The LyX User's Guide

by the LyX Team[1]

August 30, 1999

[1] This file is temporarily maintained by Mike Ressler, mike.ressler@alum.mit.edu. Please send comments or error corrections to that address.

Contents

1 Introduction 9

1.1 What Is LyX? . . . 9
1.2 About This Document . . . 10
1.3 Getting Started . . . 11
1.3.1 Invoking LyX . . . 11
1.3.2 How LyX Looks . . . 11
1.3.2.1 The Main Window . . . 11
1.3.2.2 Popups . . . 11
1.3.3 HELP! . . . 12
1.4 The LyX Interface . . . 13
1.4.1 Basic File Operations . . . 13
1.4.2 Basic Editing Features . . . 14
1.4.3 Undo and Redo . . . 14
1.4.4 Basic Mouse Bindings . . . 15
1.4.5 Basic Key Bindings . . . 16
1.5 Using LyX with Other Programs . . . 17
1.5.1 Importing ASCII files . . . 17
1.5.2 Cut and Paste Between LyX and Other X Programs . . . 18

2 LyX Setup and Supporting Applications 19

2.1 Introduction . . . 19
2.2 Requirements . . . 19
2.3 Basic LyX Setup . . . 20
2.4 Setting Up the X Keyboard . . . 21
2.4.1 `xmodmap` and `xkeycaps` . . . 21
2.4.1.1 `xmodmap` . . . 21
2.4.1.2 `xkeycaps` . . . 21
2.4.2 Modifiers and Mode_switch . . . 22
2.4.3 Helpful Hints and Tips . . . 22
2.5 LaTeX . . . 23
2.6 Dvips and Ghostscript . . . 23
2.6.1 What You Need . . . 23
2.6.2 Dvips . . . 24
2.6.3 Ghostscript, Xdvi and Ghostview . . . 25

2.7 The Printer 26

3 LyX Basics **27**

3.1 Document Types 27
3.1.1 Introduction 27
3.1.2 The Various Document Classes 27
3.1.2.1 Overview 27
3.1.2.2 Selecting a Class 29
3.1.2.3 Properties 29
3.1.3 Fine-Tuning the Defaults 29
3.1.4 Paper Size, Orientation, and Margins 31
3.1.5 Important Note: 31
3.2 Paragraph Indentation and Separation 31
3.2.1 Introduction 31
3.2.2 Global Indentation Method 32
3.2.3 Fine-Tuning 32
3.2.4 Changing Line Spacing 32
3.3 Paragraph Environments 33
3.3.1 Overview 33
3.3.2 Standard 33
3.3.3 Document Titles 34
3.3.4 Headings 34
3.3.4.1 Numbered Headings 34
3.3.4.2 Unnumbered Headings 35
3.3.4.3 Changing the Numbering 35
3.3.4.4 Special Information 36
3.3.4.5 Creating an Appendix 36
3.3.5 Quotes and Poetry 37
3.3.5.1 Quote and Quotation 37
3.3.5.2 Verse 37
3.3.6 Lists 38
3.3.6.1 General Features 38
3.3.6.2 Itemize 39
3.3.6.3 Enumerate 40
3.3.6.4 Description 41
3.3.6.5 The LyX List 41
3.3.7 Letters 43
3.3.7.1 Address and Right Address: An Overview 43
3.3.7.2 Usage 43
3.3.8 Academic Writing 44
3.3.8.1 Abstract 44
3.3.8.2 Bibliography 45
3.3.9 Special Purpose 45
3.3.9.1 Caption 45
3.3.9.2 LyX-Code 46
3.3.9.3 Comment 47

3.3.9.4 LaTeX . . . 47
3.4 Nesting Environments . . . 47
3.4.1 The Big Deal . . . 47
3.4.2 What You Can and Can't Nest . . . 48
3.4.3 Nesting Other Things: Tables, Math, Floats, etc. . . . 49
3.4.4 Usage and General Features . . . 51
3.4.4.1 The Various Bindings . . . 51
3.4.5 Some Examples . . . 52
3.4.5.1 Example #1: The Sixfold-Way and Mixed Nesting 53
3.4.5.2 Example #2: Inheritance . . . 54
3.4.5.3 Example #3: Labels, Levels, and the Enumerate and Itemize Environments. . . . 54
3.4.5.4 Example #4: Going Bonkers . . . 55
3.5 Fonts and Text Styles . . . 56
3.5.1 Overview . . . 56
3.5.2 Global Options . . . 57
3.5.3 Using Different Character Styles . . . 57
3.5.4 Fine-Tuning with the Character Layout Popup . . . 58
3.6 Printing and Previewing . . . 61
3.6.1 Overview . . . 61
3.6.2 Quick Viewing with `xdvi` . . . 62
3.6.3 Viewing the PostScript® Version with `ghostview` . . . 63
3.6.4 Printing the File . . . 63
3.7 A Few Words about Typography . . . 64
3.7.1 Hyphens and Hyphenation . . . 64
3.7.2 Punctuation Marks . . . 65
3.7.2.1 Abbreviations and End of Sentence . . . 65
3.7.2.2 Quotes . . . 66
3.7.3 Ligatures . . . 66
3.7.4 Widows and Orphans . . . 67

4 Floats: Tables, Figures, Footnotes and Margin Notes 69
4.1 Footnotes . . . 69
4.2 Margin Notes . . . 70
4.3 Figures and Imported Graphics . . . 70
4.3.1 Figure Floats . . . 72
4.3.1.1 Using Figure Floats . . . 72
4.3.1.2 Float Placement . . . 73
4.3.2 `XFig` and LyX . . . 75
4.4 Tables . . . 76
4.4.1 The Table Layout pop-up . . . 77
4.4.2 What can be placed inside a table? . . . 78
4.4.3 Cut & Paste & Tables . . . 79
4.4.4 Multiple lines in cells . . . 79
4.4.5 Table Floats . . . 81
4.4.5.1 Table Float Captions. . . . 81

4.4.5.2 Float Placement 82
4.5 Table of Contents and other Listings 82
4.5.1 The Table of Contents 83
4.5.2 List of Figures, Tables and Algorithms 83

5 Mathematical Formulas 85
5.1 Basic Math Editing 85
5.1.1 Invoking Mathed 86
5.1.2 Navigating an Equation 86
5.1.3 Selecting Text 87
5.1.4 Exponents and Subscripts 87
5.1.5 Fractions 87
5.1.6 Sums and Integrals 88
5.1.7 The Math Panel 88
5.1.8 Other Math Symbols 89
5.1.9 Math functions 90
5.1.10 More symbols 90
5.1.11 Accents 90
5.2 Brackets and decorations 91
5.3 Arrays and Multi-line Equations 92
5.4 Equation Numbering and Labels 93
5.5 User defined macros in math mode 94
5.5.1 Directions on using macros 95
5.5.1.1 How to create them 95
5.5.1.2 How to navigate 95
5.6 Fine-Tuning 95
5.6.1 Typefaces 95
5.6.2 Font Sizes 96

6 More Tools 99
6.1 Cross-References 99
6.2 Manual Fine-Tuning 100
6.2.1 Extra Horizontal Space 100
6.2.2 Extra Vertical Space 101
6.2.3 Changing Paragraph Alignment 101
6.2.4 Forcing Page Breaks 101
6.2.5 Other Tools 102
6.3 Spellchecking 102
6.3.1 Spellchecker Options 103
6.3.1.1 Dictionary 104
6.3.1.2 Personal dictionary 104
6.3.1.3 Further Options 105
6.3.2 Problems 105
6.3.3 Limitations 105
6.4 International Support 105
6.4.1 Layout->Document Language Options 106

6.4.2 The Options->Keyboard Keymap Configuration 106
6.4.3 Character Tables 107
6.4.4 International Spellcheck Support 109

7 LyX and LaTeX 111
7.1 How LyX Uses LaTeX 111
7.2 "Help! LyX generated an unreadable `.tex` file!" 112
7.3 Translating LaTeX files into LyX 113
7.4 Inserting LaTeX Code into LyX Documents 113
7.5 LyX and the LaTeX Preamble 115
7.5.1 About the LaTeX Preamble 115
7.5.2 Changing the Preamble 116
7.5.3 Examples 117
7.5.3.1 Example #1: Offsets 117
7.5.3.2 Example #2: Labels 117
7.5.3.3 Example #3: Paragraph Indentation 118
7.5.3.4 Example #4: This Document 118
7.6 LyX and LaTeX Errors 118

8 Credits 123

Chapter 1

Introduction

1.1 What Is LyX?

LyX is a program that provides a more modern approach to writing documents with a computer, an approach that breaks with the obsolete tradition of the "typewriter concept." It is designed for authors who want professional output quickly with a minimum of effort without becoming specialists in typesetting. Compared to common word processors, LyX will increase productivity a lot, since the job of typesetting is done mostly by the computer, not the author. With LyX, the author can concentrate on the contents of her writing, since the computer takes care of the look.

Technically this is done by combining the comfortable interface of a word processor with the high quality of a real typesetting system. LyX uses the most popular and, in our opinion, best typesetting system available: LaTeX. LaTeX is used for a wide range of documents, especially in science. For example, it's difficult to find a mathematics or computer science book that is not done with LaTeX. So, some people claim that its main purpose is mathematical typesetting. This isn't true. LaTeX is equally good for writing letters, articles, books, or any other kind of document, and does so much better than common word processors. What prevents some people from using this powerful, free typesetting system, one that is available for almost every computer system, is its difficult usage. With plain LaTeX, you need to enter a series of typesetting commands into the text in order to produce your document. As a result, you get no visual feedback until you feed your document to the LaTeX program. It's also difficult to read these documents before they have been printed. So, online editing isn't very easy. This is where LyX enters the game.

LyX provides an "almost-WYSIWYG" view of the document. "Almost" means that the line- and page-breaks are not displayed exactly as they will appear in the printed document. However, that's not really necessary, since LyX uses a separate typesetter program [here, LaTeX] to perform the final formatting of your text. While LyX contains everything it needs to be a comfortable user interface, the typesetting program contains everything necessary to format text, and do so very, very well. There's no need to reinvent the wheel, after all. Besides, computers are best at following a set of rules, and doing so repeatedly and consistently. Why should you do extra work remembering which subsubsection in which section in which chapter you're in, what numbering scheme your using, how big the different headings are, what font you used for the different types of headings, and so on, and so on, and so on, ... when a computer can do all of that for you? The answer is simple: you shouldn't, and with LyX, you don't have to. So, line- and page-breaks aren't your problem anymore. Remembering which number to use for the next subsection isn't your problem anymore. Recalling what font you used for all of your section headings isn't your problem anymore. With LyX, you simply choose a so-called "paragraph environment." That's it. You're done.

This gives you far more power than you may think. No longer do you need to scroll through a 75 page document, changing all of the section numbers because you deleted an old section. You could even pick a section, heading and all, up out of one document and drop it in a new one. LyX does the renumbering for you, adds the section to the Table of Contents, and more! Because you tell LyX [and LaTeX] what *kind* of document you're editing and what *type* of paragraph this-or-that text is, the computer can typeset it accordingly. Cut some paragraphs from an old document [say, an article] and paste them in a completely different one, [say, a letter] and LyX does the rest. Of course you can also still do some low-level formatting for fine-tuning. However, the proper way with LyX is to tell the computer what the text *is*, not what it should look like. So, we like to say that LyX gives you WYSIWYM editing [What You See Is What You *Mean*].

Some people might be tempted to call LyX a "frontend to LaTeX." This isn't quite fair. LyX performs some typesetting internally to generate the correct look on the screen. Furthermore, LyX has some extensions to LaTeX specially designed to work with the WYSIWYM-concept. So, it's actually better to call LyX a *High Level Wordprocessor* that uses LaTeX as its backend.

1.2 About This Document

If you haven't read the *"Introduction"* , you need to. Yes, we mean now.

The "*Introduction*" describes several things, most importantly, the format of all of the manuals. If you don't read it, you'll have a bear of a time navigating this manual. You might also be better served looking in one of the other manuals instead of this one. "*Introduction*" describes that, too.

A full list of credits, containing of the LyX Documentation Project member, appears at the end of this file.

1.3 Getting Started

1.3.1 Invoking LyX

Similar to other Linux [and other brands of Unix] programs, you start LyX by simply typing `lyx` at the command line. You can, of course, include several command-line options, including file names. We're not going to repeat all of the command-line options here, since we've already done that in the `man` page for LyX. Check there for more info.

There are one or two things we'd like to comment on:

- If you include more than one file name on the command line, LyX will load them all, though it won't display them all simultaneously. More on that in a bit.
- Many of you X-perts out there will try, in vain, to use the `-geometry` option or its cousin, which has the form: `-80x24+5+5`. For now, there is a workaround via the options `-height`, `-width`, `-xpos` and `-ypos`.

1.3.2 How LyX Looks

1.3.2.1 The Main Window

Like most applications, LyX has the familiar menu bar across the top of its window. Below it is a toolbar with a pulldown box and various buttons. There is, of course, a vertical scrollbar and a main work area for editing documents. At the bottom of the window is a small window containing a single line of text. This is the *minibuffer* [a term which we've swiped, lock, stock, and barrel, from GNU Emacs]. It not only displays status information, but takes input, as well.

Note that there is no horizontal scroll bar. This is not a bug or an oversight, but intentional. When you read a book, you expect the end of a line to wrap around to the next line. Text overflows onto new pages in a vertical fashion, hence the need for only a vertical scrollbar.

There are three cases where you might want a horizontal scrollbar. The first case is large figures, displayed WYSIWYG. This, however, is due to a flaw in the routine that displays graphics on the LyX screen in a WYSIWYG fashion; it should rescale the graphics to fit in the window, just as you'd need to rescale graphics to fit on a page. The second and third cases are tables and equations which are wider than the LyX window. That is indeed a problem, one which will be resolved in a future version of LyX.

1.3.2.2 Popups

Ever heard of a "dialog box?" We LyXers like to call them popups windows, or popups for short. Almost all of the LyX popups have three basic buttons: OK, Apply, and Cancel. The first and last ones are self-explanatory. The Apply button works similarly to the OK button, in that it activates whatever changes you made in the popup; however, it doesn't close the popup, like OK does.

Actually, pressing the OK button is exactly the same as pressing Apply to commit the changes and then Cancel to close the popup. The three buttons also have shortcut keys:

- OK is bound to the Return key, as in most programs
- Apply is bound to M-a [as you might expect]
- Cancel is bound to the Esc key.

There are some popups that have different buttons. Not to worry, they're typically self-explanatory. We just wanted to document somewhere what that Apply does.

1.3.3 HELP!

First, the bad news: the help system is not as thorough or idiot-proof as in many commercial applications. Patience. We're working on it.

Now the good news: the help system consists of the LyX manuals. You can read *all* of the manuals from inside LyX. Just select the manual you want read from the Help menu.

While we're at it, we'd like to make a comment about the manuals. They're not idiot-proof, not in the least. Here's what one of our authors, John Weiss, once said about manuals:

> I hate manuals.
>
> Yes, we've all dealt with the terse, poorly-translated, or cryptic manuals. They are aggravating. I find, however, that the overly simplified ones are even more aggravating. First, they spend about half their time carefully explaining to the user how to operate a mouse, what a menu is, et cetera, et cetera, ad nauseum. Please, if someone doesn't know how to use their own computer, or a GUI, then they should sit down and learn *before* they start up a major piece of software.
>
> Second, what information they do provide seems to assume that the user is stupid. Utter nonsense! Most users, in my experience, are some combination of clueless and intimidated, not stupid. Besides, if someone is truly slow on the uptake, they need help that a manual for a piece of computer software can't give.

Editor's Note: With this in mind, I've instructed all of the other authors to avoid patronizing you, the reader, and to be more pedagogical than pedantic. As for those who are too lazy to read and understand the manuals — well, as we say here in America, there's no such thing as a free lunch. - jw

1.4 The LYX Interface

1.4.1 Basic File Operations

Under the File menu are the 9 basic operations for any word processor:

- New
- New from template
- Open
- Close
- Save
- Save As
- Revert to saved
- Print
- Exit

They all do pretty much the same thing as in other word processors, with a few minor differences. The File->New from template command not only prompts you for a name for the new file, but also prompts you for a template to use. Selecting a template will automatically set certain layout features for the document, features you would otherwise need to change manually. They can be of use for certain classes, especially those for writing letters [see sec. 3.1.2]

Note: There is no “default file” or document named “Untitled” or “scratch.” Unless you tell LYX to open a file or create a new one, that big, blank space is just that — a big, blank space.

The Revert to saved command is useful if more people work on the same document at the same time[1]. It will simply reload the document from disk. You can of course also use it if you regret that you changed a document and want to restore it to the last save.

The second matter of note concerns the commands File->Close and File->Exit. They both feature a “nag box” to save us all from our own stupidity. That is, if you try to close a file with changes [or exit LYX], you'll be informed that there are unsaved files.[2]

[1] If you plan to do this, you should check out the Version Control feature in LYX also. Read *Extended Features*.

[2] Some of you will know that previous versions of LYX would crash if you tried to kill the window from the window manager [i.e. double-clicking the upper left corner]. This was fixed in 0.12, so it's safe to do that now.

1.4.2 Basic Editing Features

Like most modern word processors, LyX can perform cut and paste operations on blocks of text, can move by character, word, or screenful of text, and can delete whole words as well as individual characters. The next four sections cover the basic LyX editing features and how to access them. We'll start with cut and paste.

As you might expect, the Edit menu has the cut and paste commands, along with various other editing features. Some of these are special and covered in later sections, such as the Floats & Insets submenu. The basic ones are:

- Cut
- Copy
- Paste
- Find & Replace

The first three are self-explanatory. One thing to note: whenever you delete a block of text that you've selected, it's automatically placed in the clipboard. That is, the Delete and Backspace keys also functions as the Cut command. Also, if you've selected text, be careful. If you hit a key, LyX will completely delete the selected text and replace it with what you just typed. You'll have to do an Undo to get back the lost text.

The Edit->Find & Replace item opens the Find & Replace popup. LyX does its find and replace work a little bit differently than certain commercial programs. Obviously, the text you want to find goes in the Find box. The two arrow buttons search backward and forward, respectively [it's obvious which does which].[3] Once you've found a word or expression, LyX selects it. Hitting the Replace button replaces the selected text with the contents of the Replace with box.

Let's state that again: you can't do a replace unless you've done a find first.

To repeat a replace, click the arrow key to find the next/previous word, then hit the Replace button. There is no operation to do a "replace all". [At least not yet...]

The Case sensitive toggle button can be used if you want the search to consider the case of the search word. If the toggle is set, searching for "`Match`" will not match the word "`match`".

The Match Word toggle button can be used to force LyX to only find complete words. I.e. searching for "`match`" will not match "`matches`".

1.4.3 Undo and Redo

If you make a mistake, you can easily recover from it. LyX has an infinite-capacity undo/redo buffer. Select Edit->Undo to undo some mistake. If you

[3]These two arrows can be accessed from the keyboard via the shortcuts M-f and M-b for forward and back.

accidently undo too much, use Edit->Redo to "undo the undo." *The undo-mechanism has been limited to 100 steps in the beta-version, as this feature has not yet been properly tested out.*

You can't use Redo beyond the most recent change. Likewise, you can't Undo beyond the first change you made after opening the file. In such situations, the corresponding menu entries are grayed out. Notice that if you revert back all changes to arrive to the document as it was last saved, the "changed" status of the document is unfortunately not reset. This is a known bug.

The Undo and Redo work on almost everything in LyX. They have some quirks, too. They won't Undo or Redo text character by character, but by blocks of text. That can take some getting used to; you'll have to play with Undo and Redo to get a feel for just how much they'll undo/redo, and after time, you'll hopefully appreciate how it works. Also, Redo does not work inside of math mode. Tip: if you enter Math mode, change an equation, then leave math mode, Undo will revert to the old version of the equation, the one which existed before you entered Math mode.

1.4.4 Basic Mouse Bindings

We're not going to go into all of the mouse bindings here. Some of the other sections of this manual cover specific operations you can do with the mouse. Instead, we're going to cover the most basic mouse operations.

1. Motion
    - Click the *left mouse button* once anywhere in the edit window. The cursor moves to the text under the mouse.
    - The "Scroll Bar"

      The scrollbar in LyX works pretty much as any other scrollbar around, except that we've enhanced it a bit. The scrollbar arrow buttons can do more than meets the eye:
        - Use the *left mouse button* to scroll by one line in the direction of the arrow.
        - Use the *right mouse button* to scroll a page in the *same* direction as the arrow.
        - Use the *middle mouse button* to scroll a page in the *opposite* direction of the arrow.
    - References

      Click on the *left mouse button* to jump to the referenced section.
2. Selecting Text
    - Hold down the *left mouse button* and drag the mouse. LyX selects the text between the old and new mouse positions. This is similar to how other programs work.

- Click the *middle mouse button* to paste text. LyX repositions the cursor under the mouse and puts the pasted material into the text at that position. Also, if you've selected any text, LyX copies it into the paste buffer and pastes it.

