linux Forum!

If you have questions, patches, suggestions, problems or (better) solutions, come to [my Linux Forum](#) and post them there. I look forward to your feedback!

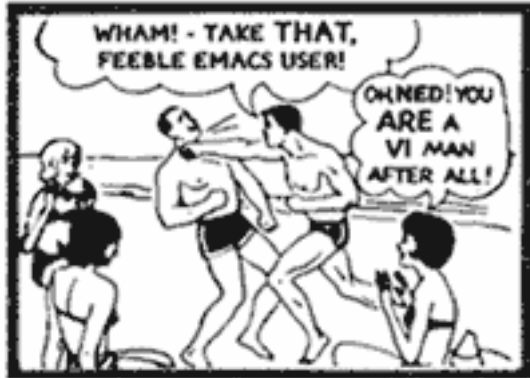
1.5. Credits

[Figure 4-2](#) is taken from W3C's working draft [CSS3 Paged Media Module](#), version of Dec. 18th 2003 and is Copyright © 2003 [W3C](#) ([MIT](#), [ERCIM](#), [Keio](#)), All Rights Reserved. Used with permission according to [W3C document licence](#).

The [CSS file for DocBook](#) that is used in this document , [ck-style.css](#) uses [QBullets](#) in links. Thanks to Matterform Media for providing QBullets for free. If you plan to use them on your website, please observe the [QBullets usage terms](#).

Chapter 2. Sexy vi

Figure 2-1. vi vs. emacs



vi vs. emacs

You can lend your `vi` an extra sex appeal, just by tweaking some values in its configuration file, `vimrc` (usually located in `/etc`):

- Syntax highlighting:

```
" enable syntax highlighting
so ${VIMRUNTIME}/syntax/syntax.vim
```

- Having enabled syntax highlighting, you will often find yourself trying hard to find out what is written on your screen, unless you are working on the console, or the background colour of your X terminal is black. Mine is grey (or red, for root), which makes the colourful text hard to read. In this case, you need to start `vi` in an `xterm` with black as the background colour. Save the following one-liner in `/usr/local/bin/bt` (bt for Black Terminal...):

```
xterm -bg black -fg white -geometry 120x60 -e vi $1 &
```

make it executable (`chmod ugo+x /usr/local/bin/bt`) and from now on, just do

```
bt filename
```

- Set `multiple search`: `MultipleSearch` allows you to have the results of multiple searches displayed on the screen at the same time. Each search highlights its results in a different color, and all searches are displayed at once. After the maximum number of colors is used, the script starts over with the first color. The command syntax is

```
:Search <pattern1>
```

which will highlight all occurrences of `<pattern1>` in the current buffer. A subsequent

```
:Search <pattern2>
```

will highlight all occurrences of <pattern2> in the current buffer, retaining the highlighting of <pattern1> as well. <pattern1> and <pattern2> are any search pattern like you would use in a normal /<pattern> search.

You enable Multiple Search with the following lines in the *vimrc* file:

```
" enable MultipleSearch functionality (extra module)
so ${VIMRUNTIME}/MultipleSearch.vim
```

Further, you will have to copy the [MultipleSearch.vim](#) file from the scripts collection at www.vim.org into your local plugins directory (/usr/share/vim/current on my system) and restart vi.

- Set [smartcase](#), [incsearch](#), [scrolloff](#), [wildmode](#)



Be careful:

No spaces between the options! See "longest,list" below - if you write "longest, list" you will get \$ signs at the end of every line and the tabs will be shown as "^I"!

```
set smartcase
set incsearch
set scrolloff=2
set wildmode=longest,list
```

- For those of you who are working in the [SAP®](#) field: there is an [ABAP/4®](#) syntax highlighting script for vi available at [vim's Sourceforge page](#). Copy the [abap.vim](#) file into your local syntax plugins directory (/usr/share/vim/current/syntax on my system) and restart vi. Happy ABAP/4® hacking!

For more vi tips see <http://www.vim.org/tips/index.php>.

Chapter 3. Netscape

3.1. Positioning and sizing of windows for Netscape Communicator

Figure 3-1. Netscape.



Netscape.

To customise the size and position of various windows of Netscape Communicator® 4.x, you can use the following geometry names. Add all or a part of the following values to the file `~/.Xdefaults` in your home directory.

```
! Navigator main window (browser)
Netscape.Navigator.geometry:
! Mail- and news-reader window
Netscape.MailThread.geometry:
! Netscape message centre (list of mail and news folders)
Netscape.MailFolder.geometry:
! Netscape HTML editor
Netscape.Editor.geometry:
! Composition window (writing of new email's and/or news)
Netscape.Composition.geometry:
! Address book
Netscape.AddressBook.geometry:
! Window to edit the bookmarks
Netscape.bookmarks.geometry:
```

The above rules are case sensitive. If you want the above rules to be global for all users, you can add them to the file `/path/to/netscape/Netscape.ad`. Then change to the directory `/usr/X11/lib/X11/app-defaults` and add a link to this file here:

```
ln -s /path/to/netscape/Netscape.ad Netscape
```

The above link makes sure, that it points always to the correct version of Netscape.ad, especially if you ever install a new version of Netscape Communicator.

(I have adapted this tip from <http://members.aon.at/theofilu/ninst.html>)

3.2. Netscape roaming profiles

Figure 3-2. Netscape.



Netscape.