3. Footnotes, Margin Notes, Figure and Table Floats, etc.

   Single click the left mouse button to open or close any of these. Also check the appropriate section of this manual for more details.

4. Tables

   Single click the right mouse button to open a pop-up that will allow you to manipulate the table.

1.4.5 Basic Key Bindings

Again, we're not going to cover all of the keybindings. You should know how to use a keyboard by now. Besides, the *Reference manual* contains a complete list of *all* of the keybindings.

Some keys, like PageUp, PageDown, Left, Right, Up, and Down, do exactly what you expect them to do. Other keys don't:

Tab There is no such thing as a tab stop in LyX. If you don't understand this, go read Section 3.2.1 and 3.3, especially sec. 3.3.6, right now. Yes, right now. If you're still confused, look in the *Tutorial*.

Esc This is the "cancel key." It's used, generically, to cancel operations. Other parts of the manual will go into greater detail about this.

Home and End These move the cursor, respectively, to the beginning and end of a line.

Backspace and Delete *If* you have your keyboard set up correctly under the X Windows System, Backspace works as expected and Delete deletes the character to the right of the cursor [if no text is selected].

If you haven't set up your keyboard under X, or have no idea what we mean by that, go read section 2.4 immediately. You'll save yourself a lot of headaches.

Then there are the modifier keys:

Control- This has a couple of different uses, depending on which keys it's used in combination with:

- With Backspace or Delete, it deletes an entire word instead of a single character.
- With Left and Right, it moves by words instead of characters.

- With Home and End, it moves to the beginning and the end of the document, respectively.

Shift- Use this with any of the motion keys to select the text between the old and new cursor positions.

Meta- As we noted earlier, this is the Alt- key on many keyboards. This key does many different things, but it also activated the *menu accelerator keys*. If you use this in combination with any of the underlined letters in a menu or menu item, it selects that menu item. If you use the accelerator key for a menu and then hit Space, you'll open that menu.

For example, the sequence "M-e p" pastes text. Typing "M-f Space" opens the File menu.

There are also other things bound to the Meta- key, but you'll have to check in the *Reference manual* for more info.

Hopefully, you'll learn more and more keybindings and short-cut keys as you use LyX, because most mouse actions will prompt a small message in the minibuffer which describe the name of the action, you've just triggered, and any existing keybindings for that action. The notation for the keybindings is very similar to the notation used in this documentation, so you should not have any problems understanding it. However, notice that Shift-modifiers are explicitly mentioned, so "M-p S-A" means Meta-p followed by a capital A. "S-C-S" means Shift-Control-s.

1.5 Using LyX with Other Programs

1.5.1 Importing ASCII files

You can import text from an ASCII file using the Insert->Import ASCII File submenu. It contains two different options: As Lines and As Paragraphs.

Insert->Import ASCII File->As Lines puts each line of the file into its own LyX paragraph. This is useful if you're importing a text file with a simple list in it. However, if your text file contains paragraphs in it, LyX will mangle the paragraphs if you use this form of import.

Insert->Import ASCII File->As Paragraphs preserves paragraphs in text files. Often in a text file, you didn't put the contents of an entire paragraph on one line. You used Return to break up the paragraph into separate lines. Using the As Paragraphs option, LyX won't mangle such paragraphs. Anything between two consecutive blank lines goes into its own LyX paragraph. *But Remember:* you must make sure there is a *completely blank* line between each and every paragraph in your text file. If not, LyX might end up merging two paragraphs.

Use these options wisely. You have been warned!

1.5.2 Cut and Paste Between LyX and Other X Programs

The Cut, Copy, and Paste operations work only *inside* LyX. At the moment, you can't copy text from LyX to another X-Windows program. Sorry, but this is not implemented yet.

The LyX team has managed, however, to get the other direction working. That is, you can copy text from another X-Windows program *into* LyX. This special form of pasting works like the ASCII import feature; you can paste as lines or as paragraphs. Use:

- Paste Primary Selection As Lines
- Paste Primary Selection As Paragraphs

from the Edit menu to do this.

Chapter 2

LyX Setup and Supporting Applications

2.1 Introduction

If you're using LyX on a system someone else has set up for you, then you can safely skip this chapter. It describes all of the things you need beyond the LyX binary and files distributed with it.

If you're installing LyX on your system, *you should read the README's that came with the LyX distribution and then* Help->LaTeX configuration. Do that first. This chapter does not describe installation or setup of the LyX binary [Well, not everything...]. It does describe all of the things you'll need to use LyX to its fullest.[1]

2.2 Requirements

The version of LyX distributed as binaries only needs X-Windows (X11R6). But it will compile under X11R5 as well. LyX is quite small (about 1M without XForms) and doesn't even need much memory to store the documents. It's also not that fast,[2] but fast enough to allow working on a 386DX40 with 8MB RAM.[3]

If you want to print and preview, you also need a complete LaTeX 2_ε installation, including `dvips` and `xdvi`. LyX will definitely not work with LaTeX2.09 (the changes you would have to do in the LyX-sources are substantially more work than upgrading your LaTeX).

[1] This is basically where we decided to document a bunch of info about running LyX, including what other programs you'll need to make LyX useful.

[2] It might take while if text in a new font has to be displayed, since the font has to be loaded into the X-server. LyX will not load fonts until they are really needed, so not all fonts are immediately accessible in all possible styles and sizes.

[3] Note from Matthias: "I know people who do this."

If you want to compile your own binary you also need the XForms-library. As of version 1.0 you will need to use one of versions 0.81, 0.86 or 0.88 of the XForms library. We recommend using version 0.88, since that will give you a few extra features. You can obtain it from:

- `http://bragg.phys.uwm.edu/xforms/`
- `ftp://ftp/cs.ruu.nl/pub/XFORMS`

You also need the pixmap library (libXpm). Since this is usually a part of most modern Unixes it shouldn't be a problem. Just make sure you have the right pixmap library for your version of the XForms library. There are some incompatibility issues. As of version 1.0 you will need version 4.7 or 4.8 of libXpm.

To compile LyX... well, you wouldn't be reading this document if you hadn't already done so, would you?[4]

2.3 Basic LyX Setup

Again, you probably wouldn't be reading this file if you didn't already have a working LyX binary, so we won't discuss that here. There are a few things we do want to mention about LyX setup and installation.

There are two ways to run LyX. The first way is to install LyX and all of its support files on your system. Of course, you need root privileges to do that. The second way to run LyX doesn't require root access, letting you "install" LyX somewhere in your own account. LyX will automatically detect where it is as long as the supporting directories are put in the correct places.

There are several features of LyX that can be configured from inside LyX, without resorting to configuration files. First, LyX is able to inspect your system to see what programs, LaTeX document classes and LaTeX packages are available. It uses this knowledge to give reasonable defaults to several variables of `lyxrc`. Although this configuration has already been done when LyX was installed on you system, you might have some items that you installed locally and which are not seen by LyX. To force LyX to re-inspect your system, you should use Options->Reconfigure. You should then restart LyX to ensure that the changes are taken into account. As far as LaTeX classes and packages are concerned, you will find information about what has been found under Help->LaTeX Configuration.

The second set of settings that you might want to change comprises all the document-level setting that you can change via the Document, Paper, Quotes and LaTeX Preamble popups of the Layout menu[5]. To do this, open a (possibly

[4] If, by some odd chance, someone has handed you a printed version of this manual and the source files, please feel free to go ask *them* for help. Or, look at the `README` files that came in the source code distribution.

[5] LyX 0.10.x users might remember that these settings used to be accessible in `lyxrc`. This is not the case anymore.

new) scrap document, set all these options according to your taste and save them with Layout->Save layout as default. This will create a template named `default.lyx` which is automatically loaded by LyX when you open a document without template such that the settings are automatically set up as you defined them.

There are many other user-configurable options that you can feed to LyX. Upon startup, LyX attempts to read a file called `~/.lyx/lyxrc` beneath your home directory. If this file does not exist, LyX will read a global options file called `lyxrc`. The document *Customization* contains more information about these setup files.

2.4 Setting Up the X Keyboard

To use LyX properly, you *must* set X up correctly. This is especially vital if you're using the international support features of LyX and want to use non-English keyboard mappings. Unfortunately, almost nobody bothers to do this, especially those who've installed Linux on a PC. Administrators of large systems can be guilty of this, too, so don't assume that you're safe if you're using a large system. Any user can instruct X how to use his or her keyboard. You needn't rely on your sysadmin for this - in fact, you *shouldn't*! The following two programs are all you need to set up your keyboard the way you want it.

2.4.1 `xmodmap` and `xkeycaps`

First of all read the man pages for these two programs. They are your best friends when you are trying to set up X key mapping correctly. If you don't have them, install them.

2.4.1.1 `xmodmap`

This document contains no information on how to use `xmodmap`. There is a sample `.Xmodmap` file in *Customization.* To load the new X keyboard mappings, place the command `xmodmap .Xmodmap` somewhere in your startup scripts [for example, `.cshrc`, `.profile`, `.login`, or `.xinitrc` are all possibilities].

2.4.1.2 `xkeycaps`

This program is a dream come true! It brings up a graphical version of your keyboard, allows you to make modifications, and then spits those modifications out to the standard output in a form readable by `xmodmap`. It is very useful when you're trying to design a new `.Xmodmap` file, though it will require you to do a bit of cut-and-pasting.

2.4.2 Modifiers and Mode_switch

LyX supports three modifiers: Shift [S-], Control [C-], and Meta [M-]. Moreover, if one of the keys of your keyboard is configured as a Compose key, then you can use it to enter some characters not available on your keyboard. This compose key can be used either as a modifier (like Shift or Control) or as a prefix key. Here are some examples of what you can do with a Compose key:

- Compose+e+' → é
- Compose+O+R → ®
- Compose+1+2 → ½
- Compose+<+< → «

This input method is particularly handy when you use accented characters only from time to time. It works by default for latin1 characters, but other input methods will be used if you setup your locale correctly.

2.4.3 Helpful Hints and Tips

First, open up two xterminals. Use one to edit a new `.Xmodmap` file and run `xkeycaps` from the other. Using `xkeycaps`, remap your keyboard the way you want it. There's a button in `xkeycaps` to output the new keymap. Once you hit it, `xkeycaps` will spit a bunch of stuff on the xterm you executed it from. Just copy and paste all of that into your `.Xmodmap` file, and you're done.[6]

Also, there are some things you can do to help you get oriented. Try executing the command `xmodmap -v -pm`. This will show you all of the currently active modifiers. Also try `xmodmap -v -pke | more` to see which keycode numbers are mapped to which symbolic names. It will also give you some idea of the syntax of the `.Xmodmap` file.

There's one thing you'll need to check. Make sure that your Delete and BackSpace keys are *not* defined as the same key symbol by X! Note that giving these two keys unique symbol names will not necessarily alter the behavior of your programs. Some programs bind Delete and BackSpace to the same operation. Emacs is one. Other programs, however, use Delete and BackSpace for different operations. LyX is one of these programs, and if you have Delete and BackSpace labeled with the same key symbol name, you'll have trouble using LyX.

[6] You could also save yourself some typing by executing `xkeycaps > .Xmodmap`. This will create a usable map file. Of course, if you hit the "output keymap" button in `xkeycaps` more than once, the resulting map file will be a mess. As with all things, `xkeycaps` is a tool, and only as intelligent as the person on the other end.

2.5 LATEX

If you want to do more with LYX than simply create documents and spit out `.tex` files, you'll need LATEX.

In case you were wondering, LATEX is an extension to TEX, a document preparation system invented in 1984 by Donald Knuth.[7] TEX takes a set of commands in an ASCII file and converts it to a "device-independent" format, or Dvi, for short. The Dvi file can then be sent to printers. TEX is programmable, and LATEX is nothing but a [really huge] set of TEX macros. LATEX will typically come as part of a TEX distribution, so all you need is a TEX package.

There are two version of LATEX commonly in use today. One is the older LATEX 2.09; the other is the newer LATEX 2_ε. LYX will only work with LATEX 2_ε and will gag on the older version 2.09.

Let us make that perfectly clear:

LYX *WILL ONLY WORK WITH* LATEX 2_ε!!!

If your system has version 2.09, get the sysadmin to upgrade.

If you're using Linux, LATEX 2_ε should have come with your distribution. Look in the instructions that came with your Linux distribution [most likely a CD-ROM] and install the TEX package, if you didn't do so already. Everything you need will be there, along with some things you probably don't need.

For other systems, you'll probably have to talk to your sysadmin to get TEX/LATEX installed on your machine if it's not on there already.

Oh, did we mention that LYX only works with LATEX 2_ε?

2.6 Dvips and Ghostscript

2.6.1 What You Need

There's one more step you need to take if you want to print your LYX documents. Obviously, you'll need to make sure your printer is configured [see next section]. You'll also need to install these programs (or compatibles), if you don't have them already:

- `dvips`
- `ghostscript`
- `xdvi`
- `ghostview`

[7] A note about pronunciation: TEX originated from the Greek letters, $\tau\epsilon\chi$, which rhymes with "blech." That's how you pronounce "TEX" and "LATEX." [If you're American, just pronounce the "X" as a "k" and you've got it.]

The latter two programs are previewer for files in Dvi and PostScript®[8] format. If you don't know what a dvi-file is, you've probably also never worked with LATEX and should read the *Tutorial* document before proceeding further. `dvips` converts dvi-files into PostScript®, which is the format most printers use nowadays. For those of you using dot-matrix and inkjet printers, you'll want to filter the PostScript® through `ghostscript`, which is capable of creating output for a variety of printers. The following section on printer setup describes how to do this automatically every time you print. For now, we'll concentrate on `dvips`.

2.6.2 Dvips

Whether you'll be running LYX on a large system or a Linux box at home, you should configure `dvips`. `dvips` will either "print" into a file, or send output directly to the printer, depending on how it's configured. If it is set up to print to a file, and if no filename is specified, it will simply turn `foo.dvi` into `foo.ps`. Most systems have `dvips` set up to send output to the default printer. For LYX, you'll want the flexibility to do both.

If you are not a mood to configure `dvips` to adapt its output to your printer, you can safely skip this section. Be warned however that the output will not match the quality that you could expect from your printer. At least, it will print.

If you are using teTEX (a TEX distribution which is particularly popular on Linux), you should run the program `texconfig`. To make the name of a new printer recognized by `dvips` you should then select menu entry Dvips, then add. Enter the required parameters and, before exiting, remember to select the function Rehash.

Let's turn now to manual configuration: in order to inform `dvips` how to automagically convert a `.dvi` file into a `.ps` file adapted to printer `foo`, you need to have a config-file, "`config.foo`," lying around somewhere. Typically, the `config.*` files for `dvips` will be in `/usr/lib/texmf/dvips` in most TEX distributions. Your system will probably be different, of course, so just look under the main TEX directory for a subdirectory called "`dvips`." It'll be there somewhere.

Typically, there will be at least one config-file: `config.ps`. This file is the default configuration file, which is *always* read by dvips[9]. Read this file and see what options could need to be changed for your particular printer. Then create a file `config.foo` containing only the relevant lines.

There's at least one thing you need to do to the config-file. There may exist a line that looks like, "`o | lpr`" [without the quotes, of course...]. Change it to "`o | lpr -Pfoo`", so that the output is sent by default to printer `foo`. However, you should probably investigate the entries "`M`" and "`D`", which define respectively the Metafont mode and the resolution of the printer. If you do not

[8] PostScript® is a registered trademark of Adobe Systems Incorporated, and is the main page description language in the UN*X world.

[9] In particular, this file is not necessarily connected to the existence of a file named `ps`.

know what a Metafont mode is, you can see it as a printer driver: it adapts the design of TEX fonts to ensure that they give the best possible result on your printer. Be warned however that, if you define different Metafont modes for different printers, `dvips` will generate several copies of your TEX fonts on disk, and these take valuable space.

Once you are satisfied that your printers are correctly configured, you should tell LyX to make use of this configuration. To do this, you should edit `lyxrc` and set correctly the entries `\print_adapt_output` and `\print_spool_command` as described there.

You can use as many config-files as you like, one for each of your printers. The default printer for LyX can be specified in `lyxrc` or with the `PRINTER` environment variable. You can also choose the desired printer from inside LyX, as described in a later section.Once you've done all that, you can print to either a PostScript® printer or file from LyX.

If your printer doesn't understand PostScript®, you'll need to use `ghostscript` as a filter for your print spooler. That's covered in numerous HOWTO's and manuals. We also have a section that covers a little bit of this.

Some people don't seem to like using the `dvips` plus `ghostscript` combination. As alternative, you can use a program that converts the dvi-file directly into your printer language. You can specify this program in `lyxrc`, too. There is a major disadvantage to this method. You can't include any PostScript® files, such as graphics, in your documents, since the printer-specific conversion programs don't understand PostScript®. For that reason, the LyX team highly recommends using `dvips` and `ghostscript` for printing.

2.6.3 Ghostscript, Xdvi and Ghostview

`Xdvi` and `ghostview` are viewers, The former handles `.dvi` files, while the later interfaces with `ghostscript` to allow you to view PostScript® files.

A quick note on both of these programs. Both automatically update themselves if the viewed file[10] changes. You can also force an update. So, once you've opened one of these two viewers, there's no reason to close it. Also, both programs are functionally the same, providing all of the same features.

The LyX team recommends using `xdvi` for fine tuning documents. Why? It's faster; there's one less layer of processing you need to do before you can view the changes. Here's an example:

1. Use `xdvi` to preview a document from LyX, and leave it running.
2. Make changes to the document using LyX.
3. To view those changes, just choose Update DVI from the File menu. When LaTEX's all done, click on the `xdvi` window, and voilà! `xdvi` will update itself.

[10] That means the `.dvi` or `.ps` file, not the files used to make these.

Now, this doesn't mean `ghostview` is useless. `ghostview` is better suited to those occasions where you *must* view the PostScript® version of the document. For repeated changes that aren't PostScript® dependent, you're better off previewing with `xdvi`. There is an alternative to `ghostview` which sports a much better interface: `gv`. LyX will automatically use it instead of ghostscript if it is available.

2.7 The Printer

Anyone working on a large system shouldn't have any problems here. Your sysadmin [or you, if you are the sysadmin] should already have the printers set up for your system. All you need to do is find out the name of the printer you want to use, and configure your `lyxrc` as described in the last section.

Those of you using Linux, however, will have a bit more work to do. Many people now receive a Linux distribution, such as Red Hat or Slackware, on CD-ROM. They follow the install instructions, get Linux up and running, but never realize that they need to set up their printer. If you've never touched the file `/etc/printcap`, you need to set up your printer. We've written a little something to help you out with that; check out the "*A Printer Tutorial*" chapter in the *Customization* manual for help.

Chapter 3

LyX Basics

3.1 Document Types

3.1.1 Introduction

Before you do anything else, before you ever start writing a document, you need to decide what *type* of document you want to edit. Different types of documents use different types of spacing, headings, numbering schemes, and so on. Additionally, different documents use different paragraph environments, and format the title of your document differently.

A *document class* describes a group of properties common to a particular set of documents. By setting the document class, you automagically select these properties, making it easier to create the type of document you want. If you don't choose a document class, LyX picks one for you by default. So, it behooves you to change the class of your document.

Read on for info about the document classes you can choose from LyX, and how to fine-tune some of their properties.

3.1.2 The Various Document Classes

3.1.2.1 Overview

There are five standard document classes in LyX. They are:

Article for basic articles

Report for basic reports

Book for writing a book

Letter for US-style letters

Slides is used to make transparencies

There are also some non-standard classes, which LyX only uses if you have a LaTeX setup that supports them:

Aapaper Journal articles in the style and format used in Astronomy & Astrophysics

Amsart Journal articles in the style and format used by the AMS [American Mathematical Society]. There are three amsart layouts available. The standard one uses a typical numbering scheme for theorems, *etc.*, that prepends the section number to the number of the result. All result-type statements (propositions, corollaries, and so on) are sequenced together, but definitions, examples, and the like have their own sequence. The "sequential numbering" scheme does not place the section number with each result, but numbers them throughout the article in a single sequence. Each type of result gets its own sequence. There is also a layout that dispenses with numbering of statements altogether.

Amsbook Books in the style and format used by the AMS. Only the standard numbering scheme is provided, under the assumption that you would not want to number results consecutively throughout a book, and that you would need to number results.

Dinbrief für Briefe nach deutscher Art

Foils is used to make transparencies, but is better than slides

Iletter per lettere all' italiana

Linuxdoc Used with the SGML-tools package (formerly known as LinuxDoc). It allows LyX to produce SGML output. SGML is a markup language and is the predecessor to HTML. The SGML-tools package allows you to convert SGML to HTML or to the format used by `man` pages.

Paper for use with the `paper` LaTeX document class [not in all LaTeX distributions]

Revtex is used to write articles for the publications of the American Physical Society (APS), American Institute of Physics (AIP), and Optical Society of America (OSA). This class is not completely compatible with all LyX features.

We won't go into any detail about how to use these different document classes here. You can find all the details about the non-standard classes in the *Extended Editing* manual. Here, we will settle with a list of some of the common properties of all of the document classes.

3.1.2.2 Selecting a Class

You can select a class using the Document Layout popup. To open the Document Layout popup, select Document... from the Layout menu. Then use Class to select the class you want to use.

After you select a class, LyX then asks if you want to use the defaults for Pagestyle, Sides, Columns, Separation, and Extra Options. It's safe to hit Yes here. If you look at the Document Layout popup again, you'll notice options for all of these. So, you can always fine-tune the defaults for a given section.

3.1.2.3 Properties

In case you're wondering what those "defaults" are for each of the document classes, here's a quick table describing them:

	Pagestyle	Sides	Columns	Max. sectioning level
article	Plain	One	One	Section
report	Plain	One	One	Chapter
book	Headings	Two	One	Chapter
letter	Plain	One	One	none
linuxdoc	Plain	One	One	Section
aapaper	Plain	Two	Two	Section
amsart	Headings	One	One	Section
dinbrief	Plain	One	One	none
iletter	Plain	One	One	none
paper	Headings	One	One	Section

There is no default value of Extra Options or Separation for any of these classes.

You're probably also wondering what "Max. sectioning level" means. There are several paragraph environment used to create section headings. Different document classes allow different types of section headings. Only two use the Chapter heading; the rest do not and begin instead with the Section heading. Some document classes, such as the three for letters, don't use any section headings. In addition to Chapter and Section headings, there are also Subsection headings, Subsubsection headings, and so on. We'll describe these headings fully in section 3.3.4.

3.1.3 Fine-Tuning the Defaults

Okay, we know we never told you what most of these "default options" set by the Class button do. That's what this section is for.

Pagestyle
: This is another list, containing five options. It controls what sorts of headings and page numbers go on a page:[1]

Default	Use default pagestyle of current class.
Empty	No page numbers or headings.
Plain	Page numbers only.
Headings	Page numbers and either the current chapter or section title and number. Whether LYX uses the current chapter or the current section depends on which is the maximum sectioning level.
Fancy	This allows you to create fully customizable headers and footers if you have the fancyhdr package installed. At the moment, support in LYX is limited to this setting. To use the full power of this package, you have to resolve to magic codes in your preamble. Check the documentation for the fancyhdr package for more details.

Sides
: No, LYX can't make your printer print on both sided of a sheet of paper! However, it can use a different format for odd-numbered pages than even-numbered pages. This way, if you *do* have a printer that duplexes,[2]your page number will always be in the upper right corner of the page and the left margin will have extra room for a binding.

There are two toggle buttons here: One for single-sided documents, Two for double-sided documents.

Columns
: Yes, this does control how many columns each page has. You can choose, using the toggle buttons, One or Two for the number of columns.

Note that LYX won't show two columns on screen. That's impractical, often unreadable, and not part of the WYSIWYM concept. However, there *will* be two columns in the generated output.

Extra Options
: The LATEX command `\documentclass` takes several options. LYX sets some of these automatically for you. This text box allows you to enter in others. Just type in a comma-separated list of options. See a good LATEX book to find out what kinds of additional options you can use.

Separation
: This has its own section. See sec. 3.2.1 for a description of what this does.

[1] LATEX does this part.
[2] i.e. prints on both sides of a sheet of paper

3.1.4 Paper Size, Orientation, and Margins

There are several other options to set in the Document Layout popup. All of them are global options, but they have special purposes and only affect certain features. We describe what these options do in the same section that describes the features they affect.

There are two options that affect the overall layout of the document, so we'll describe them here. You'll find them in the Paper popup under the Layout menu:

Orientation Two toggle buttons choose whether to print the output as Landscape or as Portrait.

Papersize What size paper to print on. The choices are

- Default
- A3, A4, A5
- B3, B4, B5
- US Letter
- US legal
- US executive
- Custom

Some of these settings require you to have the geometry package installed. This package will also allow you to set the margins in the Paper popup.

3.1.5 Important Note:

If you change a document's class, LyX has to convert *everything* into the new class. That includes the paragraph environments. Some paragraph environments are standard; all of the document classes have them. Some classes have special paragraph environments, however. If this is the case, and you change document classes, LyX sets the missing paragraph environments to Standard and places an error box at the beginning of the paragraph. Just click on them and you'll get a message popup that tells you about the conversion and why it failed.

3.2 Paragraph Indentation and Separation

3.2.1 Introduction

Before describing all of the various paragraph environments, we'd like to say a word or two about paragraph indentation.

Everyone seems to have their own convention for separating paragraphs. Most Americans indent the first line of a paragraph. Others don't indent but put extra space between the paragraphs. LyX uses the same convention you find among typographer. The *first* paragraph of a section, or after a figure, an equation, a table, a list, etc., is *not* indented. Only a paragraph following another paragraph gets indented. Some people don't like this convention, but if you want to use indented paragraphs, you'll have to live with it.[3]

The space between paragraphs, like the line spacing, the space between headings and text — in fact, all of the spacings for just about everything are pre-coded into LyX. As we said, you don't worry about how much space to add between what. LyX takes care of that. In fact, these pre-coded vertical spacings aren't a single number but a range. That way, LyX can squish or stretch the space between lines to make sure figures fit on a page with text, so that sections don't start at the bottom of a page, and so on.[4] However, pre-coded doesn't mean you can't change them. LyX gives you the ability to globally change *all* of these pre-coded spacings. We'll explain more later.

3.2.2 Global Indentation Method

To select the default method of separating paragraphs, select Document... from the Layout menu. This opens the Document Layout popup. Select Indent or Skip to indent paragraphs or add extra space between paragraphs, respectively.

3.2.3 Fine-Tuning

You can also change the separation method of a single paragraph. Choose Paragraph... from the Layout menu to open the Paragraph Layout popup. Toggle the No Indent button to change the state of the current paragraph. If paragraphs indent by default, this button will be inactive at first. If paragraphs have no indentation but use extra space for separation, this button will be depressed initially.

You should only need to change the indentation method for a single paragraph if you need to do some fine-tuning. Typically, you'll select Indent or Skip for the entire document and edit away.

3.2.4 Changing Line Spacing

In the Document Layout popup you can choose your line spacing provided you have the setspace package installed.

[3] There is a way to force LaTeX to indent all paragraphs. LyX won't show this, of course, but LaTeX *will* print it that way. You'll need to get a special package and insert an appropriate command in the preamble. [*Editor's Note: I'll insert this info when I get the time. - jw*]

[4] Actually, LaTeX does this when LyX goes to produce a printable file.

3.3 Paragraph Environments

3.3.1 Overview

The paragraph environments correspond to the various `\begin`{*environment*} ... `\end`{*environment*} command sequences in an ordinary LaTeX file. If you don't know LaTeX, or the concept of a paragraph environment is totally alien to you, we urge you to read the *Tutorial*. The *Tutorial* also contains many more examples than this section does.

A paragraph environment is, simply, a "container" for a paragraph which gives that paragraph certain properties. This can include a particular style of font, different margins, a numbering scheme, labels, and so on. Additionally, you can "nest" the different environments inside one another, allowing one environment to inherit some of the properties of another. The different paragraph environments totally replace the need for messy tab stops, on the fly margin adjustment, and other holdovers from the days of typewriters. There are several paragraph environments which are specific to a particular document type. We'll only be covering the most common ones here.

To choose a new paragraph environment, use the pull-down box on the left end of the toolbar. [Click on the arrow to open and close it.] LyX will change the environment of the *entire* paragraph in which the cursor sits. You can also change the environment of an entire group of paragraphs if you select them before choosing the new environment.

Note that hitting Return will *typically* create a new paragraph using the Standard paragraph environment. We say "typically" because this isn't always the case.[5] Usually, starting a new paragraph resets both the paragraph environment and the nesting depth [more on nesting in section 3.4]. Sometimes, however, LyX keeps the old environment. At the moment, all this is context-specific; you're better off expecting Return to reset the paragraph environment and depth. If you want a new paragraph to keep the current environment and depth, use M-Return instead.[6]

3.3.2 Standard

The default paragraph environment is Standard. It creates a plain paragraph. If LyX resets the paragraph environment, this is the one it chooses. In fact, the paragraph you're reading right now [and most of the ones in this manual] are in the Standard environment.

You can nest a paragraph using the Standard environment in just about anything else, but you can't really nest anything in a Standard environment.

[5] If you are in one of these environments:

- Quote
- Verse
- Enumerate
- List
- Quotation
- Itemize
- Description

LyX keeps the old paragraph environment when you hit Return, rather than resetting it to Standard. LyX will still reset the nesting depth, however.

[6] This invokes the function `break-paragraph-keep-layout`.

3.3.3 Document Titles

A LATEX title page has three parts: the title itself, the name[s] of the author[s] and a “footnote” for thanks or contact information. For certain types of documents, LATEX places all of this on a separate page along with today’s date. For other types of documents, the title “page” goes at the top of the first page of the document.

LYX provides an interface to the title page commands through the paragraph environments Title, Author, and Date. Here’s how you use them:

- Put the title of your document in the Title environment.
- Put the author name in the Author environment.
- If you want the date to have a certain appearance, want to use a fixed date, or want other text to appear in place of today’s date, put that text in the Date environment. Note that using this environment is optional. If you don’t provide any, LATEX will automatically insert today’s date.

Be sure to do this at the top of the document. You can use footnotes to insert “thanks” or contact information.

3.3.4 Headings

There are nine paragraph environments for producing section headings. LYX takes care of the numbering for you. All you need to do is decide what you’re going to call section 3 of chapter 9.

3.3.4.1 Numbered Headings

There are 6 numbered types of section headings. They are:

1. Chapter
2. Section
3. Subsection
4. Subsubsection
5. Paragraph
6. Subparagraph

LYX labels each heading with a series of numbers, separated by periods. The numbers describe where in the document you are. These headings all subdivide your document into different pieces of text. For example, suppose you’re writing a book. You group the book into chapters. LYX does similar grouping:

- Either Chapter or Section is the maximum sectioning level.

- Chapters are divided into Sections
- Sections are divided into Subsections
- Subsections are divided into Subsubsections
- Subsubsections are divided into Paragraphs
- Paragraphs are divided into Subparagraphs

Note: not all document types use the Chapter heading as the maximum sectioning level. In that case, the Section is the top-level heading.

So, if you use the Subsubsection environment to label a new sub-subsection, LyX labels it with its number, along with the number of the subsection, section, and, if applicable, chapter that it's in. For example: the fifth section of the second chapter of this book has the label "2.5".

3.3.4.2 Unnumbered Headings

There are 3 types of unnumbered section headings. They are:

1. Section*
2. Subsection*
3. Subsubsection*

The "*" after each name means that these headings are not numbered. They work the same as their numbered counterparts.

3.3.4.3 Changing the Numbering

You can also alter which sectioning levels get numbered and which ones appear in the Table of Contents. Now, this doesn't remove any of the levels; that's preset in the document class. Certain classes start with Chapter and go down to the Subparagraph level. Others start at Section. Similarly, not all document classes number all sectioning levels. Most don't number Paragraph or Subparagraph. This is something you can change.

Open the Document Layout popup [via the Document... item in the Layout menu]. You should see a counter labelled Section number depth. This counter controls how far down in the sectioning hierarchy LyX numbers a section heading. Unfortunately, the number you choose with the slider is really goofy, so here's a table of values and what they do:

Sec. Num. Depth value:	LYX numbers these section headings:
-1	no numbering
-0 and 0	Chapter
1	Chapter through Section
2	Chapter through Subsection
3	Chapter through Subsubsection
4	Chapter through Paragraph
5	Chapter through Subparagraph

Of course, if you're using a document class that doesn't use Chapter headings, then the numbering begins at the Section heading, and "0" also corresponds to "no numbering."

There's another counter on the Document Layout popup, called Table of contents depth. It works the same way as Section numbering depth, only it controls which sectioning levels appear in, you guessed it, the Table of Contents. This is a great control to have. Suppose you wanted to number *all* sectioning heading, but you only wanted Chapters, Sections, and Subsections in the Table of Contents. You'd just set Section numbering depth to "5" and Table of contents depth to "2" and voilà! You're all set.

3.3.4.4 Special Information

The following information applies to Chapter, Section, Subsection, Subsubsection, Paragraph, Subparagraph as well as Section*, Subsection*, and Subsubsection*:

- You cannot use a margin note in any of these environments.
- You can use inlined math in these environments.
- You cannot do any nesting with these environments.
- You can use labels and cross-references to refer to their numbers.

As for examples of these paragraph environments - look around you! We're using them everywhere in the manuals.

3.3.4.5 Creating an Appendix

Currently, LYX cannot display an appendix. You can use inserted LATEX code to create an appendix with the command:

```
\appendix
```

All headings following this command will appear in the previewed or printed document as appendices.

See section 7.4 to learn how to insert LATEX commands into a document.

3.3.5 Quotes and Poetry

LyX has three paragraph environments for writing poetry and quotations. They are Quote, Quotation, and Verse. Forget the days of changing linespacing and twiddling with margins. These three paragraph environments already have those changes built-in. They all widen the left margin and add a bit of extra space above and below the text they contain. They also allow nesting, so you can put a Verse in a Quotation, as well as in some other paragraph environments.

There is another feature of these three paragraph environments: they do *not* reset to Standard when you break a paragraph. So, you can type in that poem and merrily hit Return without worrying about the paragraph environment changing on you. Of course, that means that, once you're done typing in that poem, you have to change back to the Standard environment yourself.

3.3.5.1 Quote and Quotation

Now that we've described the similarities of these three environments, it's time for the differences. Quote and Quotation are identical except for one difference: Quote uses extra spacing to separate paragraphs and never indents the first line. Quotation *always* indents the first line of a paragraph and uses the same line spacing throughout.

Here's an example of the Quote environment:

> This is in the Quote environment. I can keep writing, extending this line out further and further until it wraps. See - no indentation!
>
> Here's the second paragraph of this quote. Again, there's no indentation, but there is extra space between me and the other paragraph.

That ends that example. Here's another example, this time in the Quotation environment:

> This is in the Quotation environment. If I keep writing, you'll see the indentation. If your country uses a writing style that shows off new paragraphs by indenting the first line, then Quotation is the environment for you! Well, you'd use it *if* you were quoting other text.
>
> Here's a new paragraph. I could ramble on and on, like a politician at election time. If I did that, though, you'd get bored.

That was our other example. As the example notes, Quote is for those people who use extra space to separate paragraphs. They should put quotes in the Quote environment. those who use indentation to mark a new paragraph should use the Quotation paragraph environment for quoted text.

3.3.5.2 Verse

Verse is a paragraph environment for poetry, rhymes, verses, and so on. Here's an example:

This is in Verse
Which I did not rehearse!

It could be much worse. This line could be long, very long, oh so long, so very long that it wraps around. It looks okay on screen, but in the printed version, the extra lines are indented a bit more than the first. Okay, so it's turned to prose and doesn't rhyme anymore. So sue me.

To break a line
And make things look fine
Use C-Return.

As you can see, Verse does not indent both margins. Each stanza of the verse or poem is in its own paragraph. To separate the individual lines of a stanza, use the `break-line` function, C-Return.

3.3.6 Lists

This is an area where LyX clearly outshines standard word processors. One of the most common complaints and "bug" reports on the LyX Developer's list is, "I can't put in more than one space!" Typically, the user in question is looking for some way to indent text so he can make an outline or a list. After all, on the brand-name commercial word processors, if you want to write a list, you need to apply tab stops, indents, an a plethora of other horizontal spacing commands.

But why should *you* have to do this at all? Tab stops are fine for a typewriter, where you have no other way to indent text. A computer, on the other hand, is quite good at counting and doing things repeatedly. So, you shouldn't need to keep track of list numbers, indentations, and such. And with LyX, you don't.

LyX has four different paragraph environments for creating different kinds of lists. In the Itemize and Enumerate environments, LyX labels your list items with bullets or numbers, respectively. In the Description and List environments, LyX lets you provide your own label. We'll present the individual details of each type of list next after describing some general features of all four of them.

3.3.6.1 General Features

The four paragraph environments for lists differ from the other environments in several ways. First, LyX treats each paragraph as a list item. Hitting Return does *not* reset the environment to Standard but keeps the current environment and creates a new list item. The nesting depth is typically reset, however. If you want to keep both the current nesting depth and paragraph environment, you should use M-Return to break paragraphs.

You can nest lists of any type inside one another. In fact, LyX changes the labels on some list items depending on how its nested. If you intend to use any of the list paragraph environments, we *strongly urge* you to read all of section 3.4.

3.3.6.2 Itemize

The first type of list we'll describe in detail is the Itemize paragraph environment. It has the following properties:

- Each item has a particular bullet or symbol as its label.
  - LyX uses the same symbol for all of the items in a given nesting level.
  - The symbol appears at the beginning of the first line.
- The items can be any length. LyX automatically offsets the left margin of each item. The offset is always relative to whatever environment the Itemize list may be in.
- If you nest an Itemize environment inside another Itemize environment, the label changes to a new symbol.
  - There are four different symbols for up to a fourfold nesting.
  - LyX always shows the same symbol, an asterisk, on screen.
  - See section 3.4 for a full explanation of nesting.

Of course, that explanation was also an example of an Itemize list. The Itemize environment is best suited for lists where the order doesn't matter.

We said that different levels use different symbols as their label. Here's an example of all four possible symbols. Note that those of you reading this manual online won't see any difference.

- The label for the first level Itemize is a large black dot, or bullet.
  - The label for the second level is a dash.
    - The label for the third is an asterisk.
      - The label for the fourth is a centered dot.
    - Back out to the third level.
  - Back to the second level.
- Back to the outermost level.

These are the default labels for an Itemize list. You can customize these labels by adding the correct commands to the LaTeX preamble (see section 7.5.1). Of course, LyX won't know about them; that's the whole point to the WYSIWYM concept. Any customizations you make appear in the printed document.

Notice how the space between items decreases with increasing depth. We'll explain nesting and all the tricks you can do with different depths in section 3.4. Be sure to read it!

3.3.6.3 Enumerate

The Enumerate environment is the tool to use to create numbered lists and outlines. It has these properties:

1. Each item has a numeral as its label.
   (a) The type of numeral depends on the nesting depth.
2. LYX automatically counts the items for you and updates the label as appropriate.
3. Each new Enumerate environment resets the counter to 1.
4. Like the Itemize environment, the Enumerate environment:
   (a) Offsets the items relative to the left margin. Items can be any length.
   (b) Reduces the space between items as the nesting depth increases.
   (c) Uses different types of labels depending on the nesting depth.
   (d) Allows up to a fourfold nesting.

Unlike the Itemize environment, Enumerate *does* show the different labels for each item. Here is how LYX labels the four different levels in an Enumerate:

1. The first level of an Enumerate uses Arabic numerals followed by a period.
   (a) The second level uses lower case letters surrounded by parentheses.
      i. The third level uses lower-case Roman numerals followed by a period.
         A. The fourth level uses capital letters followed by a period.
         B. Again, notice the decrease in inter-item spacing as the nesting depth increases.
      ii. Back to the third level
   (b) Back to the second level.
2. Back to the outermost level.

Once again, you can customize the type of numbering used in the Enumerate environment. It involves adding commands to the LaTeX preamble (section 7.5.1), however. As stated earlier, such customization only shows up in the printed version, not on the LYX screen.

There is more to nesting Enumerate environments than we've stated here. You *really* should read section 3.4 to learn more about nesting.

3.3.6.4 Description

Unlike the previous two environments, the Description list has no fixed label. Instead, LyX uses the first "word" of the first line as the label. Here's an example:

Example: This is an example of the Description environment.

LyX typesets the label in boldface and puts extra space between it and the rest of the line.

Now, you're probably wondering what we mean by, "uses the first 'word'." As stated earlier,[7] the Space key does not add a whitespace character, but separates words from one another. Inside of a Description environment, the Space key tells LyX to end the label if we're at the beginning of the first line of an item. [*Editor's Note: Hopefully, someone will change this. It would be nice if, say, the Tab key terminated the label, since LaTeX allows you to use a resizeable space in the label of a description list. -jw*]

However, what if you want or need to use more than one word in the label of a Description environment? Simple: use a Protected Blank. [Use either C-Space or Special Characters->Protected Blank from the Insert menu. See sec. 6.2.1 for more info.] Here's an example:

Second Example: This one shows how to use a Protected Blank in the label of a Description list item.

Usage: You should use the Description environment for things like definitions and theorems. Use it when you need to make one word in particular stand out in the text that describes it. It's not a good idea to use a Description environment when you have an entire sentence that you want to describe. You're better off using Itemize or Enumerate and nesting several Standard paragraphs into them.

Nesting: You can, of course, nest Description environments inside one another, nest them in other types of lists, and so on. Read section 3.4 to learn about nesting.

Notice that after the first line, LyX indents subsequent lines, offsetting them from the first line.

3.3.6.5 The LyX List

The List environment is a LyX extension to LaTeX.