In Apache's document root (`/usr/local/httpd/htdocs`), create a directory that will hold the roaming data, set the owner and group to the user and group ID of the Apache server and the permissions to 700 (read, write and change permissions only for the owner):

```
mkdir roaming
chown wwwrun.nogroup roaming
chmod 700 roaming
```

In Apache's configuration file, `/etc/httpd/httpd.conf`, insert the following:

```
#
# roaming profiles
#
PerlModule Apache::Roaming
<Location /roaming>
PerlHandler Apache::Roaming->handler
PerlTypeHandler Apache::Roaming->handler_type
AuthType Basic
AuthName "Roaming User"
AuthUserFile /etc/httpd/passwd
require valid-user
```

```
PerlSetVar BaseDir /usr/local/httpd/htdocs/roaming
</Location>
```

This assumes that you want to store the roaming profiles in the directory `/roaming` under the web server's document root (`/usr/local/httpd/htdocs`) and that you want to store the passwords in `/etc/httpd/passwd`. You may choose whatever name you like for the roaming directory and whatever location and name you wish for the password file. The roaming directory has to be located under the document root though. Change the locations above appropriately to fit your situation.

You create the password file when you create the first password using the `htpasswd` command:

```
htpasswd -cm /etc/httpd/passwd <username>
```

The `-c` flag creates a new file, and is necessary the first time you run `htpasswd`. The `-m` flag is optional, and forces MD5 encryption of the password. Omit the `-c` flag for subsequent users:

```
htpasswd -m /etc/httpd/passwd <username>
```

Next, install the [Apache::Roaming Perl module](#). Download it from the [CPAN Apache module directory](#). Extract the files from the tar.gz archive:

```
tar -xzf Apache-Roaming-0.1003.tar.gz
```

change to the newly created directory (in our case `Apache-Roaming-0.1003`) and type

```
perl Makefile.PL
make
make install
```

Answer the questions about the path of the Apache server (in my case `/usr/sbin/httpd`), the path of the `httpd.conf` file (in my case `/etc/httpd/httpd.conf`), the user and group ID of Apache (`wwwrun` and `nogroup` respectively, in my case - you will find those in the web server configuration file `/etc/httpd/httpd.conf`, as the values to the variables `User` and `Group` resp.). You may also configure the `Apache::Roaming::LiPrefs` module, assigning the default values you wish to preferences.

At last, you need to enable roaming in Netscape:

- Start your communicator and open Preferences/Roaming User. Click the "Enable Roaming Access for this profile" checkbox.
- Open Preferences/Roaming User/Server Information. Click the "HTTP Server" checkbox and enter the Base URL `http://yourwebserver/roaming/user` (in my case `http://bacchus/roaming/chris`).
- Restart Netscape.



Tip:

If you are having problems, check that your Apache loads the Perl module. In SuSE Linux systems, this is controlled by the extra files `/etc/httpd/suse_loadmodule.conf` and `/etc/httpd/suse_addmodule.conf`.

If you recompiled Apache after an initial installation, the subsequent SuSEconfig call will not overwrite these two files. Instead it will write its own version in files with the ending `.SuSEconfig` (e.g. `/etc/httpd/suse_addmodule.conf.SuSEconfig`). You are strongly advised to check the contents of those two files and copy them to the original ones:

```
cp /etc/httpd/suse_loadmodule.conf.SuSEconfig /etc/httpd/suse_loadmodule.conf
cp /etc/httpd/suse_addmodule.conf.SuSEconfig /etc/httpd/suse_addmodule.conf
```

The reason is that you will be probably missing a line like

```
LoadModule perl_module /usr/lib/apache/libperl.so
```

in your `suse_loadmodule.conf` file (and a similar one in `suse_addmodule.conf`).

3.3. How to save a HTML page as an image: from HTML to PNG

Netscape 4.x is an old browser. It has many bugs (see, for example, [Netscape CSS bugs](#), [Netscape 4.x: Web Designer's worst nightmare](#), or [why Netscape 4.x sucks](#)) but I still use it, mainly because I am reluctant to "change a running system" and I always see a difficulty as a challenge to resolve it. This was the case with the well-known problem of Netscape 4.x not printing the background colours of tables. Now, what if you have a nice table, whose groups of lines are coloured to convey some information about those groups?

Forget printing in colour - it will not work. A solution is to capture the screen as an image and print that image instead. This works as long as the table is small enough to fit on the screen - I had one that didn't.

Since I couldn't find an image capture tool for Linux that will let you scroll down and capture the whole page, I had to find some "HTML to PNG" (or HTML to JPEG, GIF or whatever) tool.

In fact, there is one: [html2jpg](#). It converts HTML pages to images (gif, jpg, png, ps, etc..). It is a Perl script, ran from the command line. It requires

- [Perl](#),
- convert, from [ImageMagick](#),
- `xwininfo`, `xwd` (standard Linux utilities),
- [Opera](#), otherwise you have to modify the `find_window()` function in the script.

The script is clearly written, but the last requirement poses a real problem - which I preferred not to solve.

3.3.1. Enter `html2ps`...

In fact, there is a much simpler way to accomplish the same thing: a pipe of `html2ps` and `ps2png`! [html2ps](#) is contained in every distribution. It offers a variety of features:

- Many possibilities to control the appearance, mostly done using configuration files.
- Support for processing multiple documents, also automatically by recursively following links.
- A table of contents can be generated, either from the links in a document, or automatically from document headings.
- Configurable page headers/footers, that for example can contain document title, URL, page number, current heading, and date.
- When converting the PostScript® document to PDF - using some other program such as version 5.0 or later of Aladdin Ghostscript®, or Adobe AcrobatDistiller® - the original hyperlinks in the HTML documents will be retained in the PDF document.
- Automatic hyphenation and text justification can be selected.