Now, if you jumped here without reading sections 3.3.6.2-3.3.6.4, you've goofed. The List environment does *not* create numbered lists. That's what Enumerate does, and it's documented in section 3.3.6.3.

Like the Description environment the List environment has user-defined labels for each list item. There are some key differences between this list environment and the other three:

[7] *Editor's Note: This should be stated in the beginning somewhere... -jw*

item labels LYX uses the first "word" of each line as the item label. The first Space after the beginning of the first line of an item marks the end of the label. If you need to use more than one word in an item label, use a Protected Blank [C-Space] between them.

margins As you can see, LYX uses different margins for the item label and the body of the item text. The body of the text has a larger left margin, which is equal to the default label width plus a little extra space.

label width LYX uses one of two things for the label width: the actual width of the label, or the default width, whichever is larger. If the actual width is larger, then the label "extends" into the first line. In other words, the text of the first line isn't aligned with the left margin of the rest of the item text.

default width You can very easily set this default width. It's quite painless, actually. So, you can easily ensure that the text of all items in a List environment have the same left margin.

uses You should use the List environment the same way you'd use as Description list: when you need one word to stand out from the text that describes it. The List environment gives you another way to do this, using a different overall layout.

nesting You can nest List environments inside one another, nest them in other types of lists, and so on. They work just like the other list paragraph environments. Read section 3.4 to learn about nesting.

As you can see, this is a feature-packed paragraph environment!

To change the default width of the label, select the items in the list to change. You can also simply move the cursor into a List item if you want to change only its label width. Now open the Paragraph Layout popup [use Paragraph... in the Layout menu] and find the Label Width text box. The text in the Label Width box determines the default label width. If you really, really want to, you can use the text of your largest label here, but you don't need to. We recommend using the letter "M" multiple times. It's the widest character and is a standard unit of width in LATEX.[8] The default label width in the example List is 6 "M"s wide. Using "M" as your unit of width in the Label Width box has one more advantage: you don't need to keep changing the contents of Label Width every time you alter a label in a List environment.

There's yet another feature of the List environment we need to tell you about. As you can see in the examples, LYX left-justifies the item labels by default. In reality, there is an HFill built into the end of the label. We'll document HFills

[8]Of course, you could use some other character multiple times, too. Or, you could get fancy and use combinations of "|"s, "0"s and "M"s. However, the width of the letter "M" *is* a standard unit of horizontal space under LATEX. That's why we recommend using "M" multiple times.

later in section 6.2.1. You can use additional HFills to change how LyX justifies the item label. Here are some examples:

Left — The default for List item labels.

Center — One HFill at the beginning of the label centers it.

Right #1 — You can sort-of right justify the label by using several Protected Blanks at the beginning of the line. As you can see, however, it alters the margin of the first line. It'll be a pain in the butt to figure out just how many Protected Blanks are enough.

Right #2 — This is somewhat better, though still a kludge. Uses several HFills at the beginning of the label. [*Editor's Note: This is another one of those things that seemed okay at the time, but now look really clunky. The developers should change it. - jw*]

Don't worry if you have no idea what HFills are yet. Just remember that you can use them to customize the look of the List environment.

That does it for the four paragraph environments for making lists. Oh - did we mention that you should read about nesting environments in section 3.4 if you want to use any of these list environments?

3.3.7 Letters

3.3.7.1 Address and Right Address: An Overview

Although LyX has document classes for letters, we've also created two paragraph environments called Address and Right Address.[9] To use the letter class, you need to use specific paragraph environments in a specific order, otherwise LaTeX gags on the document. In contrast, you can use the Address and Right Address paragraph environments anywhere with no problem. You can even nest them inside other environments, though you can't nest anything in them.

Of course, you're not limited to using Address and Right Address for letters only. Right Address, in particular, is useful for creating article titles like those used in European academic papers.

3.3.7.2 Usage

The Address environment formats text in the style of an address, which is also used for the opening and signature in some countries. Similarly, the Right Address environment formats text in the style of a right-justified address, which is used for the sender's address and today's date in some countries. Here's an example of each:

[9] Note from MATTHIAS: I'm a bit proud about this right address layout. Try to reach the same typesetting effect with usual word processors. You will have to create a frame and position it manually with the mouse and all those funny things.

Right Address
WhoAmI
WhereAmI
When is it? What is today?

That was Right Address. Notice that the lines all have the same left margin, which LyX sets to fit the largest block of text on a single line. Here's and example of the Address environment:
WhoAreYou
Where do I send this
Your post office and country

As you can see, both Address and Right Address add extra space between themselves and the next paragraph. Speaking of which, if you hit Return in either of these environments, LyX resets the nesting depth and sets the environment to Standard. This makes sense, however, since Return is the `break-paragraph` function, and the individual lines of an address are not paragraphs. Thus, you'd use `break-line` [C-Return or Special Character->Linebreak from the Insert menu] to start a new line in an Address or Right Address environment.

3.3.8 Academic Writing

Most academic writing begins with an abstract and ends with a bibliography or list of references. LyX contains paragraph environments for both of these.

3.3.8.1 Abstract

The Abstract environment is used for the abstract of an article. Technically, you *could* use this environment anywhere, but you really *should* only use it at the beginning of the document, after the title. Also, don't bother trying to nest Abstract in anything else or vice versa. It won't work.

The Abstract environment does several things for you. First, it puts the centered label "Abstract" above the text. The label and the text of the abstract are separated by some extra vertical space. Second, it typesets everything in a smaller font, just as you'd expect. Lastly, it adds a bit of extra vertical space between the abstract and the subsequent text. Well, that's how it will appear on the LyX screen. If your document is in the "report" class, the abstract actually appears on a separate page in the printed version of the file.

Starting a new paragraph by hitting Return does *not* reset the paragraph environment. The new paragraph will still be in the Abstract environment. So, you will have to change the paragraph environment yourself when you finish entering the abstract of your document.

We'd love to give you an example of the Abstract environment, but we can't, since this document is in the "book" class. If you've never heard of an "abstract" before, you can safely ignore this environment.

One last note: the Abstract environment is only useful in the "article" and "report" document classes [as well as "amsart," which is just a specialized version of "article"]. The "book" document class ignores the Abstract completely, and it's utterly silly to use Abstract in the "letter" document class.

3.3.8.2 Bibliography

The Bibliography environment is used to list references. Technically, you *could* use this environment anywhere, but you really *should* only use it at the end of the document. Also, don't bother trying to nest Bibliography in anything else or vice versa. It won't work.

When your first open a Bibliography environment, LYX add a large vertical space, followed by the heading "Bibliography" or "References," depending on the document class. The heading is in a large boldface font. Each paragraph of the Bibliography environment is a bibliography entry. Thus, hitting Return does *not* reset the paragraph environment. Each new paragraph is still in the Bibliography environment.

At the *beginning* of the *first line* of each paragraph, you will see a grayish box showing a number. If you click on it, you will get a popup in which you can set a key and a label. The key is the symbolic name by which you will refer to this bibliography entry. For example, suppose your first entry in the bibliography was a book about LaTeX. We could choose the key "latexguide" for that entry. You can also give a label, which will be displayed in the gray inset box.

The key field isn't useless. You can refer to your bibliography entries using the Insert->Citation Reference command. Just choose the key inside in the combox key. An example of the Bibliography appears [where else] at the end of this document. "See [4] or [3, Chapter 3]" is an example of how to cite two of the entries in it. In the second one, we used the remark field of the citation popup to add the text "Chapter 3". The texts "latexguide" and "latexcompanion" that you see on screen will be replaced in print by the number or the label of the bibliography entry.

The more advanced LaTeX bibliography package BibTeX is also supported by LYX. For a description of how to use it, please refer to the *Extended LYX Features* document.

3.3.9 Special Purpose

There are three standard paragraph environments that simply don't fit any category, as they are very specialized for a particular purpose. We'll point out the highlights and uses of each.

3.3.9.1 Caption

The Caption environment is the default paragraph environment for Figure Floats and Table Floats. On the LYX screen, you'll see either the label "Figure:" or

"Table:", depending on which type of Float it's in. Additionally, the Caption environment generates a figure or table number in the final output.

You can't really nest things into a Caption environment or vice versa. Additionally, hitting Return resets the paragraph environment to Standard, so a Caption can only be a single paragraph.

You cannot use a Caption environment outside of a Figure Float or a Table Float. See sections 4.3 and 4.4 for more information on Figure Floats and Table Floats.

3.3.9.2 LyX-Code

The LyX-Code environment is another LyX-extension. It typesets text in a typewriter-style font. It also treats the Space key as a fixed whitespace;[10] this is the only case in which you can type multiple whitespaces in LyX. If you need to insert blank lines, you'll still need to use C-Return [the `break-line` function]. Return breaks paragraphs. Note, however, that Return does *not* reset the paragraph environment. So, when you finish using the LyX-Code environment, you'll need to change the paragraph environment yourself. Also, you *can* nest the LyX-Code environment inside of others.

There are a few quirks with this environment:

- You cannot use C-Return at the beginning of a new paragraph [i.e. you can't follow Return with a C-Return].
- You can't follow a C-Return with a Space.
  - Use a Return to begin a new paragraph, then you can use a Space.
  - Or: use C-Space instead.
- You can't have an empty paragraph or an empty line. You must put at least one Space in any line you want blank. Otherwise, LaTeX generates errors.
- You cannot get the typewriter double quotes by typing " since that will insert *real* quotes. You get the typewriter double quotes with C-" or C-q.

Here's an example:

```
#include<stdio.h>

int main(void)
{
    printf(``Hello World\n'');
    return 0;
}
```

[10]In the LyX-Code environment, the Space key is treated as a Protected Blank instead of an end-of-word marker.

This is just the standard "Hello world!" program.

LyX-Code has one purpose: to typeset code, such as program source, shell scripts, rc-files, and so on. Use it only in those very, very special cases where you need to generate text as if you used a typewriter.

3.3.9.3 Comment

The comment paragraph environment can be used to write comments about your document that will not appear in the final output. In order to make this clear, LyX will present the comment environment in a different color than the rest of the text.

3.3.9.4 LaTeX

The LaTeX paragraph environment has a specific purpose: to pass large blocks of commands to LaTeX. It has the following features:

- Text appears in red, indicating that it is LaTeX code.
- You can insert whitespace multiple times using the Space key [i.e. Space gets treated as a Protected Blank].
  - This does not work, however, after a C-Return. Use C-Space in that case.
- If you wish to break up your LaTeX commands over several lines, use C-Return to break the lines.
- The Return key resets the paragraph environment to Standard.
- You *can* nest a LaTeX environment in other environments.

See section 7.4 for more information.

3.4 Nesting Environments

3.4.1 The Big Deal

Throughout the previous sections, we've been constantly nagging you to "go read section 3.4." So, you're probably wondering what the big deal is.

The big deal is that LyX differs rather strongly from the traditional "word-processor-as-overglorified-typewriter" concept. With a typewriter, text is merely ink on a page. Most word processors aren't much better, treating text as pixels on the screen and bytes in memory. In contrast, LyX treats text as a unified block with a particular context and specific properties. However, what if you wanted one "block" to inherit some of the properties of another "block?"

Here's a more specific example: outlines. You have three main points in your outline, but point #2 also has two subpoints. In other words, you have a list *inside* of another list, with the inner list "attached" to item #2:

1. one
2. two
    (a) sublist - item #1
    (b) sublist - item #2
3. three

How do you put a list inside of a list?

By now, the answer should be obvious: you nest one list inside the other. However, nesting isn't just limited to lists. In LYX, you can nest just about anything inside anything else, as you're about to find out. This is the real power of nesting paragraph environments.

3.4.2 What You Can and Can't Nest

Before we fire a list of paragraph environments at you, we need to tell you a little bit more about how nesting works.

The question of nesting is a bit more complicated than a simple yes or no, can you or can't you. There's also the question of how. Can you nest this environment into anything else? Can you nest another environment into it? A "yes" to one of these doesn't guarantee a "yes" to the other.

The paragraph environments in LYX can do one of three things when it comes to nesting. First, an environment may be completely unnestable. Second, there are environments that are fully nestable. You can nest them inside of things and you can also nest other things inside of them. There is one last type of environment. You can nest them into other environments, but that's it. You can't nest anything into them.

Here's a list of the three types of nesting behavior, and which paragraph environments have them:[11]

Unnestable Can't nest them. Can't nest into them.

- Chapter
- Section
- Subsection
- Subsubsection
- Paragraph
- Subparagraph

[11] For some odd reason, LYX allows you to fully nest both Bibliography and Abstract. Also, LYX allows you to nest Title, Author, and Date into other environments. We urge you not to. LATEX may barf if you try it. Then again, it may not. We don't know for certain. However, it makes no sense contextually to perform any nesting with these environments, so why would you ever want to?

- Section*
- Subsection*
- Subsubsection*
- Bibliography
- Abstract
- Title
- Author
- Date

Fully Nestable You can nest them. You can nest other things into them.

- Verse
- Quote
- Quotation
- Itemize
- Enumerate
- Description
- List
- LyX-Code

Nestable-Inside You can nest them inside of other things. You can't nest anything into them.

- Standard
- Right Address
- Address
- LaTeX

The Caption environment doesn't really fit anywhere, since it's only used inside of Table Floats and Figure Floats.

3.4.3 Nesting Other Things: Tables, Math, Floats, etc.

There are several things that aren't paragraph environments, but which are affected by nesting anyhow. They are:

- Equations
- Tables
- Figures

[Note: if you put a figure or a table in a Float, this is no longer true. See below or look in sections 4.3 or 4.4 for more info.]

LYX can treat these three objects as either a word or as a paragraph. Well, you can't inline a table, but you can inline math and figures. If a figure or an equation is inlined, it goes wherever the paragraph it's in goes.

On the other hand, if you have an equation, figure or table in a "paragraph" of its own, it behaves just like a "nestable-inside" paragraph environment. You can nest it into any environment, but you [obviously] can't nest anything into it.

Here's an example with a table:

1. Item One

   (a) This is (a) and it's nested.

a	b
c	d

   (b) This is (b). The table is actually nested inside (a).

2. Back out again.

If we hadn't nested the table at all, the list would look like this:

1. Item One

   (a) This is (a) and it's nested.

a	b
c	d

1. This is (b). The table is *not* nested inside (a). In fact, it's not nested at all.

2. Back out again.

Notice how item (b) is not only no longer nested, but is also the first item of a new list!

There's another trap you can fall into: nesting the table, but not going deep enough. LYX turns anything after the table into a new [sub]list.

1. Item One

   (a) This is (a) and it's nested.

a	b
c	d

(a) This is (b). The table is actually nested inside Item One, but *not* inside (a).

2. Back out again.

As you can see, item (b) turned into the first item of a new list, but a new list *inside* item 1. The same thing would have happened to a figure or an equation. So, if you nest tables, figures or equations, make sure you go to the right depth!

Then there are the so-called Floats. A Float is a block of text associated with some sort of label, but which doesn't have a fixed location. It can "float" forward or backward a page or two, to wherever it fits best. Footnotes and Margin Notes are floats, as are Table Floats, Figure Floats, Wide Table Floats, and Wide Figure Floats.[12] When you're editing a document in LyX, a closed Float looks like a superscripted word in red and goes wherever the paragraph it's in goes. However, because a Float has not fixed location in the final text, nesting has no effect on its actual location after you feed your document to LaTeX.

3.4.4 Usage and General Features

3.4.4.1 The Various Bindings

Now that we've told you what you can and can't nest, it's about time we explained *how* to nest things![13]

First, you have your choice of keybindings, a toolbar button, and a menu item. Selecting Change Environment Depth from the Layout menu *increases* the nesting depth by one level. You can also increase the depth by using the toolbar button showing a picture of a page, with two arrows on either side of an indented paragraph. Lastly, you can *increase* the nesting depth with the keybinding M-p Right, while M-p Left *decreases* the nesting depth.

There are two ways to nest text. You can select the text to nest, or you can simply move the cursor into the paragraph whose depth you want to change. The first method lets you alter several paragraphs at once.

Note that LyX only changes the nesting depth if it can. If it's invalid to do so, nothing happens if you try to change the depth. Additionally, if you change the depth of one paragraph, it affects the depth of every paragraph nested inside of it. It's hard to describe what exactly LyX does in this case. That depends

[12] The difference between the wide and non-wide floats is only relevant in two-column documents, where wide floats are two columns wide instead of one. In one-column documents, the menu entries for Wide Table Floats, and Wide Figure Floats are unavailable.

[13] *Author's Note: I debated putting this section before the previous two, but all three are so tied together, that it was hard to figure out the order. -jw*

specifically on what your text looks like. Your best bet is to simply play with changing the nesting depth and see what happens.

Lastly, we'd like to note the following:

- If you *decrease* the depth multiple times [e.g. using M-p Left repeatedly] LyX stops at the outermost level.
- If you *increase* the depth multiple times [e.g. using M-p Right repeatedly] LyX "wraps around." That is, when you reach the innermost level, LyX doesn't stop there, but returns to the outermost level if you try to increase the depth once more.

Note that the "innermost level" does not mean the innermost possible depth. You can't nest to an arbitrary level; there has to be something in the level above it.

Speaking of levels, LyX can perform up to a sixfold nesting. In other words, "level #6" is the innermost possible depth. Here's an example to display what we mean:

1. level #1 - outermost
    (a) level #2
        i. level #3
            A. level #4
                - level #5
                    – level #6

Once again, LyX has a maximum of 6 levels, regardless of which specific paragraph environments you're using at a given level.[14] That means that you can perform a sixfold nesting of a Description list, or a Verse environment, and so on. You can also mix environments, as we shall see later.

There are two exceptions to the sixfold nesting limit, and you can see both of them in the example. Unlike the other fully-nestable environments, you can only perform a fourfold nesting with the Enumerate and Itemize environments. For example, if we tried to nest another Enumerate list inside of item "A.", we'd get errors.[15]

3.4.5 Some Examples

The best way to explain just what you can do with nesting is by illustration. We have several examples of nested environments. In them, we explain how we created the example, so that you can reproduce them.

[14] Unfortunately, LyX doesn't enforce this limitation. If you try to exceed it, however, LaTeX will return errors when you go to produce output for your document.

[15] Once again, LyX doesn't enforce this limitation. If you try to exceed it, however, LaTeX will return errors when you go to produce output for your document.

3.4.5.1 Example #1: The Sixfold-Way and Mixed Nesting

#1-a This is the outermost level. It's a List environment.

  #2-a This is level #2. We created it by using M-Return followed by M-p Right.

    #3-a This is level #3. This time, we just hit Return, then used M-p Right twice in a row. We could have also created it the same way as we did the previous level, by hitting M-Return followed by M-p Right.
    This is actually a Standard environment, nested inside of "#3-a". So, it's at level #4. We did this by hitting M-Return, then M-p Right, then changing the paragraph environment to Standard. Do this to create list items with more than one paragraph - it also works for the Description, Enumerate, and Itemize environments!
    Here's another Standard paragraph, also at level #4, made with just a M-Return.

      #4-a This is level #4. We hit M-Return and changed the paragraph environment back to List. Remember - we can't nest anything inside of a Standard environment, which is why we're still at level #4. However, we *can* keep nesting things inside of "#3-a".

        #5-a This is level #5...

          #6-a ...and this is level #6. By now, you should know how we made these two.

        #5-b Back to level #5. Just hit M-Return followed by a M-p Left.

      #4-b After another M-Return followed by a M-p Left, we're back at level #4.

    #3-b Back to level #3. By now it should be obvious how we did this.

  #2-b Back to level #2.

#1-b And last, back to the outermost level, #1. After this sentence, we'll hit Return and change the paragraph environment back to Standard to end the list.

There you have it! Oh — we could have also used the Description, Quote, Quotation, or even the Verse environment in place of the List environment. The example would have worked exactly the same.

3.4.5.2 Example #2: Inheritance

```
This is the LyX-Code environment, at level #1, the outermost
level. Now we'll hit Return, then M-
p Right, after which,
we'll change to the Enumerate environment.
```

```
  1. This is the Enumerate environment, at level #2.

  2. Notice how the nested Enumer-
     ate not only inherits its margins from
     its parent environment [LyX-
     Code], but also inherits its font
     and spacing!
```

We ended this example by hitting Return. After that, we needed to reset the paragraph environment to Standard and resetting the nesting depth by using M-p Left once.

3.4.5.3 Example #3: Labels, Levels, and the Enumerate and Itemize Environments.

1. This is level #1, in an Enumerate paragraph environment. We're actually going to nest a bunch of these.

    (a) This is level #2. We used M-Return followed by M-p Right. Now, what happens if we nest an Itemize environment inside of this one? It will be at level #3, but what will its label be? An asterisk?

    - No! It's a bullet. This is the *first* Itemize environment, even though it's at level #3. So, its label is a bullet. [Note: we got here by using M-Return, then M-p Right, then changing the environment to Itemize.]

        – Here's level #4, produced using M-Return, then M-p Right. We'll do that again...

            i. ...to get to level #5. This time, however, we also changed the paragraph environment back to Enumerate. Notice the type of numbering! It's *lowercase Roman*, because we're the *thirdfold* Enumerate environment [i.e. we're an Enumerate inside an Enumerate inside an Enumerate].

            ii. What happens if we *don't* change the paragraph environment, but decrease the nesting depth? What type of numbering does LyX use?

            iii. Oh, as if you couldn't guess by now, we're just using M-Return to keep the current environment and depth but create a new item.

iv. Let's use M-p Left to decrease the depth after the next M-Return.

i. This is level #4. Look what type of label LyX is using!

i. This is level #3. Even though we've changed levels, LyX is still using a lowercase Roman numeral as the label. Why?!

ii. Because, even though the nesting depth has changed, the paragraph is *still* a thirdfold Enumerate environment. Notice, however, that LyX *did* reset the counter for the label.

(b) Another M-Return M-p Left sequence, and we're back to level #2. This time, we not only changed the nesting depth, but we also moved back into the twofold-nested Enumerate environment.

2. The same thing happens if we do another M-Return M-p Left sequence and return to level #1, the outermost level.

Lastly, we reset the environment to Standard. As you can see, the level number doesn't correspond to what type of labelling LyX uses for the Enumerate and Itemize environments. The number of *other Enumerate environments* surrounding it determines what kind of label LyX uses for an Enumerate item. The same rule applies for the Itemize environment, as well.

3.4.5.4 Example #4: Going Bonkers

1. We're going to go totally nuts now. We won't nest as deep as in the other examples, nor will we go into the same detail with how we did it. [level #1: Enumerate]

   [Return, M-p Left, Standard: level #2] We'll stick an encapsulated description of how we created the example in brackets someplace. For example, the two keybindings are how we changed the depth. The environment name is, obviously, the name of the current environment. Either before or after this, we'll put in the level.

2. [Return, Enumerate: level #1] This is the next item in the list.

   Now we'll add verse.
   It will get much worse.
   [Return, M-p Left, Verse: level #2]
   Fiddle dee, Fiddle doo.
   Bippitey boppitey boo!
   [M-Return]

   Here comes a table for you:

one-fish	two-fish
red-fish	blue-fish

   [M-Return, Table, M-p Right 3 times, M-Return, Verse, M-p Left]

3. [Return, Enumerate: level #1] This is another item. Note that selecting a Table resets the nesting depth to level #1, so we increased the nesting depth 3 times to put the table inside the Verse environment.

   We're now ending the Enumerate list and changing to Quotation. We're still at level #1. We want to show you some of the things you can do by mixing environments. The next set of paragraphs is a "quoted letter." We'll nest both the Address and Right Address environments inside of this one, then use another nested Quotation for the letter body. We'll use M-Return to preserve the depth. Remember that you need to use C-Return to create multiple lines inside the Address and Right Address environments. Here it goes:

   1234 Nowhere Rd.
   Moosegroin, MT 00100
   9-6-96

   Dear Mr. Fizlewitz:

   > We regret to inform you that we cannot fill your order for 50L of compressed methane gas due to circumstances beyond our control. Unfortunately, several of our cows have mysteriously exploded, creating a backlog in our orders for methane. We will place your name on the waiting list and try to fill your order as soon as possible. In the meantime, we thank you for your patience.
   >
   > We do, however, now have a special on beef. If you are interested, please return the enclosed pricing and order form with your order, along with payment.
   >
   > We thank you again for your patience.

   Sincerely,
   =-=-=-=-=

   That ends that example!

As you can see, nesting environments in LyX gives you a lot of power with just a few keystrokes. We could have easily nested an Itemize list inside of a Quotation or Quote, or put a Quote inside of an Itemize list. You have a huge variety of options at your disposal.

3.5 Fonts and Text Styles

3.5.1 Overview

Many modern typesetting and markup languages have begun to move towards specifying character styles rather than specifying a particular font. For example, instead of changing to an italicized version of the current font to emphasize text, you use an "emphasized style" instead. This concept fits in perfectly with LyX. In LyX, you do things based on contexts, rather than focusing on typesetting details.

Right now, LyX allows you to specify a global default font, and has two character styles, Emphasized and Noun. The Emphasized style corresponds to an italics font. The Noun style corresponds to a font in smallcaps, which some languages and writing styles use to typeset proper names. The LyX Team actually hopes to someday have a full set of character styles, and to also allow the user to customize which font changes correspond to what styles. At the moment, though, you'll have to be satisfied with what we've done already.

3.5.2 Global Options

You can set the default font from the Document Layout popup, activated using Document... from the Layout menu. There are two options of interest here, Fonts and Font Size. The possible options under Fonts include "default" and a list of fonts available on your system. The option "default" uses the standard TeX fonts, known as "computer modern" (cm) or "European modern" (ec). Most systems will typically have some version of a Times and Helvetica font, with other variants. You'll have to examine this for yourself.

As for the Font Size option, there are three possible values: 10, 11, and 12. Remember, this is the *base* font size. LyX actually scales all of the other possible font sizes [such as those used in footnotes, superscripts, and subscripts] by this value. You can always fine-tune the font size from within the document if you need to. It's also rather silly to use an 8pt or 24pt font as the default font size, as this typically renders your document unreadable.

Note that once you choose a new value for Fonts or Font Size, LyX does *not* change the screen. You'll only see a difference once you generate the final output. This is part of the WYSIWYM concept. Besides, if you're intelligent enough to use a computer, you're intelligent enough to recognize that "Roman" text on the LyX screen corresponds to the default font.

3.5.3 Using Different Character Styles

As we've already seen, LyX automatically changes the character style for certain paragraph environments. We also mentioned two other character styles, Emphasized and Noun. You can activate both of these styles via keybindings, the menus, and the toolbar.

To activate the Noun style, do one of the following:

- select Noun Style from the Layout menu
- click on the toolbar button with the person-shaped icon
- use the keybinding M-c c

These commands are all toggles. That is, if Noun style is already active, they deactivate it.

One typically uses the Noun style for proper names. For example: "Matthias Ettrich is the original author of LyX."

A more widely used character style is the Emphasized style. You can activate [or deactivate - it's also a toggle] the Emphasized style by:

- selecting Emphasize Style from the Layout menu
- clicking on the toolbar button with the "!" character on it
- using the keybindings M-c e

At the moment, the Emphasized style is equivalent to an italicized font. We have plans to make that association more user-configurable in the future.

We've been using the Emphasized style all over the place in this document. Here's one more example:

Don't overuse character styles!

It's also a warning in addition to an example. One's writing should parallel ordinary conversation. Since we don't all constantly scream at each other, we should also avoid the common tendency to overuse character style.

Oh — one last note: You can always reset to the default font using the keybinding M-c Space.

3.5.4 Fine-Tuning with the **Character Layout** Popup

There are always occasions when you'll need to do some fine-tuning, so LyX gives you a way to create custom character style. For example, an academic journal or a corporation may have a style sheet requiring a sans-serif font be used in certain situations.[16] Also, writers sometimes use a different font to offset a character's thoughts from ordinary dialogue.

Before we document how to use custom character style, we want to issue a warning yet again: Don't overuse character styles. Many modern word processors have a vast array of fonts available to them, providing you with the power of a printing press. Unfortunately, there is this tendency to overuse that power. The phrase, "Using a sledgehammer to swat a fly," comes to mind. And, as the old saying implies, documents that overuse different fonts and sizes tend to look like someone's knocked huge holes in it.

Enough complaining.

To use custom fonts, open the Character Layout popup using Character... from the Layout menu. There are six buttons on this popup, each corresponding to a different font property which you can choose. You can choose an option for one of these six properties, or select No change, which keeps the current state of

[16] Note from John Weiss: There is, in fact, such a style sheet for the LyX Documentation, since manuals need a certain amount of consistency.

that property. The item Reset will reset the property to whatever is the default for the hosting paragraph environment. You can use this to reset attributes across a bunch of different paragraph environments in a snap.

The six font properties, and their options [in addition to No change and Reset] are:

Family The "overall look" of the font. The possible options are:

- **Roman** This is the Roman font family. It's also the default family. [keybinding = M-c r]
- **Sans Serif** This is the Sans Serif font family. [keybinding = M-c s]
- **Typewriter** `This is the Typewriter font family.` [keybinding = M-c p]

Series This corresponds to the print weight. Options are:

- **Medium** This is the Medium font series. It's also the default series.
- **Bold** **This is the Bold font series.** You can toggle this series on or off using Bold Style from the Layout menu, or with the keybinding M-c b.

Shape As the name implies. Options are:

- **Upright** This is the Upright font shape. It's also the default shape.
- **Italic** *This is the Italic font shape.*
- **Slanted** *This is the Slanted font shape* (although it might not be visible on screen, this is different from italic).
- **Small Caps** THIS IS THE SMALL CAPS FONT SHAPE.

Size Alters the size of the font. You'll find no numerical values here; all possible sizes are actually proportional to the default font size. Once again, you don't feed LyX the details, but a general description of what you want to do.

The options [and their keybindings] are:

- **Tiny** This is the "Tiny" font size. [keybinding = M-s t or M-s 1]
- **Smallest** This is the "Smallest" font size [keybinding = M-s 2]
- **Smaller** This is the "Smallest" font size [keybinding = M-s S or M-s 3]

Small	This is the "Small" font size. [keybinding = M-s s or M-s 4]
Normal	This is the "Normal" font size. It's also the default size. [keybinding = M-s n or M-s 5]
Large	This is the "Large" font size. [keybinding = M-s l or M-s 6]
Larger	This is the "Larger" font size. [keybinding = M-s S-L or M-s 7]
Largest	This is the "Largest" font size. [keybinding = M-s 8]
Huge	This is the "Huge" font size. [keybinding = M-s h or M-s 9]
Huger	This is the "Huger" font size. [keybinding = M-s H or M-s 0]

We'll warn you *yet again*: don't go crazy with this feature. You should almost never need to change the font size. LyX automatically changes the font size for different paragraph environments - use that instead. This is here for fine-tuning *only!*

Misc — Here you can change a few other things at the character level. Options are:

Emph	*This is text with emphasize on.* This might seem like the same as *Italic*, but it is actually a bit different. If you use emphasize on italicized text, it will make it upright. In future versions of LyX, we hope to let you customize the exact behavior of this *logical* property.
Underbar	This is text with Underbar on. [keybinding = M-c u]
Noun	THIS IS TEXT WITH NOUN ON. Like Emph, this is a logical attribute. For the moment, it is equivalent to Small Caps, but that is bound to change some day.

LaTeX mode — This is text in LaTeXmode.
This is a special font attribute that is used when you want to insert raw LaTeX codes in your document. What appears in LaTeX mode is put verbatim into the LaTeX file.

Avoid using underbar if you can! It's a holdover from the typewriter days, when you couldn't change fonts. We no longer need to resort to emphasizing text by overstriking it with an underscore character. It's only included in LyX because it's also in LaTeX, and because some people *may* need it in order to follow style sheets for journal submissions.

Color — You can adjust the color of the text with this control. Of course, you need to have a color printer to exploit this, but you also need to have the color LaTeX package installed. Notice that `xdvi` is not able to display these colors. Besides No color, which is the standard "color", you can choose between Black, White, Red, Green, Blue, Cyan, Magenta and Yellow text. We won't illustrate here, because most people do not have a color printer.

You have a huge number of combinations to choose from.

Once you've chosen a new character style via the Character Layout popup, you can activate it using the toolbar button labelled "Font". [You can, of course, also use the usual OK and Apply buttons.] The toolbar button lets you toggle the state of your custom character style even when the Character Layout popup isn't visible.

As we stated earlier, to completely reset the character style to the default, use M-c Space.

We conclude with the same warning we've been spewing: Don't overuse the fonts. They are, more often than not, a kludge and a horrible substitute for good writing. Your writing should speak for itself — and will.

3.6 Printing and Previewing

3.6.1 Overview

Now that we've covered some of the basic features of document preparation using LyX, you probably want to know how to print out your masterpiece. Before we tell you that, however, we want to give you a quickie explanation of what goes on behind-the-scenes. We cover some of this information in sec. 7.1 as well.

LyX uses a program called "LaTeX" as its backend. [Actually, LaTeX is just a macro package for the TeX typesetting system, but to prevent confusion, we'll just refer to the whole magilla as "LaTeX."] Think of it this way: LyX is what you use to do your actual writing. Then, LyX calls LaTeX to turn your writing into printable output. This happens in a couple of stages:

1. First, LyX converts your document to a series of text commands for LaTeX, generating a file with the extension, "`.tex`".
2. Next, LaTeX uses the commands in the `.tex` file to produce printable output. It doesn't know anything about your printer, however. Instead, LaTeX produces what's known as a *device-independent* file, or Dvi for short. The actual output is in a file with the extension, "`.dvi`". Dvi files are completely portable; you can move them from one machine to another without needing to do any sort of conversion.

   NOTE: The dvi-file only contains what was in the LaTeX file itself. If you have included PostScript® pictures in your document, there will only be a link to these files. So don't forget these files if you move your `.dvi` file to another computer.

3. Consider the `.dvi` file to be the "final output." Once you have it, you can view it, print it, or convert it to other formats.
   (a) You can view `.dvi` files using a program called `xdvi`.
   (b) Some printers and Unix systems understand Dvi, and can print your `.dvi` file directly.
   (c) Nowadays, most printers understand the PostScript® format. LyX automatically converts the `.dvi` file to a PostScript® file for you when you go to print out your document. LyX will also let you preview a PostScript® version of your document using the program `ghostview`.
   One advantage of using PostScript® is that the converter program [called `dvips`] takes any PostScript® graphics you may have included in your document and puts it into the resulting PostScript® version of your document. It also includes any special fonts you may have used. That makes the PostScript® version much, much more portable than the Dvi version.

LyX does all of these steps automagically for you.

As you have seen, a lot of things happen before you get a hardcopy or a preview of your document. So, don't worry if printing requires a bit more time than with other word processors. The printed result is worth the wait. Quality always has its price.

3.6.2 Quick Viewing with `xdvi`

To get a look at the final version of your document, with all of the pagebreaks in place, the footnotes correctly numbered, and so on, select View dvi from the File menu. Then wait a while.

When all of the behind-the-scenes action is done, LyX calls the program `xdvi`. You can now look at the results. [If you want more info on the `xdvi` program, see the `man`-pages.]

Helpful-Tip: Keep the `xdvi` window open, maybe moving it to another desktop. Then, after you make changes to your document, just use Update dvi from the File menu. Now click on the `xdvi` window. The `xdvi` program will automatically reread the `.dvi` file and give you an updated view.

3.6.3 Viewing the PostScript® Version with `ghostview`

In general, using `xdvi` to view your document is the easiest and fastest way. There may be times, however, when you want to look at the PostScript® version. One reason is fonts.[17] You can use PostScript® fonts in a LaTeX document, but `xdvi` won't show this. You'll need to use `ghostview` or some other PostScript® file viewer to see the actual results.

To view the PostScript® version of your document, select View PostScript from the File menu. Then go get some coffee.[18] When all of the magic behind-the-scenes is done, LyX calls the program `ghostview`. You can now look at the results.

You've guessed what the Update PostScript command from the File menu does, haven't you? Remember to click once in the `ghostview` window after this command to update the view.

3.6.4 Printing the File

To print a file, select Print... from the File menu, or click on the toolbar button with the printer on it. This opens the Print popup.

Here's a list of what all of the different options do:

Print Three toggle buttons that determine which pages to print. You can choose:

- All Pages
- Only Odd Pages
- Only Even Pages

You can use these to print on both sides of a page even if your printer only prints single-sided. Just print the odd-numbered pages first, then turn the paper over and print the even-numbered pages.

Order Two toggle buttons that determine what order to print in. Some printers spit out pages face-up, others, face-down. By choosing a particular order, you can take the entire stack of pages out of the printer without needing to reorder them.

[17] Note from JOHN WEISS: Another reason is paranoia. I always like to look at the PostScript® file before I print it, just so I see exactly what went to the printer...

[18] Note from MATTHIAS ZENKER: A reasonably fast machine will not give you the time, however...

Print to Two toggle buttons tied to text boxes. You can print to either a file or a printer:

- Printer This is the name of the printer to print to.[19] The printer should understand PostScript® files.
- File The name of a file to print to. The output will be in PostScript® format.

  Be careful with this option, though. LyX currently uses a temporary directory to hold all of the different files that it generates. Unless you've disabled this feature, you need to specify the full path name for the output file. For example: "`/home/me/docs/myfile.ps`". If you don't specify the full path name, LyX writes the file to the temporary directory.[20]

Note that printing may need little time, since LaTeX, `dvips` and, if you don't have a PostScript printer, `ghostscript` have to process your document.

3.7 A Few Words about Typography

3.7.1 Hyphens and Hyphenation

In LyX, the "`-`" character comes in three lengths, often called the *hyphen*, the *en dash*, and the *em dash*:

1. hyphen - made with "`-`"
2. en dash – made with "`--`"
3. em dash — made with "`---`"
4. minus sign $-$ a "`-`" in math mode

You generate these by using the "`-`" character multiple times in a row. LyX automatically converts them to the appropriate length dash in the final output.

The three types of dash are distinct from the minus sign, which appears in math mode and has a length of its own. Here are some examples of the "`-`" in use:

1. line- and page-breaks (*hyphen*)
2. From A–Z (*en dash*)
3. Oh — there's a dash. (*em dash*)

[19] Note that this printer name isn't for the `lpr` command but for `dvips`. That means `dvips` has to be configured for this printer name. See the section 2.6.2 or the `dvips` documentation for details. The default printer can also be set in `lyxrc`.

[20] *Is this still true? -Asger*

4. $x^2 - y^2 = z^2$ (*minus sign*)

Those of you reading this from within LyX will see no difference, though there is one in the printed version.

One last note about hyphenation — LyX automagically breaks up words and inserts hyphens in English text. [Actually, it's LaTeX that does this, and it will also hyphenate words in *some* other languages.] The words won't be hyphenated until you generate the final output.

If, for some reason, LaTeX can't break a word correctly, you can set hyphenation points manually. This is done with the menu item Hyphenation Point under Special Characters in the Insert menu. Note that these extra hyphenation points are only recommendations to LaTeX. If no hyphenation is necessary, LaTeX will totally ignore them.

3.7.2 Punctuation Marks

3.7.2.1 Abbreviations and End of Sentence

When LyX calls LaTeX to generate the final version of your document, LaTeX automatically distinguishes between words, sentences, and abbreviations. LaTeX then adds the "appropriate amount of space": sentences get a little bit more space between the period and the next word. Abbreviations get the same amount of space after the period as a word uses.

Unfortunately, the algorithm for figuring out what's an abbreviation and what's the end of a sentence is really quite brain-dead. If a "`.`" is at the end of a lowercase letter, it's the end of a sentence; if it's at the end of a capitalized letter, it's an abbreviation.

Here are some examples of *correct* abbreviations and the end of a sentence:

- M. Butterfly
- Don't worry. Be happy.

. . . and here's an example of the algorithm going wrong:

- e. g. this is too much space!
- This is I. It's okay.

You won't see anything wrong until you view a final version of your document.

To fix this problem, use one of the following:

1. Use a Protected Blank after lowercase abbreviations (see section 6.2.5).
2. Use an End of sentence period found under the Insert->Special Character menu to force the use of inter-sentence spacing. This function is also bound to C-period for easy access.

With the corrections, our earlier examples look like this:

- e. g. this is too much space!
- This is I. It's okay.

Some languages don't use extra spacing between sentences. If your language is such a language, you don't need to worry about all of this. For those that do need to bother, there is help to catch those sneaky errors: check out the Edit->Check TEX feature described in *Extended Editing*.

3.7.2.2 Quotes

LYX usually sets quotes correctly. Specifically, it will use an opening quote at the beginning of quoted text, and use a closing quote at the end. For example, "open close". The keyboard character, ", generates this automatically.

You can change the behavior of the " key using the Quotes popup. Choose Quotes... from the Layout menu. Selecting the Double button makes the " key produce the sequence: “”. The Single button, in contrast, makes the " key produce: ‘’.

You can also select quotes for different languages via the Type option. There are five choices:

“Text”	Use quotes like this “double” or ‘single’
„Text“	Use quotes like „this“ or ‚this‘
„Text”	Use quotes like „this” or ‚this’
«Text»	Use quotes like «this» or ‹this›
»Text«	Use quotes like »this« or ›this‹

Again, this affects what character the " key produces.

On the other hand, if you want to produce a bona-fide quote character, type C-q. This produces: ".

3.7.3 Ligatures

It is standard typesetting practice to group certain letters together and print them as single characters. These groups are known as *ligatures*. Since LATEX knows about ligatures, your LYX documents will contain them, too. Here are the possible ligatures:

- ff
- fi
- fl
- ffi

- ffl

Once in a while, though, you don't want a ligature in a word. While a ligature may be okay in the word, "graffiti," it looks really weird in compound words, such as "cufflink" or the German "Dorffest." To break a ligature, use the LaTeX command "\/" between the letters. [Use M-c t or the toolbar button labelled TeX to mark text as LaTeX commands. See sec. 7.4.] This changes "cufflinks" to "cufflinks" and "Dorffest" to "Dorffest".