3.3.2. ...and ps2png!

ps2png can be implemented in many ways, a simple one is the following short script¹ (taken from <http://stud3.tuwien.ac.at/e9902644/work/invocation.html>):

```
#!/bin/sh
# /usr/local/bin/ps2png
for i in "$@"
do
    i_new='dirname $i/'`basename $i .ps`.png
    echo convert $i to $i_new
    gs -dNOPAUSE -sDEVICE=png256 -sOutputFile=$i_new -q -dBATCH $i
#    convert -rotate 90 $i_new $i_new
#    convert -resize 50% $i_new $i_new
done
#convert -transparent white test.png test.png
```

Now, to output both pages in one (-2 option), switch to landscape mode² (-L) and dump the contents in a PNG image with 256 colours (in order to be "web safe"), we only need to type:

```
html2ps -2 -L http://somesite.com/somepage.html > somepage.ps
ps2png somepage.ps
```

You can write a html2png script containing just these two lines - and you are done! 😊

3.4. Restore HTML sessions

As said above, Netscape 4.x is old. And every so often it encounters some code it cannot handle, even worse: it crashes. This will of course happen exactly when I will have 20+ HTML sessions open, tracking all hyperlink branches of some original interesting page I have long forgotten. 😞

1. Commented code is not needed, but is "nice to have" just in case.
2. Remember that the original HTML page was too large for a screen dump without scrolling.

It is a pain to restore all lost HTML sessions after a browser crash. Wouldn't it be nice if we could keep track of all visited URLs in a text file, then, after restarting the browser, just type a command that will open them all, each one in its own browser window? In fact, this is what the following script does:

```
#!/bin/bash
# Remote invocation of my favorite URLs in netscape.
#
# Motivation: netscape sucks too often. I _had_ to find
# a way to load my lovely URLs without much typing.
# Now, just typing "netscape-recover" will do the trick.
#
# Enter all the URLs you wish to recover in URLs_Netscape,
# one URL per line.
URLS_Netscape="URLS_Netscape"
for url in `cat $URLS_Netscape`; do
    Netscape -remote "openURL($url,new-window)"
done
```

The remote invocation of Netscape is described in detail in [Remote Control of UNIX Netscape](#), by Jamie Zawinski.

Chapter 4. CSS for DocBook SGML

Figure 4-1. Valid CSS.



Valid CSS.

You have written your document using [DocBook SGML](#). You have used all the available tools to transform ("render") your document to various formats, including, of course, HTML. But now you realize that your document looks somewhat "boring". It's HTML is well-written - no question. But you just can't avoid the feeling that there is still a lot that can be done to produce a more visually appealing version. In fact, you need a Cascading Style Sheet, a CSS.

"So what's the problem?" - you might say. "I'll grab the first CSS I can find and I will use it". Well, the problem is, that you will not be able to. The documents created by Jade & Co. use some classes that you will not find in every CSS file - so you will have to find the right formatting yourself.

Searching the Internet did not bring many results, see for example the [List of DocBook CSS](#) at the DocBook Wiki.

Those of you who would like to experiment with a CSS that works with DocBook right away may copy the following one which I use for my own documents, including the one you are reading:

```
BODY
{
font-family: verdana;
}

DIV.ABSTRACT
{
border: solid 2px;
padding-left: 10pt;
padding-right: 10pt;
}

PRE.SCREEN
{
font-family: monospace;
white-space: pre;
width: 100%;
    background-color: #ffffcc;
    border: solid;
color: #000000;
    border-color: #009999;
    border-left: solid #009999 2px;
    border-right: solid #009999 2px;
```

```

border-top: solid #009999 2px;
border-bottom: solid #009999 2px;

padding-left: 15pt;
}

PRE.PROGRAMLISTING
{
font-family:monospace;
white-space: pre;
width: 100%;
background-color: #ffffcc;
border:solid;
color: #000000;
border-color: #009999;
border-left: solid #009999 2px;
border-right: solid #009999 2px;
border-top: solid #009999 2px;
border-bottom: solid #009999 2px;

padding-left: 15pt;
}

H1
{
color: #ffffff;
border: solid 3px #a0a0d0;
background-color: #606090;
font-variant: small-caps;
width: 100%;
}

H1.TITLE
{
color: #ffffff;
border: solid 3px #a0a0d0;
background-color: #606090;
font-variant: small-caps;
width: 100%;
}

.TITLE a {
color: #ffffff;
text-decoration: none;
}

.TITLE a:active {
color: #ffffff;
text-decoration: none;
}

.TITLE a:visited {
color: #ffffff;
}

```

```

        text-decoration: none;
    }

H2
{
    COLOR: #ffffff ;
    font-style: italic;
    BACKGROUND-color: #a0a0d0;
    BORDER: solid 3px #606090;
    PADDING: 1px
}

TABLE.IMPORTANT
{
font-style:italic;
border: solid 2px #ff0000;
width: 70%;
margin-left: 15%;
}
TABLE.CAUTION
{
font-style:italic;
border: ridge 2px #ffff00;
width: 70%;
margin-left: 15%;
}

TABLE.NOTE
{
font-style:italic;
border: solid 1px #000000;
width: 70%;
margin-left: 15%;
}
TABLE.TIP
{
font-style:italic;
border: solid 1px #000000;
width: 70%;
margin-left: 15%;
}

TABLE.WARNING
{
font-style:italic;
    font-weight: bold;
border: ridge 4px #ff0000;

width: 70%;
margin-left: 15%;
}

DIV.VARIABLELIST {

```

```

        font-family: sans-serif;
        font-style: normal;
        font-weight: normal;
        padding-left: 20px;
    }

    .VARLISTENTRY {
        font-weight: bold;
        margin-top: 10px;
        COLOR: #ffffff ;
        BACKGROUND-color: #a0a0d0;
        BORDER: solid 1px #606090;
        PADDING: 1px
    }

/*
 * See http://diveintoaccessibility.org/day_26_using_relative_font_sizes.html
 * for an explanation of the following few commands.
 * They are really too complicated to explain here in all depth. ;-)
 */

P {
font-size: 12px;
}

/**/A{}
BODY P {
font-size: x-small;
voice-family: "\"}\"";
voice-family: inherit;
font-size: small;
}

HTML>BODY P {
font-size: small;
}
/* */

/* Add an external-link icon to absolute links */
a[href^="http:"] {
    background: url(images/remote.gif) right center no-repeat;
    padding-right: 12px;
}

a[href^="http:"]:hover {
    background: url(images/remote_a.gif) right center no-repeat;
}

/* Add a note icon to footnote links */
a[href^="#FTN"] {
    background: url(images/qbullet-note.gif) right center no-repeat;
    padding-right: 12px;
}