3.7.4 Widows and Orphans

In the early days of word processors, page breaks went wherever the page happened to end. There was no regard for what was actually going on in the text. You may remember once printing out a document, only to find the heading for a new section printed at the very bottom of the page, the first line of a new paragraph all alone at the bottom of a page, or the last line of a paragraph at the top of a new page. These dangly-bits of text became known as *widows* and *orphans*.

Clearly, LyX can avoid breaking pages after a section heading. That's part of the advantage of paragraph environments. But what about widows and orphans, where the page breaks leave one line of a paragraph all alone at the top or bottom of a page? There are rules built into LaTeX governing page breaks, and some of those rules are there to specifically prevent widows and orphans. This is the advantage LyX has in using LaTeX as its backend.

There's no way we can go into how TeX and LaTeX decide to break a page, or how you can tweak that behavior. Some LaTeX books listed in the bibliography [such as [3] or [4]] may have more information. You will almost never need to worry about this, however.

Chapter 4

Floats: Tables, Figures, Footnotes and Margin Notes

4.1 Footnotes

Unlike other typesetting programs, LyX uses "foldable" floats instead of displaying its footnotes at the bottom of the screen or somewhere else in your text. When you insert a footnote via, what else, Insert->Footnote, you'll first see a red box with a label "foot" appearing within your text. This box is LyX's representation of your footnote. You can enter your text into this box. If you click the "foot"–label, the box will "fold". You can access it at a later time by clicking on the word "foot" written as a superscript in red in the text, thereby "unfolding" the footnote.[1] You will not see any numbers within LyX. You don't need to worry about those, anyhow, because LyX does the numbering for you, as well as putting the footnote at the bottom of the correct page, when it processes your file. If you want to turn already existing text into a footnote, simply mark it and click on the footnote button (a picture of text with an arrow pointing to stuff in the bottom margin).

What LyX cannot do, yet, is take care of special needs like setting the footnote–numbering back to 1 after each section in the "`article`" document class or changing the counter–style. You'll need to insert LaTeX commands like those described in the *Tricks for Footnotes and Marginpars* section of "*Extended Editing.*"

NOTE: A float in LaTeX and LyX isn't a simple paragraph as with usual word processors. It is a complex text structure that may contain everything except floats. That means you can use all the layouts inside a float, even figures and tables. You may not need this too often, but if you do occasionally need it, it's a neat feature.

If you're intending to use the spellchecker, remember that floats *must be*

[1] To close this footnote, click on the red box on the left side.

open to be checked. In order to open or close *all* footnotes (and margin notes), choose Edit->Floats & Insets->Open All Footnotes/Margin Notes.

4.2 Margin Notes

This is a margin note.

Margin notes look and behave just like footnotes in LyX. When you insert a margin note via Insert->Margin Note or the toolbar button (which contains a picture of text in a margin with an arrow pointing to it), you'll see a red box with a label "margin" appearing within your text. This box is LyX's representation of your margin note. You can enter your text into this box. If you click the "margin"–label, the box will "fold", making an exclamation mark appear beside the line in which you inserted the margin note. You can access it at a later time by clicking on the word "margin", written as a superscript in red in the text, thereby "unfolding" the margin note.

As a default, LyX uses 1,9 cm (0.75 inches) as the margin width to allow room for margin notes. This might not be what you're looking for, but as with footnotes, LyX cannot yet do everything LaTeX has to offer. You might want to consult your LaTeX handbook for additional commands.

4.3 Figures and Imported Graphics

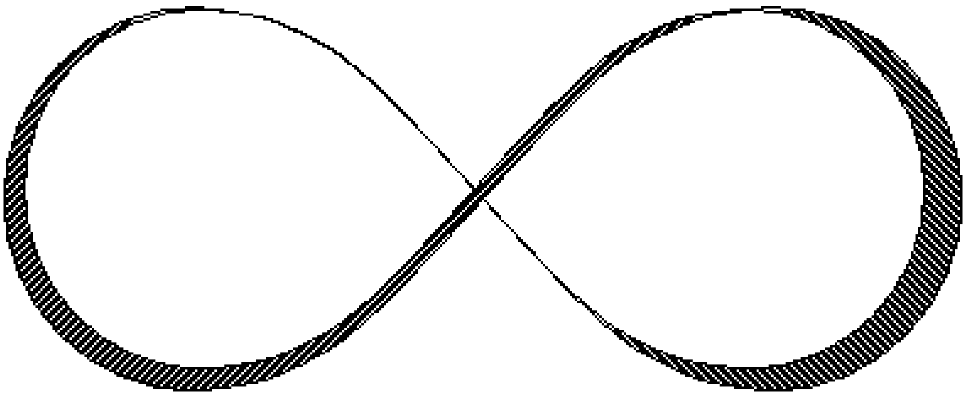
No document preparation system is complete without the ability to import graphics from other utilities into the document. In LyX, these are referred to as "figures" whether they are actually figures in the traditional sense or simply some kind of imported image. Encapsulated PostScript® figures are handled very well by LyX, which uses `ghostview` to generate an on-screen image for the LyX window, and the `\epsfig` LaTeX command to insert the figure in the final document.

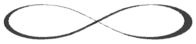
Note that figures referred to here are uncaptioned and sit wherever in the document you place them. If you need one of these features, see sec. 4.3.1.1 below.

Placing a figure in your document is done in two stages; first you tell LyX that you want a figure, then you open the blank figure and tell LyX the details about your figure. To place the figure, you click on the second right-most icon on the toolbar, or select Insert->Figure... from the menu.

You will be presented with a popup which contains two options:

Encapsulated Postscript: This is for inserting normal PostScript® figures. The figures will sit between lines on their own, such as here:



Inlined EPS: This is for PostScript® figures as well, but these figures behave differently; they are inserted in the text stream like  this tiny fellow.

This popup used to have another option to insert a LaTeX or TeX figure. This option is not present anymore, since you can directly insert any LaTeX file with the Insert->Include File command.

The second stage of manipulating a figure is done by clicking the left mouse button on the figure. A dialog box will appear, which allows you to enter the filename and various other parameters. The changes you make will not be applied to the figure until you hit Apply.

This dialog has numerous parameters, so I'll step though each one in order.

EPS File: This is the filename, and is fairly self-explanatory. The Browse... button lets you wander through the file hierarchy in a graphical manner and select a file with your mouse.

Width: This section lets you manipulate the width of the image. "Default" indicates that the image size will be whatever the "EPS file" says it should be. However, you can alter the size by checking one of "cm", "inches", "% of Page", or "% of Column"' and entering a number in the box to the right of those checkboxes.

Height: This is similar to "Width" above. Note that if only *one* of either the width or the height is specified and the other is set to "Default" then the ratio of width to height is will be kept the same as the original image.

Display: You can tweak the way the figure is displayed in LyX (this does not effect how the image will appear in the printed document, only on your screen). You may find that selecting Do not display this figure will speed up the performance of LyX. LyX will just put an empty box in the document.

Options: This box contains two options which didn't fit anywhere else. Display Frame places a frame around the figure in the LyX window (but has no effect in the final document). The Do Translations checkbox does something wonderful, I am sure. I have no idea what it is though *[Help me out here - what's it do? - PR; Ed. Note-I'm not sure myself -jpw]*

Full Screen Preview: This option simply calls up ghostview to view the PostScript® figure in a window by itself. This can be useful for checking the figure outside of LyX.

4.3.1 Figure Floats

4.3.1.1 Using Figure Floats

The problem with inserting figures straight into your text is that they might make the pagination of your document extremely awkward. To suit the LyX mentality of automating such processes, you might find it preferable to use Figure Floats, which LyX (actually, LaTeX) is free to move about your document as it deems necessary for a good fit. In return, LyX automates the listing of these figures and allows you to place a caption on them, using the Caption environment explained in sec. 3.3.9.1.

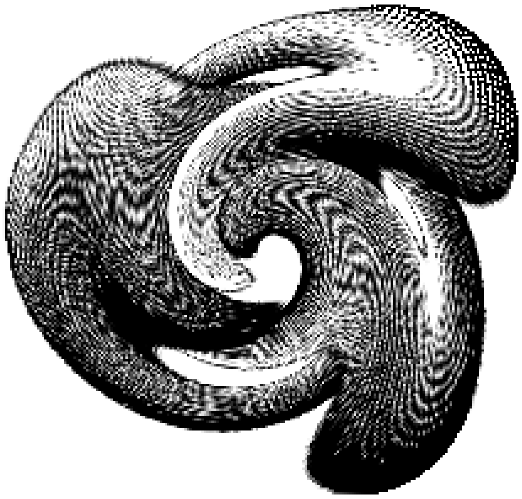
To place a Figure Float simply select Insert->Floats->Figure Float from the menu bar. You will get a float without a figure in it; use the toolbar icon described above to insert the actual figure. In two-column documents, you can select Insert->Floats->Wide Figure Float to get a float that spans the width of the page.

It seems simple, but there is subtlety involved in the placement of the caption. If you prefer your caption to appear below the figure, then insert the Figure when the cursor is at the very start of the caption (or you can delete the caption and recreate it by selecting the Caption environment after the figure has been inserted). This is what we did for figure 4.2. If the cursor is at the end of the caption when you insert the Figure then it will be inserted after the caption, as was the case for 4.1. It is always best to use one Figure per Float. This allows LyX [actually LaTeX] to best position each figure.

This figure also shows how we place a label and create a cross-reference to it; as you would expect from reading section 6.1 you can simply insert a Label in the caption and refer to it using a Cross-Reference as normal. It is especially important to use these with figure floats, rather than using vague references to "the above figure," as LaTeX will reposition your floats for you in the final document; it might not be "above" at all. If it is not possible to fit the floats neatly on the same page as the text which refers to it, the figures will be placed on a separate page by themselves. Rest assured that the overall effect is usually quite nice.

There are two more features of the Figure Float and Wide Figure Float; the figure is automatically included in a List of Figures (as described in sec. 4.5.2) should you choose to include one in your document. The second feature is that

Figure 4.1: M.C. Escher on acid.



(like all floats) it can be closed or opened, which lets you concentrate on your text and hide the figures.

4.3.1.2 Float Placement

Now, the whole idea behind Figure Floats [as well as Table Floats, which we introduce later] is to allow LyX to place a figure [or table] on a page in a consistent, sensible fashion. This feature would be useless if you couldn't control that "sensible fashion" in some way. LyX has four different locations where it will try to put a figure:

Here: LyX tries to put the Float at the same point in the text where you put it.

If there isn't enough room, LyX tries one of the other three location types.

Top: LyX tries to put the Float at the top of the current page. If the figure won't fit on the current page, it goes to the next page.

Bottom: LyX tries to put the Float at the bottom of the current page. If there isn't room, it goes to the next page.

Page: LyX tries to put the Float on a page of its own.

You can specify this using the Document Layout popup [opened using Layout->Document]. There is a box called Float Placement which controls the global location of Floats.

Now, there is some subtlety to how this all works. You can place any combination of four letters in the Float Placement box, in any order:

Figure 4.2: A severely distorted platypus in a float.

- h for *here*
- t for *top*
- b for *bottom*
- p for *page*

The order specifies what location LyX should try first. If that one fails, it tries the next one, and so on. Here are some example entries and what they do:

1. hbp

   Try putting the figure/table at its actual position in the text. If that doesn't work, put it on the bottom of the page. If that fails, put it on a separate page.

2. tp

   Put the figure/table at the top of each page. If it's too long, put it on a separate page.

3. p

   Always put figures and tables on their own page.

One last note: Float Placement controls the position of *both* figure and table floats.

4.3.2 `XFig` and LyX

One obvious question is "how would I create the figures?" Fortunately, the answer is included in most Linux and/or LaTeX distributions. `XFig` is a powerful and highly recommended drawing tool. If you want to include figures that you have created with `XFig` there are several ways. We recommend the following:

1. Export the figure as Encapsulated PostScript®. This could be very easy included into LyX as described in the previous sections. The great advantage of this way is, that you have the full power of PostScript® available. That means Bezier curves, colors, all line thicknesses and many more. If you have inserted text into your fig-document this will be printed with PostScript® fonts, which is OK. The figure can be manipulated like any other EPS figure, as described above.
   The only disadvantage is that you cannot create formulas as PostScript® text except by hand. If you also need formulas or simple exponents or indices in your figure, the next way is recommended.

2. Export the figure as LaTeX. This is just as easy include into LyX, with the advantage that you may use all LaTeX commands within the text inside XFig. Therefore you have to set the *special flag* for text in XFig. This is automatically if you invoke XFig with `xfig -specialtext`. If this is

done and you have also chosen a LaTeX font you may simply write "H_2" in `XFig`. If you export this figure as LaTeX and include it in LyX with Insert->Include File (see description in *Extended Features*) this text will appear as "H_2".

The disadvantage of this way is that the graphical power of LaTeX isn't as strong as PostScript®. You cannot use all thicknesses of lines and, more annoyingly, not all slopes. This is why we recommend the third way for more complex figures.

3. Export the figure as LaTeX/PostScript® combined. Then `XFig` [`transfig`, really] will generate two files:

    (a) the PostScript® part `foo.ps_tex`, that contains all painting.

    (b) the LaTeX part `foo.ps_tex_t`, that contains all text and a link to the PostScript® part.

Then you just have to include the LaTeX part as described above. This will automatically include the PostScript® part, too.[2] This way you have the full PostScript® and LaTeX power combined except for the possibility to scale the figure after creating. So if you want scalable pictures the PostScript® format is your only choice. Another little advantage of letting LaTeX typeset the font is that the same font will appear in your figures as in your text, which looks a little nicer.

4.4 Tables

LyX now has the capability of creating and editing WYSIWYM tables. LaTeX can do many more things with tables than LyX is currently capable of, so you might want to look at a good LaTeX book if the table feature described here should turn out to be inadequate to fill your needs.

You can insert a table using either the table button or Table->*Insert Table* from the Edit menu. A popup will appear, asking you for the number of rows and columns. The default table has lines at the top and to the left of every cell, a line to the right of the rightmost column and a line at the bottom of the lowest row, forming a box around the table. Additionally, the topmost row also has a line at the bottom, which causes this row to appear separated from the rest of the table. Here's an example:

---

[2]If you get an error like "unknown graphics extension ps_tex" you have to declare this graphic extensions. I think this is a transfig bug that occurs with LaTeX 2_ε. Simply add a line like

\@namedef{Gin@rule@ps_tex}#1{{eps}{ps_tex}{#1}}

in the file `/usr/lib/texmf/tex/latex/graphics/dvips.def`. Then add `ps_tex` to the extension:

\def\Gin@extensions{eps, ps, ps_tex, eps.gz, ps.gz, eps=2EZ}

This should fix the whole thing. Alternatively you may export the postscript part as `foo.eps` and change the LaTeX part `foo.ps_tex_t` manually. But this is annoying.

	12	45	98
A			
B		multi	
C			

4.4.1 The Table Layout pop-up

You can alter a table by clicking on it with the right mouse button, which pops up a window of options. Among these options are:

- Adding/removing border lines from a row or column. If you remove the top line from one of the rows, you'll get a dotted line in LyX, but no line will appear in the printout. If you set the bottom line of one row and the top line of the row below, then the rows are separated by a small space, as you can see with the top row in the example above. You can do the same vertically if you set the right line of a column and the left line of the column to the right.
- Text alignment in a column
- Appending rows and columns
- Deleting rows, columns, or the entire table
- Multicolumn
- Setting a fixed width for a column
- Longtable options - this is useful if your table is higher than the paper. Then the table is split on the bottom of the page and continued on the next one
- Rotate the whole table or a single cell sideways, by 90 degrees

You can also use the menu to perform these options. Try Edit->Table when the cursor is inside a table.

Most of these options also work on selections. This means that if you select more cells, columns or rows (you always have to select cells, but it depends on the action, how these selected cells are interpreted) the action is done on all of your selection.

When you append a row, it is added *below* the row containing the cursor. Similarly, columns are appended to the *right* of the cursor. This makes it difficult to add columns on the left edge of a table without a lot of cutting and pasting. Deletion is always performed on the row or column containing the cursor.

Multicolumn merges two adjacent cells on a given row. For example, in the above table, row "B" has had multicolumn applied to the columns labelled "45" and "98." To use it, you must first select the cells, then choose Multicolumn from the menu.

You can also use Multicolumn if you need to have a special handling for a single table cell, regarding then top and bottom border-lines and the text alignment. Here an example of this special handling of a cell:

*	x	y
point a & b	103	9
point b & a	599	340
point abc	1009	52
point abcd	96	11

You see here that the header line cells are aligned to the center, were as the left column is aligned to the left and the other columns are aligned to the right. Also the bottom and top line of two cells have been removed.

If you want your column to have a fixed width, then you can insert a width in the Width input-field of the Table Layout popup.

If your table becomes too large to fit on a portrait document layout, you can select the Rotate 90° button, and the table will appear sideways (this means landscape in a portrait document style). You might also like to rotate single table cells for having more horizontal space. Look at the following example, which demonstrates what you may need this feature for.

Notice: This Rotate 90° option will *not* display on screen, and works *only* for PostScript® output. So, if you want to preview them, use View PostScript, as View dvi will not work.

Description	Flag 1	Flag 2	Flag 3	Flag 4	Flag 5	Flag 6	Flag 7	Flag 8	Flag 9	Flag 10	Total
Desc. 1	*		*	*		*		*	*	*	7
Desc. 2		*	*	*	*	*	*	*	*	*	9
Desc. 3	*	*	*		*	*	*	*		*	8
Desc. 4	*		*	*			*	*	*		6
Total											30

As you can see in the above example, the text in the header-line (Flag 1...Flag 10) would not fit inside the paper width if they weren't rotated.

4.4.2 What can be placed inside a table?

Many objects can be placed inside a table cell. Any single line of text, a math inset (not a displayed or multilined equation, though), or a figure can be in a cell, in fact, all three kinds of objects can be placed in the same cell. Font sizes and shapes can be altered, and the table will adjust to display them properly. However, you can't put a table inside a cell of another table, nor can you put a special environment in a cell (like Section*, etc.), since LyX then tries to put the whole table in that environment, treating it like a paragraph. The results of that will be unpredictable.

4.4.3 Cut & Paste & Tables

Cutting and pasting between tables works reasonably well. You can cut and paste even more than one row. Selection with the mouse or with Shift plus the arrow-keys works as usual. The values in the second table below were cut and pasted from the first, using the mouse to select and paste.

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7		

Note that the cutting and pasting may give undesirable results unless the target table has the same number of columns as the source table.

To be able to copy/cut and paste a *whole* table you need a little trick. There has to be a paragraph *before* and *after* the table. This means that if the table is the first and/or last entry in your document, you have to create paragraphs around it! Now select the whole table starting from the last character of the paragraph before the table till before the first character of the paragraph after the table. Now you will cut/copy the entire table including the structure, and not only the contents of the table.

4.4.4 Multiple lines in cells

It is possible to have multi-line entries in tables, but not in a completely WYSIWYM manner. There are several methods that can be used to achieve this:

1. Define a fixed length for the column in the Table Layout popup. After this, your text is automatically split into more lines and the cell enlarged vertically, when the length of the text exceeds the given fixed length.

2. Remove the line between two (or more) adjacent rows. This gives only an approximation of a truly multi-line cell.

3. Define a fixed length for the column in the Table Layout popup and press the Linebreaks button. After this you should be able to force linebreaks by inserting "\\" marked as TeX, but this shouldn't be needed as selecting Linebreaks means that the linebreaks are WYSIWYG in the resulting document.

   Here is an example table:

1	2	3
4	This is a multiline entry in a table.	5
6	This is longer now.	7
8	This is a multiline entry in a table. This is longer now.	9
10	This is a *multiline* entry in a table. This should be the same as below.	11
12	This is a *multiline* entry in a table. This should be the same as above.	13

   You can see that the multilined text will not appear that way within LyX, but it will print that way. If the lines are very long, you can hit C-Return and you'll get a new row in the same cell where you can continue to write. Be aware that the output will *NOT* respect this row breaks but will treat the whole text of the cell as if inputed in one row.

A related concern is that text within a cell will not wrap to fit the page, so if a line of text in a table is too long, the table will extend beyond the right margin of the page. Similarly, LyX's table inset will not split itself at the bottom of a page, and so might extend below the bottom margin. You have these options to resolve this problem:

1. Split it into two tables, to correctly handle pagebreaks and margins.
2. Push the Longtable button in the Table Layout popup. This automatically splits the table over more pages, if it is too high. After doing this, the list of Longtable buttons activate themselves and you may now define:
   (a) FirstHead: The current row and all rows above that don't have any special options defined, are defined to be the header-lines of the first page of the multipage-table.

(b) Head: The current row and all rows above that don't have any special options defined, are defined to be the header-lines of all pages of the multipage-table except for the first page if FirstHead is defined.

(c) Foot: The current row and all rows above that don't have any special options defined, are defined to be the footer-lines of all pages of the multipage-table except for the last page if LastFoot is defined.