```

```

a[href^="#FTN"]:hover {
    background: url(images/qbullet-note_a.gif) right center no-repeat;
}

/* ...and a back icon to the backlinks in the footnotes themselves */
a[name^="FTN"] {
    background: url(images/scrollup.gif) right center no-repeat;
    padding-right: 12px;
}

a[name^="FTN"]:hover {
    background: url(images/scrollup_a.gif) right center no-repeat;
}

/* Add a download icon to .gz links */
a[href$=".gz"],a[href$=".tar"],a[href$=".zip"] {
    background: url(images/disk.gif) right center no-repeat;
    padding-right: 12px;
}

/* Add an Acrobat Reader icon to PDF links */
a[href$=".pdf"] {
    background: url(images/acrobat.gif) right center no-repeat;
    padding-right: 12px;
}

a[href$=".pdf"]:hover {
    background: url(images/acrobat_a.gif) right center no-repeat;
}

/* Add a Word icon to RTF links */
a[href$=".rtf"] {
    background: url(images/word.gif) right center no-repeat;
    padding-right: 12px;
}

/* ...but not to absolute links in this domain... */
a[href^="http://www.karakas-online.de"] {
    background: transparent;
    padding-right: 0px;
}

a[href^="http://www.karakas-online.de"]:hover {
    background: transparent;
}

/* ...or to the translation links on each page */
DIV.translatelink > a[href^="http:"] {
    background: transparent;
    padding-right: 0px;
}

```

```

}

DIV.translatelink > a[href^="http:"]:hover {
    background: transparent;
}

/* ...or to any images */
DIV.imagelink a[href^="http:"] {
    background: transparent;
    padding-right: 0px;
}

DIV.imagelink a[href^="http:"]:hover {
    background: transparent;
}

P.C2 {
    COLOR: #ffffff ;
    BACKGROUND-color: #a0a0d0;
    BORDER: solid 1px #606090;
    PADDING: 1px
}

DIV.NAVFOOTER {
color: #000000;
    background-color: #EFEFF8;
    padding: 5px;
    margin-top: 10px;
width: 100%;
border: thin solid #a0a0d0;
}

DIV.NUKEFOOTER {
color: #000000;
    background-color: #B0E0E6;
    padding: 5px;
    margin-top: 10px;
width: 100%;
border: thin solid #a0a0d0;
}

DIV.NAVHEADER {
color: #000000;
    background-color: #EFEFF8;
    padding: 5px;
    margin-bottom: 10px;
width: 100%;
border: thin solid #a0a0d0;
}

DIV.SECT1,DIV.SECT2,DIV.SECT3 {

```

```

        margin-left: 20px;
    }

    DIV.EXAMPLE,DIV.TOC {
        border: thin dotted #70AAE5;
        padding-left: 10px;
        padding-right: 10px;
    color: #000000;
        background-color: #EFF8F8;
    }

    DIV.TOC {
        margin-left: 20px;
        margin-right: 20px;
    width: 95%;
    }

    UL {
    list-style: url("images/tux-bullet.png") disc;
        }

```

It incorporates important elements from the [Newbiedoc CSS file for DocBook](#) and Mark Pilgrim's influential [dive into Accessibility](#). It is an integral part of my method of [document processing with LyX and SGML](#), which I use for all the documents on [my Homepage](#).

You can control the navigation header style with

```

    DIV.NAVHEADER {
        color: #000000;
        background-color: #EFEFF8;
        padding: 5px;
        margin-bottom: 10px;
        width: 100%;
        border: thin solid #a0a0d0;
    }

```

and the navigation footer style with

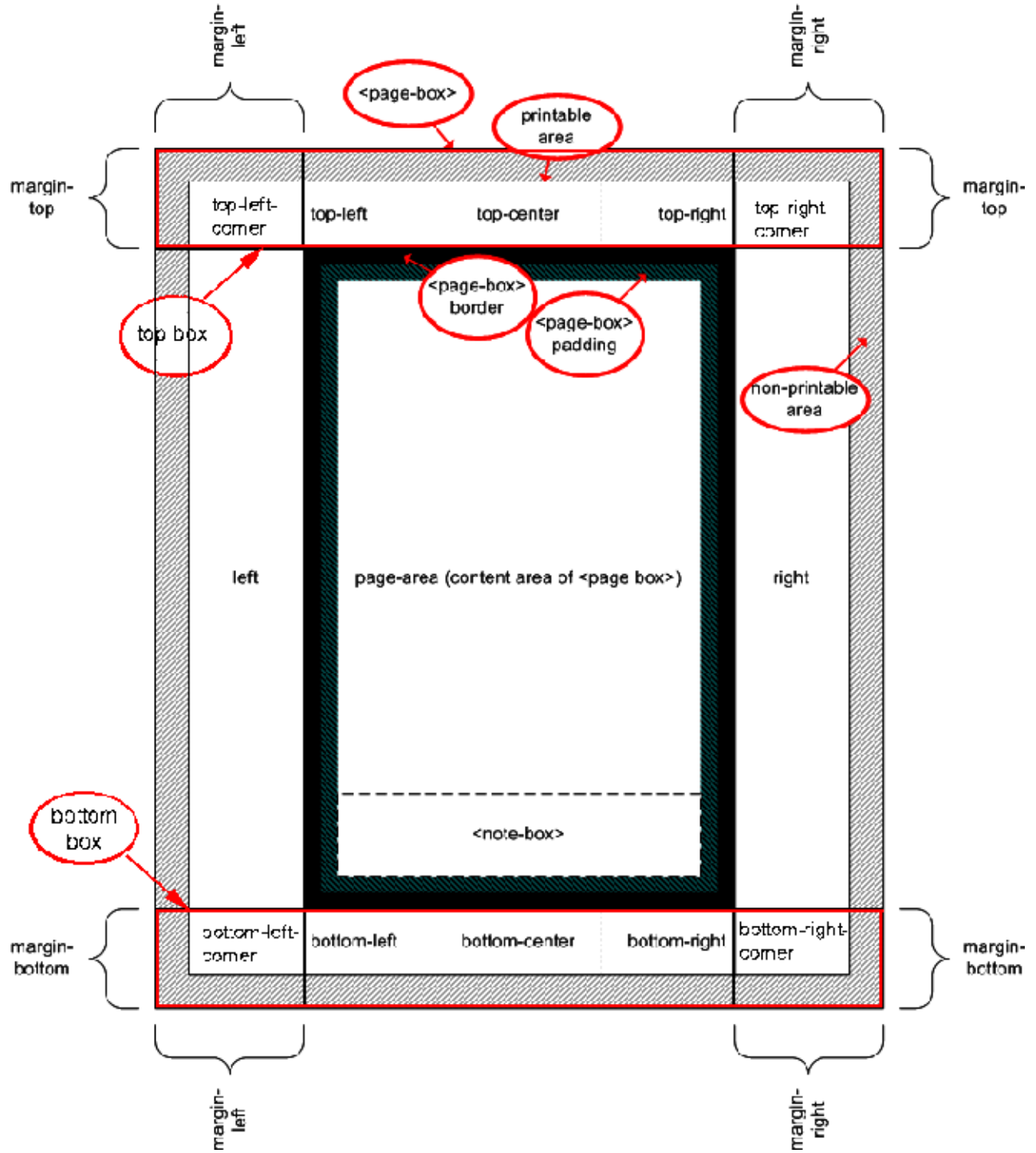
```

    DIV.NAVFOOTER {
        color: #000000;
        background-color: #EFEFF8;
        padding: 5px;
        margin-top: 10px;
        width: 100%;
        border: thin solid #a0a0d0;
    }