(d) LastFoot: The current row and all rows above that don't have any special options defined, are defined to be the footer-lines of the last page of the multipage-table.

(e) NewPage: This forces a pagebreak after the row where this flag is defined.

If you define more flags in the same table row, you should be aware of the fact that only the first flag is used in the defined table rows. The others will the be defined as *empty*. In this context, first means first in this order: Foot, LastFoot, Head, FirstHead [*Look at the longtable.lyx example file to see how this works – jv*].

3. Use the Width entry in the Table Layout popup to restrict the width of the table till it fits horizontally.

4. A table can also be placed in a float, as described below, which will allow TEX to place it as well as it can within the page.

One last remark: Longtable and Rotate 90° use special LATEX packages, so you should look into LATEX configuration in the Help menu to see if your system supports these features.

4.4.5 Table Floats

If you simply use the table button on the icon toolbar to create your table, or the Edit->Table->Insert Table menu, the table will be positioned exactly where the cursor was located, displayed on a separate line and centered. Using a Table Float or Wide Table Float, from the Insert->Floats menu, will enable LATEX to place the table where it fits best, rather than exactly where you insert it. Float placement for table floats is similar to that for figure floats 4.3.1.1, and is described in the section on float-placement for figures 4.3.1.2. Table 4.1 is an example of a table float, whose insertion point in LYX is at the end of this sentence. As with Figure Floats, the difference between a Table Float and a Wide Table Float is that the wide version spans both columns in a two-column document.

4.4.5.1 Table Float Captions.

When you open a table float the first thing you are prompted for is a caption. To actually create the table, click on the table icon on the toolbar when the

Table 4.1: A table float.

1	2	3
Joe	Mary	Ted
$\int x^2 dx$	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	$1+1=2$

1	2	3
4	5	6
7	8	9

Table 4.2: A second table float, with the caption below.

cursor is in the float. You can control whether the caption is above the table, or below. If you insert a caption (which must be a single paragraph) immediately after the word "**Table:**" inside the float, and click on the table icon after the caption, the caption will appear above the table, as in Table **??**. If you first click on the icon to open the table popup, then write the caption afterward, the caption will appear below the table, as in Table 4.2. You can also create a table float without a caption, by creating the table as a displayed table (not within a float), highlighting the whole thing, and using the Insert->Table Float menu item. This was done for Table 4.4.5.1 , which, since it has no caption, will not be numbered, even though the reference is numbered. It is not possible to place a caption with a table unless it is a table float. Note that labelling the tables automatically numbers them, and cross-references (cf. 6.1) will display that number in the output.

4.4.5.2 Float Placement

The techniques of how to control the placement of table floats is the same as that for Figure Floats. See section 4.3.1.2 for details.

4.5 Table of Contents and other Listings

One of the really nice features of LaTeX is the ease with which it lets you create various "Lists," such as a ToC (Table of Contents). All you need to do is to use certain environments and insert a reference at the place where you want the list

1	2	3
4	5	6
7	8	9

to appear.

If you are a LaTeX user or are used to an older version of LyX, you might remember that LaTeX should be run several times to get things right. Fortunately, this is not needed anymore. LyX will make sure that LaTeX is run a sufficient number of times in order for the lists to appear in your dvi–output.

4.5.1 The Table of Contents

In order to get a Table of Contents (ToC), you need to do 5 things:

1. Use a document–class that supports ToC (all but `letter`).
2. Use styles that are supported by the ToC function: Chapter, (Sub...) Section, (Sub...) Paragraph. Note that styles with a *, like Section*, will *not* appear in the ToC.
3. Make sure you set the Section number depth and Table of contents. depth in the Document Layout popup to the appropriate value. If, for example, you only want chapters and sections to appear in the ToC, set the value to 3.
4. Insert the ToC– command at a place of your liking. You'll find it under Insert->Lists & TOC->Table of Contents.

But there is something else that you can do with a ToC. Whether or not you've created a ToC in your text, you can invoke the function Edit->Table of Contents. You will get a second window with a ToC that lets you navigate the text.

4.5.2 List of Figures, Tables and Algorithms

Lists of Figures (LoF), Lists of Tables (LoT) and Lists of Algorithms (LoA) are very much like ToCs. There are some slight differences, though, that you need to heed. Although the following description is for Lists of Figures, the same procedure applies to LoTs and LoAs.

1. If you want you figures to appear in your LoF, you *must* use figure floats. You can find them under Insert->Floats->Figure Float.
2. Insert the LoF – command at a place of your liking. Select either List of Figures, from the Insert->Lists & TOC submenu.
3. [*Author's note: the astute reader might notice that there is no mention of algorithms in this User's Guide. This is because algorithm floats is a new feature that is not yet documented -jml*]

Chapter 5

Mathematical Formulas

5.1 Basic Math Editing

Editing mathematical expressions in LyX is now almost-completely WYSIWYM. This is something that makes many old TeX users worried[1]. They like to be able to use the keyboard to enter things like `$\alpha$` (this gets, in ordinary TeX, an α in the final document), believing that it is faster than chasing around menus for a symbol. Here's a testimonial of one of those old TeX users, David Johnson:

> I was finally convinced that the math-editor (Mathed) was the way to go when I found that, with a few modifications, I could use it the same way I was accustomed to writing TeX. As an example, I created this α by typing the following keys: First type M-c m to enter `math-mode`, then type `\alpha` , then Space and Esc. As soon as I typed that Space the α was right there on the screen.

The M-c m sequence starts `math-mode`, the `\alpha` is of course the standard TeX code, and the Esc leaves `math-mode`. It is a little slower getting in to `math-mode` than with standard TeX (for this mode, maybe, at least), but the advantages:

- You have immediate visual feedback to be sure your TeX was correct,
- You have the real mathematical expression on the screen, correctly displayed, to make sure your mathematics is correct (correctly written, at least),
- All the new LaTeX fuss with special environments and such are taken care of by LyX, not you,
- You won't have to chase through the code trying to find that missing `$` or extra { any more.

[1] *Author's Note: I'm one of them. - dlj*

- If you don't remember the LaTeX name of a particular symbol, like $\wp$, you can find it in the popups.

– these advantages mean that you will be faster writing mathematics with LyX than with raw LaTeX.

5.1.1 Invoking Mathed

There are several ways to invoke Mathed. You can just click on the icon with $\frac{a+b}{c}$ on it. That will open a little blue square, with a purple rectangle around it. That blue square is the math-cursor position, and the rectangle indicates that you are in math-mode. There is also a choice on the Math menu that does the same thing, or you can use a keyboard macro, M-c m, or C-m. Clicking on the icon will list all the equivalent keyboard macros to start Mathed. Starting Mathed in either of these ways produces an in-line Mathed box. To make it a displayed equation box, select Display from the Math menu.

If you are using `math-mode` to simply type, say, a Greek letter, α, there is a special shortcut, just type M-m g a to get α, M-m g b to get β, *etc.*

You can also invoke Mathed from the Math Panel popup, or Math Panel for short. The Math Panel is incredibly useful, so you may want to open it and leave it somewhere on the screen. Select Math Panel from the Math menu to open it. Selecting anything from the Math Panel will also start Mathed.

5.1.2 Navigating an Equation

You can click on an existing Mathed equation (anything in blue text) and you will automatically activate Mathed, and place the cursor near where you clicked. The best control over cursor position within an existing equation is achieved with the arrow keys. Mathed uses small squares to indicate places where something can be inserted, since Mathed needs more than simple linear arrangement of symbols.

The arrow keys can be used to navigate between text areas in any Mathed structure. Pressing an arrow key when at the edge of Mathed will leave Mathed, if the arrow is pointing out of the Mathed box. Pressing Space will leave a fraction or other Mathed structure (a square root $\sqrt{2}$, or parentheses (f), or a matrix $\left[\begin{array}{cc} 1 & 2 \\ 3 & 4 \end{array}\right]$), staying within Mathed. Pressing Esc will exit Mathed, placing the cursor to the right of the box. Pressing Space will also exit Mathed, but place the cursor one space beyond the box (creating that space if it is not there. Tab can be used to move horizontally in a Mathed structure, like the rows of a matrix or the positions in a multiline equation.

Space seems to do nothing in Mathed, since it does not in fact add a space between characters, but it does exit a nested structure. For that reason, you have to be careful about using Space. For example, if you want $\sqrt{2x+1}$, type `\sqrt` then Space, then 2x+1, not `\sqrt` Space 2x Space + Space 1, since in the latter case only the $2x$ will be under the square root sign, $\sqrt{2x}+1$. For those who learned to space out expressions in this way, it takes a little unlearning.

Speaking of the Space bar, you may want to create blank spaces, beyond the standard spacing that LaTeX provides. We don't recommend this as a matter of course, since the whole idea of WYSIWYM is that you don't think about the typesetting, but the content. However, there are situations where you will want to add spaces. The first thing to do is to type C-Space. This generates a small space, and prints a small red underscore on the screen: $a\,b$. The next trick is to change that space to different sizes. *Before* you move the cursor, after typing C-Space, if you hit Space again, you will change the size of the space, through a number of variable sizes. The last one in the list is blue, and is a negative space. For example: $a\quad b$, or $a\!\!\!b$.

You can leave many Mathed structures, like this matrix, partially filled in, such as:

$$\begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}.$$

If you leave a fraction only partially filled in, or a subscript with nothing in it, the results will be unpredictable, but most objects don't mind.

5.1.3 Selecting Text

You can select text within Mathed in two different ways. Place the cursor at one end of the string of text you want, and press Shift-Arrow to select text. It will be highlighted as with regular text selection. Alternately, you can select text with the mouse in the usual way. That text can then be copied or cut by pressing the menubar icons, and can then be pasted within any math inset (not outside of LyX, or even in a plain text region in LyX, though). Selected text from elsewhere in LyX cannot (yet) be pasted into a math inset, nor can selections from outside of LyX be pasted into a math inset.

5.1.4 Exponents and Subscripts

You can use the Math menu to add in superscripts (Exponent on the menu) or subscripts (called Index), but the much easier way is to use the standard TeX method. To get x^2, type (in Mathed) `x^2` then Space. The final Space puts the cursor back down on the base line of the expression, instead of the superscript. If you type `x^2y`, you will get x^{2y}, to get x^2y, type `x^2` then Space then `y`. Subscripts are similar, to get a_1, type (in Mathed) `a_1` then Space.

5.1.5 Fractions

Create a fraction with either `\frac` (in Mathed) or using the fraction icon in the Math Panel popup or the Math menu item Fraction. You will be presented with an empty fraction, with two Mathed insertion squares top and bottom. The cursor moves immediately to the top of the fraction. To move to the bottom, simply press Down. To move back up, press Up. Any math structure can be placed in a fraction, as this example shows:

$$\left[\frac{1}{\left(\begin{array}{cc} 2 & 3 \\ 4 & 5 \end{array}\right)}\right]$$

5.1.6 Sums and Integrals

Sum $\sum$ and integral $\int$ signs are very often decorated with one or more sets of "limits", which are placed in LyX (as in TeX) as superscripts and subscripts. Sum will automatically place their "limits" over and under the symbol in displaystyle, but will move them to the side when inlined, such as $\sum_{n=0}^{\infty}\frac{1}{n!} = e$, versus

$$\sum_{n=1}^{\infty}\frac{x^n}{n} = \ln\left(\frac{1}{1-x}\right).$$

Integral signs, however, will not by default move the limits to directly over and under the integral sign in displaystyle, as in $\int_a^x f(t)dt := F(x)$, versus

$$\int_{-\infty}^{\infty}\frac{dx}{1+x^2} = \pi.$$

Both symbols will be automatically re-sized when placed in display mode. In display mode , the placement of the limits (directly above and below, or offset to the right from the sign) can be changed by placing the cursor in front of the sign and hitting M-m l. Exactly what change occurs depends on the sign. Certain other mathematical expressions have this feature as well, such as

$$\lim_{x\to\infty} f(x),$$

which will place the $x \to \infty$ underneath the "lim" in display mode, but not in inlined mode, $\lim_{x\to\infty} f(x)$. The lim, in red, has a special significance. It is considered as a special object in LaTeX, so that it treats subscripts that way. You get it in LyX by typing `\lim` in math-mode, or choosing from the "functions" menu in the Math Panel popup. Other special "words" are sin and most other trigonometric functions; described below, see 5.1.9.

5.1.7 The Math Panel

The Math Panel popup has a more extensive list of symbols and structures. As stated earlier, you can keep the Math Panel open when writing mathematics. That allows you to have easy access to the less trivial features of `math-mode`.

The first two icons in the Math Panel, as indicated, produce square root, $\sqrt{2}$, or fraction $\frac{1}{2}$ regions. The third icon, with square brackets around an insertion square, pops up the Delimiter popup with choices for left and right large delimiters, $\{3\rangle$. These will adjust to fit whatever is placed inside them. The fourth icon calls up the Decoration popup, which gives choices for bars, arrows, tildes and braces, placed either above or below the insertion point, $\underbrace{\widehat{V}}$. These also adjust to accommodate any Mathed structure. The next icon pops up the Math space panel, which allows the possible types of protected spaces available in math mode $2 \qquad 3$. Sixth is the Matrix popup, which has controls for the size of matrix created, as well as alignment $\begin{bmatrix} 1 & 2 & 3 \\ 45 & 67 & 89 \end{bmatrix}$. The braces around a matrix are added using the Delimiter popup. The last icon, if clicked while the cursor is in text mode, creates an empty math-mode region:

$$1 + 2 = 3.$$

It will be created in display-math mode. Clicking this icon while in math-mode will toggle the math-insert between inlined and display mode.

In addition, the Math Panel has several submenus containing math symbols. We'll outline how to use them in the next section.

5.1.8 Other Math Symbols

Commonly-used symbols, and other menus, can be reached directly from the Math menu, including Fraction, Square root, Exponent, Index, Sum, Integral, Math Mode, and Math Panel... . Most math symbols can be found in the Math Panel popup under one of several categories, Greek $\Gamma\rho\epsilon\epsilon\kappa$, operators $\pm\times$, relations $\leq\cong$, arrows $\uparrow\Leftrightarrow$, large operators $\sum\int$, and the dreaded miscellaneous, *Misc*. If you know the standard LaTeX macro for a particular symbol you which to use, you do not have to use these popups, but they will help for those symbols whose LaTeX name you do not know. For a symbol whose LaTeX macro you know, all you need to do is type it, including the leading \ that LaTeX uses, such as in `\alpha`, and it will be converted automatically to the real symbol as soon as you type a Space, or other non-alphabetical symbol. The text as type it appears in red, (the \ will not appear) but will change to a blue symbol that tells you visually that LyX understands what you wanted. If Mathed is not able to convert the symbol as typed, it will remain in red (this is called *macro mode*).

Not all symbols available in LaTeX can (yet) be displayed WYSIWYM in LyX. If you try a symbol whose LaTeX name you know, like `\oint` for $\oint$, when you finish typing that string, and then hit Space (or a number, or another non-standard character), LyX will display the symbol whose macro you typed if it can. If it can't, it will leave the macro in red, such as $A \hookrightarrow B$. If LaTeX can handle that macro, it will print accordingly. Some of the symbols accessible from the Math Panel will display in this fashion, as will any that are defined in LaTeX but not yet in the LyX menus. This also allows you to define your own shortcut macros (put the command string in the preamble (see 7.5.1). For example, the string:

```
\newcommand{\a}{\alpha}
```

in the preamble will allow you to use `\a` in Mathed instead of `\alpha` to generate α. One caution, though, is that LyX will not concern itself with whether such a macro is valid, and if not you will have error messages when you try to preview or print (or run LaTeX). For more details, see 7.4.

5.1.9 Math functions

The Math -> Math Panel... menu contains a number of "functions", such as sin, lim, *etc.* (you can type them in a Mathed inset by typing `\sin` (*etc.*). Standard mathematical practice is that functions which are names, like sin, should not be italicized. Entering just the letters *sin* within Mathed will give italics, of course, so these special macros are available. they do more, however, to the final output than just change the typeface. For example, the expression $\sin t$ will typeset with a little extra space between the n and the t. For words which are more sophisticated mathematical objects, like lim, the macro changes the way that subscripts are placed, depending on whether the math-inset is inlined or displayed: $\lim_{x\rightarrow0}f(x)=L$ versus

$$\lim_{x\rightarrow0}f(x)=L.$$

These two expressions were typed the same way, but using the macro `\lim` (or the menu choice) alters the appearance (actually, it is the inlined version that is altered, to improve linespacing).

5.1.10 More symbols

There are some mathematical symbols that are not accessible from plain LaTeX (or LyX), but are fairly common in mathematical typesetting, such as the old-German Fraktur font and the stylized "blackboard bold" fonts commonly used to denote the real or complex numbers, or the integers. Adding these packages is now a menu item, in the Layout -> Document... popup there is a radio button marked Use AMS Math. Once activated, all AMS-LaTeX symbols and environments are available. You will run into trouble if you include these packages from the preamble, since LyX now defines a few of the macros used in these packages on its own. The AMS layouts include these packages automatically.

5.1.11 Accents

In math mode you insert accented characters in the same way as in text mode. This may depend on your keyboard, or the bindings file you use. You can also use TeX macro equivalents, as macros. That is, you can enter $\hat{a}$ to get the same effect if your keyboard does not have accents enabled. This is entered by typing "`\hat a`" in Mathed. These are the equivalences between the text names and math-mode names for the various accents:

text	math	example
circumflex	hat	$\hat{a}$
grave	grave	$\grave{a}$
acute	acute	$\acute{a}$
umlaut	ddot	$\ddot{a}$
tilde	tilde	$\tilde{a}$
dot	dot	$\dot{a}$
breve	breve	$\breve{a}$
caron	check	$\check{a}$
macron	bar	$\bar{a}$
—	vec	$\vec{a}$

5.2 Brackets and decorations

There are several brackets available through LyX. For most purposes, using just the keys `[{]}()|\<>` should suffice, but the effect, especially if you want to surround a large structure, such as a matrix or a fraction, or if you have several layers of brackets, is better using the Math Panel's Delimiter popup [see sec. 5.1.7]. For example, that's how you would construct the brackets around a standard matrix such as :

$$\left[\begin{array}{cc}1 & 2\\ 3 & 4\end{array}\right],$$

and to make it easier to see the layers of parentheses of an abomination such as:

$$\frac{1}{\left(1+\left(\frac{1}{1+\left(\frac{1}{1+x}\right)}\right)\right)}$$

or:

$$f\left(g\left(h\left(k\left(l\left(x\right)\right)\right)\right)\right).$$

The parentheses, and other brackets, from that menu will automatically re-size to accommodate the size of what is inside (This is done in straight LaTeX by `\left( blah \right)`).

It is very easy to construct the braces you want to use. Click on the brace you want on the left side with the left mouse button, the right side with the right button, and place them in the document by clicking on the button displaying your choices. If you want one side to not have a bracket, use the blank button. It will appear in LyX with a dotted line, but nothing will print.

If you decide after the fact to place parentheses (or other math structure, like a square root, or other decoration) around some math structure, you can do that by highlighting (selecting) the structure that is to go inside the parentheses (that is done by holding the Shift key down and moving the cursor with the arrow keys, or selecting with the mouse). Then, choose the appropriate brackets for left and right, and click on Apply. The parentheses will be drawn around the selected structure.

5.3 Arrays and Multi-line Equations

Arrays, such as matrices, are easily entered in LyX. In the Math Panel there is a matrix icon, which will open a popup for you to choose the number of rows/columns. Here is an example:

$$\left(\begin{array}{ccc} 1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9 \end{array}\right).$$

The parentheses aren't automatic, but you can add them with the delimiter menu. Remember that you can add this after the fact, by highlighting the matrix inside Mathed (Position the mouse on one side of the matrix, hold the Shift key down, and hit the appropriate arrow key to move the cursor across the matrix). You can, when you construct the matrix, decide whether the columns (or some of them) will be left-, right-, or center-justified. That is that panel which appears as `ccc` by default. You should be careful about determining how many rows and columns you need. You can add more rows to an existing matrix by hitting C-Enter while in the matrix, and you can delete rows with M-e k, but you can't add or delete columns, yet. Here's another example, with different alignments on the three columns:

$$\begin{array}{lcr} this & this\,column & this\,column\\ column & has & has\,right\\ has\,left\,alignment & center\,alignment & alignment \end{array}.$$

There are numerous other arrays used in LaTeX math-mode, particularly with the AMS-LaTeX packages included, such as `\cases` and commutative diagrams. These are not yet supported in LyX (one such array, the `\cases` environment, is now supported in development versions of LyX). However, it is possible to build these structures from a matrix (which is exactly what the `\cases` macro is, anyway). Here is a simple example of how LyX can display cases:

$$f(x):=\left\{ \begin{array}{ll} \frac{1}{q}, & \textrm{if }x=\frac{p}{q}\textrm{ (in lowest terms)}\\ 0, & \textrm{if }x\textrm{ is irrational} \end{array}\right.$$

This was constructed by first choosing delimiters, where the one on the right is a blank delimiter, then choosing inside the delimiters a 2×2 matrix, with both columns left-justified. The Roman text setting will be explained later.