```

The concepts of margin, border and padding follow a page model that is described in W3C's working draft [CSS3 Paged Media Module](#), version of Dec. 18th 2003. [Figure 4-2](#), taken from this document (Copyright © 2003 [W3C](#) ([MIT](#), [ERCIM](#), [Keio](#)), All Rights Reserved) illustrates the various geometric notions of this page model. Note that the [XSL area model](#) is deliberately very similar to the CSS one.

Figure 4-2. CSS page area model.



CSS page area model.

If you output admonitions as tables, rather than graphics (see [Admonitions](#)), then you can control their style with a code like the following (shown here for the "Important" admonition):

```
TABLE.IMPORTANT
{
    font-style:italic;
    border: solid 2px #ff0000;
    width: 70%;
    margin-left: 15%;
}
```

Screen output (code) is controlled by

```
PRE.SCREEN
{
    font-family:monospace;
    white-space: pre;
    width: 100%;
    background-color: #ffffcc;
    border:solid;
    color: #000000;
    border-color: #009999;
    border-left: solid #009999 2px;
    border-right: solid #009999 2px;
    border-top: solid #009999 2px;
    border-bottom: solid #009999 2px;
    padding-left: 15pt;
}
```

while examples and the Table of Contents by

```
DIV.EXAMPLE,DIV.TOC {
    border: thin dotted #70AAE5;
    padding-left: 10px;
    padding-right: 10px;
    color: #000000;
    background-color: #EFF8F8;
}
DIV.TOC {
    margin-left: 20px;
    margin-right: 20px;
    width: 95%;
}
```

An "external link" icon to absolute links (i.e. links starting with "http:") is added through

```
/* Add an external-link icon to absolute links */
a[href^="http:"] {
    background: url(images/remote.gif) right center no-repeat;
    padding-right: 12px;
}
a[href^="http:"]:hover {
    background: url(images/remote_a.gif) right center no-repeat;
}
```

However, this alone would put the icon on every link with an absolute URL, including links pointing to the local domain. This is corrected by

```
/* ...but not to absolute links in this domain... */
a[href^="http://www.karakas-online.de"] {
    background: transparent;
    padding-right: 0px;
}
a[href^="http://www.karakas-online.de"]:hover {
    background: transparent;
}
}
```

“External link” icons tell you what a link will do before you click on it. There are icons specifically designed for this purpose, like [QBullets](#). QBullets are a collection of elegant, animated icons that attach to hypertext links to indicate their function. You can [download QBullets](#) for free from matterform media.

To use this idea for footnotes, the name attribute is used as a selector:

```
/* Add a note icon to footnote links */
a[href^="#FTN"] {
    background: url(images/qbullet-note.gif) right center no-repeat;
    padding-right: 12px;
}
a[href^="#FTN"]:hover {
    background: url(images/qbullet-note_a.gif) right center no-repeat;
}
}
```

This will select all links whose href attribute starts with “#FTN” and append a note icon to them, which will even show an animated page curl upon passing with the mouse over it (hover). The links whose href attribute starts with “#FTN” look like

```
<a name="AEN1175" href="#FTN.AEN1175">[1]</a>
```

and point to a footnote. The footnote itself also contains a link - which points back to the referring text. That link will not be affected by the above selection, since its href attribute does not start with “#FTN”:

```
<a name="FTN.AEN1175" href="explain-runsed-sed-sedscr.html#AEN1175">[1]</a>
```

To display a back icon besides those links, a selector on the name attribute is used:

```
/* ...and a back icon to the backlinks in the footnotes themselves */
a[name^="FTN"] {
    background: url(images/scrollup.gif) right center no-repeat;
    padding-right: 12px;
}
a[name^="FTN"]:hover {
    background: url(images/scrollup_a.gif) right center no-repeat;
}
}
```



String matching on attributes is a CSS3 feature!

To be able to use string matching on attributes, as we have done in the QBullets examples above, the user must view our document with a browser that supports this CSS3 feature. If you are wondering whether your browser belongs to this cutting edge category (Mozilla 1.5 does, tip, tip 😊), you can do this [W3C browser test on CSS selectors](#) and test for "Substring matching attribute selector".

To get an icon in place of the usual bullet in itemized lists, the list-style property is used for the UL tag:

```
UL {
    margin-bottom: 10px;
    list-style: url(images/tux-bullet.png) square;
}
```

Last but not least, a cross-browser relative font setting can be achieved with

```
P {
    font-size: 12px;
}
/**/A{}
BODY P {
    font-size: x-small;
    voice-family: "\"}\"";
    voice-family: inherit;
    font-size: small;
}
HTML>BODY P {
    font-size: small;
}
/* */
```

which is indeed too complicated to explain here in all its depth. See [Dive into Accessibility, day 26](#) for this.

Chapter 5. Transparent proxying with Squid

Figure 5-1. Squid.



Squid.

I assume you have already set up a [Squid](#) proxy and a firewall and that both work correctly. You may now wish to force all your users to use the Squid proxy for surfing the WWW. This is what "transparent proxying" is about: your users surf, even without having defined a proxy in their browser settings, but they in fact all use the transparent proxy and don't notice it.

To enable transparent proxying with Squid, insert the following lines in the configuration file (squid.conf, usually in /etc) at the appropriate place (search the configuration file for the respective keywords, httpd_accel_host, httpd_accel_with_proxy and httpd_accel_uses_host_header):

```
httpd_accel_host virtual
httpd_accel_with_proxy on
httpd_accel_uses_host_header on
```

You will also need to accept and redirect the WWW traffic to port 3128 of the Squid proxy:

```
ipchains -A input -p TCP -d 127.0.0.1/32 www -j ACCEPT
ipchains -A input -p TCP -d 192.168.0.0/32 www -j ACCEPT
ipchains -A input -p TCP -d any/0 www -j REDIRECT 3128
```

or, if you use SuSEfirewall2

```
FW_REDIRECT_TCP="192.168.0.0/24,0/0,tcp,80,3128"
```

Restart Squid and the firewall. Transparent proxying should be working now. However there are some issues associated with the above settings. You can read about them in the corresponding comments in the squid.conf file.

For more details, see the [Transparent Proxy HOWTO](#).

Chapter 6. LyX

Figure 6-1. LyX.



LyX.

In case you are wondering what LyX is, here is what <http://www.lyx.org> says on the subject:

LyX is an advanced open source document processor that encourages an approach to writing based on the structure of your documents, not their appearance. LyX lets you concentrate on writing, leaving details of visual layout to the software.

LyX runs on many Unix platforms, OS/2, and under Windows/Cygwin (this port requires an X server). It can also run natively on Mac OS X, thanks to the Qt/Mac library.

LyX produces high quality, professional output – using LaTeX, an industrial strength typesetting engine, in the background; LyX is far more than a front-end to LaTeX, however. No knowledge of LaTeX is necessary to use LyX, although it will give a user more power.

LyX is stable and fully featured. It has been used for documents as large as a thesis, or as small as a business letter. Despite its simple GUI interface (available in many languages), it supports tables, figures, and hyperlinked cross-references, and has a best-of-breed math editor.

6.1. Mass insertion of cross-references in LyX

If you have to add hundreds of cross-references in just one section (e.g. more than 500, as in [Credits for version 2.0 of the PHP-Nuke HOWTO](#)), you will soon notice that, although a single cross-reference is inserted very easily in LyX (just choose Insert->Cross-reference from the menu, then choose the label of the reference you want), it becomes a real pain if you have to enter hundreds of them.

My solution to this was to write a script that reads a LyX file and outputs another LyX file that contains references to all labels of the first one. It was then easier to copy the references from the file thus created, paste them in [Credits for version 2.0 of the PHP-Nuke HOWTO](#) and then delete the unneeded ones, than try to insert all cross-references by hand using the LyX menu.

The following script, call it `lyxrefs`, will print a LyX file in standard output, containing cross-references to each and every label of the LyX file whose name was passed on the command line as argument:

```

#!/bin/bash
#
AWK="/usr/bin/awk"
function preamble() {
cat █-EOF
#LyX 1.2 created this file. For more info see http://www.lyx.org/
\lyxformat 220
\textclass article
\language english
\inputencoding auto
\fontscheme default
\graphics default
\paperfontsize default
\papersize Default
\paperpackage a4
\use_geometry 0
\use_amsmath 0
\use_natbib 0
\use_numerical_citations 0
\paperorientation portrait
\secnumdepth 3
\tocdepth 3
\paragraph_separation indent
\defskip medskip
\quotes_language english
\quotes_times 2
\papercolumns 1
\papersides 1
\paperpagestyle default
EOF
}
function label() {
n=$1
echo ""
echo "\layout Standard"
echo ""
echo ""
echo "\begin_inset LatexCommand \label{cit:$n}"
echo ""
echo "\end_inset"
}
preamble
# Output LyX commands for Chapter "All references"
cat █-EOF
\layout Section
All references
EOF
# Output all references.
$AWK 'BEGIN {FS=" "} /\begin_inset LatexCommand \\label{/ {gsub("label","ref");
printf("\n%s\n\n%s%s\n\n%s\n","\layout Standard",
"\begin_inset LatexCommand",$3,"\end_inset")}' $1 > all-references.tmp
cat all-references.tmp
rm all-references.tmp

```

```

# Output LyX commands for Chapter "All references"
cat ■-EOF
\layout Section
All figure references
EOF
# Output only the figures.
$AWK 'BEGIN {FS=" "} /\begin_inset LatexCommand \\label{fig-/ {gsub("label","ref");
printf("\n%s\n\n%s%s\n\n%s\n", "\\layout Standard",
"\\begin_inset LatexCommand", $3, "\\end_inset")}' $1 > fig-references.tmp
cat fig-references.tmp
rm fig-references.tmp
# Output LyX commands for Chapter "All references"
cat ■-EOF
\layout Section
All table references
EOF
# Output only the tables.
$AWK 'BEGIN {FS=" "} /\begin_inset LatexCommand \\label{tab-/ {gsub("label","ref");
printf("\n%s\n\n%s%s\n\n%s\n", "\\layout Standard",
"\\begin_inset LatexCommand", $3, "\\end_inset")}' $1 > tab-references.tmp
cat tab-references.tmp
rm tab-references.tmp
echo "\the_end"

```

The script will even create three sections, with cross-references to all labels, all figures and all tables respectively. If you named it `lyxrefs`, you would call it as follows:

```
lyxrefs some-LyX-file.lyx > refs.lyx
```

Then `refs.lyx` will contain cross-references to all labels of `some-LyX-file.lyx`. You can open `refs.lyx` with LyX, copy all or part of the cross-references there and paste them in `some-LyX-file.lyx`. The cross-references in [Credits for version 2.0 of the PHP-Nuke HOWTO](#) were entered this way.

6.2. Automatic Index generation

LyX provides an easy way to insert an Index entry: from the menu, choose either `Insert->"Index entry of preceding word"` (which I personally find easier), or `Insert->"Index entry"`, then enter the required word. This method works fine - if you have a small document, with only a few keywords to insert. But what if your document has grown to hundreds of pages, with hundreds (or even thousands) of index entries to insert? See the Index of the [PHP-Nuke HOWTO](#) for an example of an Index that cannot be generated manually - unless you want to drive yourself crazy!

Clearly, for a comprehensive Index of large documents, an automatic procedure is necessary. However, the general problem of automatic Index generation is subject of extensive (and still not conclusive) research and I am not going to address it in its full generality here. For our purposes, even a semi-automatic procedure would be very helpful. To this end, I have created the following 4 scripts:

- [sedscr_ist_index_items](#): lists all index entries contained in a LyX document.

- `sedscr_delete_index_items`: deletes all index entries from a LyX document.
- `awkscr_create_index_items`: creates a list of words used in a LyX document. The list can be subsequently edited manually, mostly deleting unwanted or uninteresting words, to yield a list of words that are used in the document and are interesting enough to be part of its Index.
- `awkscr_insert_index_items`: uses an externally supplied document containing a list of index entries to insert an index entry in a LyX document for every word appearing in that list.

They can be used in the following semi-automatic Index generation procedure:

1. Optional: create a list of all existing index entries in your document. This is useful not only because you are going to eliminate all index entries from the document in the next step, but also as a backup of the index entries that were currently in use - you might want to reuse them in some later step.

To create a list of all existing index entries in your document, type:

```
sedscr_list_index_items document.lyx > indexitems
```

The generated `indexitems` file will contain a list of all index entries in `document.lyx`, one index entry per line.

2. Remove all previous index entries from the LyX document. You need this preliminary step because, if you forget to remove already existing index entries, a subsequent run of the `awkscr_insert_index_items` script may substitute even the existing index terms (those already inside the LyX `\index` commands) with LyX `\index` commands. This may or may not happen, depending on the regular expressions used in the current implementation of `awkscr_insert_index_items`, but it is better to err on the side of caution. Besides, a LyX text cluttered with index entries may still be a breeze to read for a computer, but quite a headache to read for humans.

To remove all index entries from a LyX document, type:

```
sedscr_delete_index_items document.lyx > document-noindexitems.lyx
```

The generated `document-noindexitems.lyx` will contain everything from `document.lyx` - except the index entries.

3. Create a list of all index entries to be used in the LyX document. This is the most difficult part: as said above, this problem is not trivial. We will thus content ourselves with a list of all *words* used in the document. Once we have all words, we can still edit the list manually and delete all unwanted entries. This is what makes this procedure *semi-automatic* and not automatic. The idea is that it is still better having to delete 10000 lines from a 12000 line document, than having to insert 2000 index entries from the LyX Insert menu.

To create a list of all words used in a LyX document, type:

```
awkscr_create_index_items document.lyx > words
```

There is even some code in `awkscr_create_index_items` that checks whether the current word is in some "trivia" list of trivial words and discards it. In such a case, you would call the script with two arguments, as follows:

```
awkscr_create_index_items trivia document.lyx > words
```

However, this part of the code is either too slow, or buggy, so it is commented for the moment (feel free to send corrections or suggestions).

Once the list of all words of your document is created, all you have to do is open it with a text editor and delete all unwanted words or correct the ones that are in plural or have some punctuation at the end and so on. This is still hard if your document is large, but still a faster alternative than targeting the Insert menu with the mouse 8000 times (I guess each one of my 2000 index entries appears 4 times in my document, which gives me an estimate of 8000 menu selections with the mouse - unfortunately no keyboard bindings were found to work on my system).

4. Once you have a file, say `indexitems`, with all words that should appear in the Index of a LyX file, type:

```
awkscr_insert_index_items indexitems - < document-noindexitems.lyx > document-indexitems.lyx
```

to create from `document-noindexitems.lyx` a document with index entries (`document-indexitems.lyx`) for all words in `indexitems`.

Some notes on `awkscr_insert_index_items`'s mode of operation:

The `-` in the above invocation is important: it forces the `awk` script to continue reading from standard input, after it has read `indexitems`. This, together with the code

```
FILENAME == "indexitems" {
    n++
    indexentry[$1] = $1
    next
}
```

in `awkscr_insert_index_items`, causes the words in `indexitems` to be imported into the `indexentry[]` associative array.

The file separator in `awkscr_insert_index_items` is set to the semicolon `;`, instead of the default, which is space. This makes it possible to enter index entries with more than one words. Accordingly, the `awkscr_create_index_items` script appends a semicolon at the end of each word it prints.

`awkscr_insert_index_items` follows a simple algorithm to insert the index entries at the right places in the document: to insert an index entry, we have to know what **LyX environment** we are in. In essence, this means we have to parse the LyX document. Since the `\layout` commands in the LyX file do NOT have what we would call "closing tags" in other markup languages, we cannot tell `awk` "if you are between the start and the end of the Paragraph environment, do the following", or anything like that - there is no easy way to find the "end" of an environment, given all the environment nestings that are possible. Luckily, another easy way exists: whenever a `\layout` command is encountered, we are in the environment specified by that `\layout` command, so we only need to set a variable, call it `layout`, accordingly:

```
/\\layout SGML/ { layout = "SGML"; print; next }
/\\layout Chapter/ { layout = "Chapter"; print; next }
```

```

/\layout Section/ { layout = "Section"; print; next }
/\layout Subsection/ { layout = "Subsection"; print; next }
/\layout Subsubsection/ { layout = "Subsubsection"; print; next }
/\layout Standard/ { layout = "Standard"; print; next }
...and so on

```

Clearly, we should not insert index entries everywhere, e.g. in the "Code" environment. That's why we check if we are in the "Standard", "Itemize", "Quotation", "Description" environment (warning: the way `sedscr` works currently, you should not insert index entries in the "Caption" environment) and, if we are (and only then), we substitute every word in the `indexentry[]` array with the LyX "insert index entry" command:

```

{
    if (layout == "Standard" || layout == "Itemize" || layout == "Quotation"
|| layout == "Description" ) {
        for (item in indexentry) {
            if (gsub(item "$", item "\n\\begin_inset LatexCommand
\\index{" indexentry[item] "}\n\n\\end_inset \n")) { print; next }

                ...other substitutions here
            }
        }
    }
}

```

Some tips regarding the (necessary) manual editing of the words file, the file output by `awkscr_create_index_items` above:

- You will see a lot of words (or their declinations) that are not useful. It is one thing to have a lot of words and another to have a set of really useful words and phrases. That's the price we pay for the simplicity of our method.
- You may need to supply some extra terms you feel are missing from that file. Feel free to do this, `awkscr_insert_index_items` does not know how you created the `indexitems` file you give it.
- Keep backups of your word lists from subsequent runs of the scripts. Combine word lists from other projects. No matter how long your word list, only the terms that really appear somewhere, will make it to the Index, so don't worry if your list is too long - given enough computing time, that is.
- Take care to delete everything in your word list that looks like a regular expression with metacharacters - because it will be interpreted as such, with unpredictable results (unless you really know what you are doing). I once had `".*"` on one line and I forgot to delete it. I then wondered how come that my document was full of index entries to `".*"` while the text was almost gone! See [regular expressions](#), for a brief introduction to regular expressions.
- Take out any `":", ";", "?"` from the end of the words, as well as enclosing double quotes. Those characters are already taken care of when it comes to inserting the entries, i.e. the `indexentries` file should contain only the "pure" words, without any punctuation signs.
- Don't leave in "config" if your LyX file contains "config.php". If you do, the latter will look ugly in the LyX editor, as it will contain an index entry for "config" just in the middle of it. This will not affect the rendered formats, however.

- Don't leave in words that might form parts of a LyX command. I once left "Enumerate" in my word list. The resulting LyX file contained an index entry for "Enumerate" in front of every item in every enumeration list! Clearly, the awk script "sees" the LyX commands in the file that are invisible to you.

Appendix A. Appendix

A.1. The GNU Free Documentation Licence

This is an exact copy of the [GNU Free Documentation License Version 1.2](#), November 2002:

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

A.1.1. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

A.1.2. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

A.1.3. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control

the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in [Section A.1.4](#).

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

A.1.4. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

A.1.5. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of [Section A.1.3](#) and [Section A.1.4](#) above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History

section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

A.1.6. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in [Section A.1.5](#) above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

A.1.7. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

A.1.8. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of [Section A.1.4](#) is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

A.1.9. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of [Section A.1.5](#). Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement ([Section A.1.5](#)) to Preserve its Title ([Section A.1.2](#)) will typically require changing the actual title.

A.1.10. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

A.1.11. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the

option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

A.1.12. ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index

Symbols

GNU Free Documentation License, [2](#)

A

Apache, [6](#)
automatic, [26](#)

B

buffer, [??](#)

C

capture, screen, [8](#)
capture, tool, [8](#)
Cascading Style Sheet, [11](#)
Communicator, [5](#)
convert, [8](#)
CSS, [11](#)

D

Distiller, [9](#)
DocBook, [11](#)

F

Free Software Foundation, [2](#)

G

Ghostscript, [9](#)

H

highlighting, [3](#)
HTML (HyperText Markup Language), [1](#)
HTML (HyperText Markup Language)], [1](#)
HTML sessions, [9](#)
HTML to PNG, [8](#)
html2ps, [8](#)
htpasswd, [7](#)
httpd_accel_host, [22](#)
httpd_accel_uses_host_header, [22](#)
httpd_accel_with_proxy, [22](#)

I

Index, [26](#)
invocation, remote, [10](#)

L

LYX (LaTeX frontend LyX)], [1](#)

M

markup, [28](#)
MultipleSearch, [3](#)

N

nestings, [28](#)
Netscape, [5](#)

O

Opera, [8](#)

P

PDF (Portable Document Format)], 1
Perl script, 8
plugins (vi), 4
plugins (vi), syntax, 4
printing, 8
profile, roaming, 7
proxying, transparent, 22
PS.GZ (Compressed Postscript)], 1
ps2png, 9

R

redirect, 22
remote, invocation, 10
roaming profile, 6
RTF (Rich Text Format)], 1

S

scroll, 8
separator, 28
sessions, 10
SGML, 11
SGML (Standard Generalized Markup Language)], 1
Squid, 22
SuSEfirewall2, 22

T

TXT (ASCII Text)], 1

V

vi, 3
vimrc, 3

X

X terminal, 3
xterm, 3
xwd, 8
xwininfo, 8