Multi-line equations are very easy to construct in LyX. The display-math mode of Mathed will automatically switch to an `eqnarray` format (LaTeX's multi-line displayed equation format) if, in the displayed-equation box, you hit C-Enter. The best way to do this, if you decide you want a multi-line displayed equation, is to insert a new line (with C-Enter) immediately. Each line then has three regions, left, center, and right, which you can move through using either the arrow keys, the mouse, or the Tab key. Here is an example:

$$\begin{aligned} 3 &= 1+2 \\ 4+5 &= 9. \end{aligned}$$

You can also turn an existing displayed formula into a multi-line formula by hitting C-Enter while the cursor is anywhere on the original formula. However, LyX will *not* try to decide where to break the formula up into three parts, but places everything in the left side of the line. To change the alignment points of the equation, place the cursor where you want to start the middle part of the line, and hit C-Tab. It then puts everything to the right of the cursor in the middle region of the equation (which, by the way, is not typeset by LaTeX in display-math size, so you should not put large expressions like fractions there). Move to where you want the right side of the line to begin, and hit C-Tab again. The "extra" insertion points in the line will disappear.

You can delete a line of a multi-line formula by placing the cursor on the line and hitting M-e k. You create new lines with C-Enter.

5.4 Equation Numbering and Labels

Equation numbering is very easy in LyX. All it takes to change a displayed equation like:

$$1+2=3$$

into the numbered equation :

$$1+2=3 \tag{5.1}$$

is to go to the Insert menu, and select the Label... option. This opens a popup in which you must enter some string as the label. There is no need to call it by a specific number, since LaTeX will take care of re-numbering the equation. Labels will not appear as such on the final output. LaTeX will insert appropriate numbers for the equations. The labels are used internally for cross-referencing. You can turn on numbering without a specific label by entering `math-number` in the minibuffer (the line below the main LyX window, which usually displays information about the document. Click on it, and the mini-buffer clears, allowing you to enter commands such as these.) while the cursor is in the equation, such as:

$$1+1=2. \tag{5.2}$$

This behaves as a toggle. Entering `math-number` in the mini-buffer a second time removes numbering. You can reference a labelled (not just numbered) equation, (cf. (5.1)) by using the Cross-Reference popup, which you open using Insert->Cross-Reference... . Note that you should *not* use spaces in your labels, since they get deleted and cross-references will no longer work. This is a LyX bug, but one that can be dealt with.

For numbered (or labelled) multi-line formulas, the default is that all lines are numbered separately. One you attach a label to make the equation numbered, all subsequent lines receive a label of #. That label can be changed to another so that you can refer to that line, like (??) below.

$$\begin{aligned} 1 &= 3-2 && (5.3)\\ 2 &= 4-2 && (5.4)\\ 4 &\leq 7. && (5.5) \end{aligned}$$

You can turn off numbering of a specific line by entering `math-nonumber` in the minibuffer (below the main LyX window) while the cursor is on that line of a multi-line numbered equation. This also toggles. Though it may seem odd, entering `math-nonumber` again will turn the numbering back *on* for that line.

$$\begin{aligned} 1 &= 4-3 && (5.6)\\ 2 &= 7-5 && (5.7)\\ 1 &= e^{2\pi i} \\ 16 &\equiv 2\,(mod\,7) && (5.8) \end{aligned}$$

Note that the first equation in this set (5.6) is labelled, the next is numbered but unlabelled, the third is unnumbered, and the last (5.8) is again labelled.

5.5 User defined macros in math mode

LyX allows the user to define macros. An user defined macro in math mode is not a red colored name that LyX didn't know how to draw (formerly known as macro mode, now simply TeX mode). A macro definition box appears on screen as a grayed button with the name of the macro in blue (math color). But if you click on it, it will appear as a normal math box to allow you edit it. Just try it:

Now, to use this macro in other math boxes just type the name in TeX mode, in this case `\macro`, and it will be automatically expanded: $c = a + b$. As you can verify, the cursor can't go inside the macro, the whole macro is like a single character, and the TeX generated code of this expression is `c = \macro`.

However the cursor could go inside of some kind of macros, those that have *arguments*. In a macro definition box an argument looks like a `#` followed by the argument number:

Once expanded, this macro includes the usual empty rectangle to indicate that you can insert there whatever you want: `\macroarg =` $2+\sqrt{}$. Example: $b = 2 + \sqrt{x-2}$.

When exported to LaTeX, a macro definition will produce the command `\newcommand`.

5.5.1 Directions on using macros

5.5.1.1 How to create them

To create a macro definition box use this syntax in the minibuffer:

`math-macro <macro name> [number of arguments]`

Par example `\macro` was created with "`math-macro macro`", and `\macrowarg` was created with "`math-macro macrowarg 1`".

To insert an argument mark (only inside a macro definition box) use this:

`math-macro-arg <number>`

The argument mark in `\macrowarg` was introduced with "`math-macro-arg 1`".

You can use no more than 9 arguments, numbered from 1 to 9. An argument can be repeated inside the macro definition box, but of course can be edited only once. A number can't be greater than the number of arguments originally allowed for the macro.

5.5.1.2 How to navigate

With the arrow keys: Opening a macro form the left side will put the cursor in the first argument, to move to the second argument use the TAB key. Remember that pressing the Space bar will get the cursor out and at the right side of the macro.

With the mouse: As usual, click on the desired argument box. Sometimes this fails if the box is empty or too small.

Currently the user can only define command macros, but internally it's possible to define also environment macros.

There are several predefined macros, mainly to allow AMS-LaTeX users to use AMS macros in a WYSIWYM way, like `\cases` and `\binom`.[2]

5.6 Fine-Tuning

5.6.1 Typefaces

You can enter various typefaces in Mathed, but (at the moment) you have to use keyboard commands to set them. Not all appear exactly WYSIWYM, and a few don't appear WYSIWYM at all. The standard font for text is italic, *text*, but for numbers the standard is Roman. To set text in Roman font in math-mode, type M-c r once in math-mode text. To get bold, **text**, type M-c b in math-mode. To get calligraphic font for capital letters (which looks like Helvetica italics on-screen, but is fancier on the final output), $\mathcal{TEXT}$, type M-c i in math-mode. Here is a table with all the supported fonts:

[2] *LyX 1.0 note: `\cases` are currently disabled because after some simplification and improving of the macro stuff, the only valid arguments are subparagraphs (neither tabs nor newlines). Perhaps they will be again included in a later version. AAS*

Text-mode	Math	Mathed Keybinding
Bold	**Bold**	M-c b
Emphasize	$\mathcal{CALLIGRAPHIC}$	M-c e or M-c i
———	Roman	M-c r
`Typewriter`	`Typewriter`	M-c p
Default	$Default$	M-c Space

The keyboard command to switch to a particular text font is interpreted in math-mode as indicated. Math-mode does not support all characters in all fonts, and only letters will be supported with these font styles; only capital letters for calligraphic font.

For any of these fonts, you have to be careful how you enter the text. If there is text to the right of the entry point, the font reverts to that style after one character. To be able to type a string in a particular font, make sure there is a protected-space to the right of the cursor. Also, entering a protected-space will revert subsequent text to standard font.

It is possible (in AMS-LaTeX) to embolden (not italicize) numbers and special symbols in math-mode. However, LyX does not yet support this in WYSIWYM. It will print correctly, though. To get emboldened symbols, for example a bold α, enter `\boldsymbol{\alpha}` in Mathed. The close-brace appears (in red) automatically when you type the opening brace. This works for all symbols, as well as numbers. It is also possible to get Fraktur (traditional German) fonts by entering, say, `\mathfrak{g}` for a Fraktur g, or the "blackboard-bold" double-line capitals with `\mathbb{R}`. These symbols will not appear (yet) WYSIWYM in LyX, and the AMS-LaTeX packages (including `amsfonts`) must be used. Adding these packages is now a menu item, in the Layout -> Document... popup there is a radio button marked Use AMS Math. Once activated, all AMS-LaTeX symbols and environments are available.

We do not have examples of these fonts in this Guide so that the Guide can be previewed and printed without these packages, which some people would not have available.

5.6.2 Font Sizes

There are four (relative) font sizes (or "styles") used in math-mode, which are automatically chosen in most situations. These are called *textstyle*, *displaystyle*, *scriptstyle*, and *scriptscriptstyle*. For most characters, textstyle and displaystyle are actually the same size, but fractions, superscripts and subscripts, and certain other effects, are set larger or placed differently in displaystyle. Except for some operators, which re-size themselves to accommodate various situations, all text will be set if these various sizes as LaTeX thinks is appropriate. These choices can be over-ridden by setting the `math-size` in the minibuffer. For example, you can set $\frac{1}{2}$ normally (textstyle), or you can make it larger, which also changes the linespacing, in displaystyle, by entering `math-size displaystyle` in the minibuffer while the cursor is in the main line of the math-inset, $\dfrac{1}{2}$. Careful, though, if the cursor is on the denominator of that fraction, only the numerator will be enlarged,$\frac{1}{2}$! This reflects a LaTeX "unintended feature"[3], not a LyX one. These font-size changes are not as apparent in LyX as they are in the output. Here are some text in the various styles: $\displaystyle displaystyle$, $\textstyle textstyle$, $\scriptstyle scriptstyle$, $\scriptscriptstyle scriptscriptstyle$.

All these math-mode font sizes are relative, that is, if the whole math inset and surrounding text are set in a particular size, all these sizes will be adjusted. Similarly, if the base font size of the document is changed, all fonts will be adjusted to correspond.

Here is a paragraph in "largest" font, with symbols: α.

This applies to math-fonts in Titles, *etc.* as well.

[3]That is, a bug.

Chapter 6

More Tools

6.1 Cross-References

Those of you reading this manual online will see a grey box with text in it, right before the beginning of this sentence. This is a Label. Properly speaking, it is one half of a cross-reference. The other half is the Reference proper, and it looks like this: 6.1. Again, those of you reading the manual online will see a red box with red text in it. Those reading printed versions, however, will see a number — in this case, the number of this section. There's also another variety of cross-reference: 99. This is the page number containing the location of the label. That's what cross-references do: they let you reference other parts of your document. You don't need to remember which section number was what anymore — LyX will do that for you! All you need to do is use a Label to mark a section, figure, table, equation, etc., and then refer to it via a Reference.

To insert a label, use Insert -> Label. A box will appear where you can enter your label. You can't change labels once you've created them [at least, not yet]. You'd have to delete the old label, make a new one, then remove all of the old references to the now-defunct label and insert new references. Yes, it's a pain, which is why you should choose label names carefully.

To insert a reference, select Insert -> Cross Reference. The Insert Cross-Reference popup appears with a list of labels. Double-clicking on a list item inserts a reference into the text, as does the two different Insert Reference buttons. Lastly, the Update button is there in case you decide to leave this popup open and add labels in the meantime.

Note that if you cut & paste text from another document that contains a Label or Reference, or if you delete a label in your text, LaTeX will complain:

```
LaTeX Warning: Reference 'X' on page Y undefined on input line Z.
LaTeX Warning: There were undefined references.
```

You'll also see two question marks in the output instead of the reference.

There are a few more comments we need to make about the Labels. They always print the number of the section heading closest to them. So — if you want to put a label on a Chapter, but a Section heading immediately follows it, you need to put the Label *into* the Chapter environment. It doesn't matter where, and it will look weird on the LyX screen. However, you need to do this if you want to label the Chapter separately from the Section. The same goes for all other section headings. If plain text follows a section heading you want to label, then and only then can you put the Label in the text without worrying. You have been warned.

Also, a Label *only* makes sense in *numbered* section headings and table and figure floats. Bare figures and tables aren't numbered, so, like unnumbered section headings, you can't really use a Label on it.[1] See sections 4.3.1.1, 4.4.5, and 5.4 for details on using a Label with figures, tables, and equations, respectively.

6.2 Manual Fine-Tuning

6.2.1 Extra Horizontal Space

HFills are a special LyX feature for adding extra space in a uniform fashion. An HFill is actually a variable length space, whose length always equals the remaining space between the left and right margins. If there is more than one HFill on a line, they divide the available space equally between themselves.

Note: if you put an HFill at the beginning of a line, and it's *not* the first line in a paragraph, LyX ignores it. This prevents HFills from accidentally being wrapped onto a new line.

HFills can be inserted with Insert->Special Character->HFill. Here a few examples what you can do with them:

> This is on the left side   This is on the right
>
> Left   Middle   Right
>
> Left   1/3 Left   Right

That was an example in the Quote environment. Here: :is one in a standard paragraph. It may or may not be apparent in the printed text, but it *is* sitting in-between the two ":".

Remember that we said that an HFill always fills the remaining space between the margins? There may be more than one set of margins on a line. Here's an example with the List environment.

one   two   :three   four   five   six

[1] Well, you *can*, but only if you use the Page number reference. The regular Reference — the one that refers to a section/table/figure number — won't work, because there's no numbered thingy to refer to! You could also use bare-Labels as page markers, then refer back to them using the Page number reference. Once again, the regular Reference won't work very well. It will refer to something, but that something will typically be the number of the previous numbered section heading.

The ":" marks the beginning of the item. [There is actually a "hidden" HFill inside of the label of the List environment; it's put at the end of the label automagically.] HFills work similarly in other "multimargin" situations, like two-column mode.

6.2.2 Extra Vertical Space

To add extra vertical space above or below a paragraph, use Layout->Paragraph to open the Paragraph Layout popup.

We're not going to provide an example of a VFill, as it's a waste of paper. They work the same as any other type of filler, including HFills: they fill the remaining vertical space on a page with blank space. If there are several VFills on a page, they divide the remaining vertical space equally between themselves. You can therefore use VFills to center text on a page, or even place text 2/3 down a page, or 1/4, and so on.

6.2.3 Changing Paragraph Alignment

You can also change the paragraph alignment with the Paragraph Layout popup. There are four possibilities:

- Block
- Left
- Right
- Center

The default in most cases is block alignment, in which the inter-word spacing is variable and each line of a paragraph fills the region between the left and right margins. The other three alignments should be self-explanatory, and look like this:

This paragraph is right aligned,
this one is centered,
this one is left aligned.

In some paragraph environments, the default is something other than block alignment.

6.2.4 Forcing Page Breaks

If you don't like the way LaTeX does the page breaks in your document, you can force a pagebreak where you want one. In general, this will *not* be necessary because LaTeX is good at pagebreaking, as was already mentioned in section 3.7.4.

So in general there is no need to use the option described below, and we *strongly* recommend *not* to use it until the text is *really* finished, and until you have checked in the preview to see if you *really* have to change the pagebreaking.

Only in those cases, you can force a pagebreak above or below a paragraph in the Paragraph Layout popup by selecting Pagebreak-Above or Below.

You might want to use a pagebreak to ensure that a figure or table appears at the top of a page. This is, of course, the wrong way to do it. LyX gives you a way of automatically ensuring that your figures and tables appear at the top of a page [or the bottom, or on their own page] without having to worry about what precedes or follows your figure or table. See sections 4.3 and 4.4 and read about Floats to learn more.

6.2.5 Other Tools

This section describes briefly how some features well known by the LaTeX user can be generated with LyX.

- The protected blank: It is used to tell LyX (and LaTeX) not to break the line at that point. This may be necessary to avoid unlucky linebreaks, like in:

  > A good documentation should weight no more than 1
  > kg.

  Obviously, it would be a good thing to put a protected blank between "1" and "kg". A protected blank is set with Insert->Special Character->Protected Blank or with C-Space.

- The linebreak: You can force linebreaks within a paragraph by selecting Insert->Special Character->Linebreak or with C-Return. You should, however, not use this to correct LaTeX's linebreaking, as LaTeX is *very* good at linebreaking... (see section 6.2.4). There are, however, a number of situations where it is necessary to actively set a linebreak, e.g. in a poem or for an Address (see sections 3.3.5.1, 3.3.5.2 and 3.3.7.2).

- Horizontal lines above or below a paragraph: These can be useful for headlines. You get them in the Paragraph Layout popup. Here:

  ---

  is one.

6.3 Spellchecking

LyX itself has no built-in spell checker. Rather it uses the external `ispell` program as a backend[2] or the newer and supposedly better `aspell`[3].

For installation notes and where to find dictionaries, see the `ispell` documentation included with the `ispell` distribution. After installing `ispell` you can use the menu entry Spellchecker... from the Edit menu (make sure that the

[2] the latest version can be found at the ispell WWW page `http://ficus-www.cs.ucla.edu/ficus-members/geoff/ispell.html`

[3] `http://metalab.unc.edu/kevina/aspell/`

`ispell` binary is somewhere in your `PATH`. If you have version `4.*` of `ispell`, it may be a good idea to replace it with the current version, `3.1.20`. Version 4 was a short-lived GNU release of ispell, and had numerous problems.)

To start spellchecking, select the item Spellchecker... from the Edit menu. Checking will start just after the current cursor position. A popup window will appear showing any incorrect (or unknown) word found by `ispell` in the first line and allowing you to edit and replace it in a second line. Whenever `ispell` reports an unknown word, the word is highlighted and the view in your text buffer is updated to make the word visible. In the Spellchecker popup, there is also a box showing "near misses", if `ispell` can find any. Clicking to one of the near misses will copy the near miss into the replace input field (double-click to invoke replace). The buttons in the Spellchecker popup are quite self-explanatory:

Start spellchecking begins to verify the spellings of words, beginning at the cursor position in the document.

Insert in personal dictionary inserts the word in an ispell dictionary such that the word is recognized forever and forever.

Accept word in this session accepts the word rejected by `ispell` for the rest of this spellchecking session, but does not add the word to your personal dictionary.

Ignore word does not change the word caught by `ispell`, but it will catch the word again if it appears later in the document.

Replace word replaces the word by what you have typed to the replace input field.

Stop spellchecking stops the operation of `ispell`, but does not close the spellchecker popup. Note that you can cancel spellchecking at any time, not only after an incorrect word has been found.

Spellchecker Options... opens up another popup that allows you to alter the options fed to `ispell`.

Close Spellchecker does what you might expect.

Finally, there is a slider informing you about the progress of spell checking.

Note: If you want to spellcheck footnotes, margin notes, or captions in table- and figure-floats, you *must* open them first. This is easy to do with the Edit->Floats & Insets->Open All commands.

6.3.1 Spellchecker Options

The following options can be set in the Spellchecker Options popup (which you can also access from the Options menu):

6.3.1.1 Dictionary

By default, the dictionary file to use is determined by the language of the document you're checking, which is set in the Layout->Document popup. If you do not have a dictionary for the document language, `ispell` will exit immediately after invocation. There will be a message on your stderr saying something like:

```
Can't find file /usr/lib/ispell/xyz.hash
```

and a popup window will inform you about the failure. In this case, you can specify another dictionary file in the Spellchecker Options popup, by typing in the name in the space provided, and checking the Use alternate language... box, or change the Language option in the Document Layout popup (opened using Layout->Document).

A different solution is to make a link from say `deutsch.(aff|hash)` to `german.(aff|hash)` or whatever applies to your language. This is because these ispell-files normally have the native language name ("deutsch") whereas `ispell`, when started from lyx, searches for the English version of the name used with the LaTeX babel package ("german").

You may also have problems if the font encoding is not correct for that dictionary. If you use a language with latin1 encoding and set the Encoding option in the Document Layout popup to latin1 (or other than default), you must have this option in your language-dictionary as well. If your dictionary doesn't support the Encoding you chose, you'll have an error like this on stderr:
`ispell: unrecognized formatter type 'latin1'`

The spellchecker gives you an error that it couldn't start the `ispell` process and that you probably have some problems with your dictionary file.

There are four solutions to this problem. The easiest is to try the Input Encoding switch to ispell option. If that does not help, you can set Encoding to default when calling the spellchecker (which is probably annoying). The third is to add the `latin1` option to your dictionary *<language>*`.aff` file and recompile the dictionary (which probably isn't easy if you installed the whole stuff with some distribution and don't have the language directory of the `ispell` sources). Read the `ispell` documentation for this task! The fourth is to send a message to your package-maintainer, or better yet to the maintainer of the dictionary file in question and ask him to solve your problem.

6.3.1.2 Personal dictionary

`ispell` uses a personal dictionary file where words not found in the main dictionary can be saved so you don't get the same words reported every time you check a document. The name of this file is usually formed as `$HOME/.ispell_`*dictionary*, where *dictionary* is the name of the global dictionary used. Often the file (for the default dictionary) is simply `$HOME/.ispell_words`. If you want to use another file as your personal dictionary, you can also set this in the Spellchecker Options popup. Note that the directory in which `ispell` will put your file is your home directory, not the directory the file you're editing is in! Specifying a filename which does not already exist will result in an error message on stderr which you can ignore (`ispell` will create the file at the end of your spell checking).

6.3.1.3 Further Options

The Spellchecker Options popup has some additional options which are fairly self-explanatory:

- Treat run-together words as legal
  Keeps ispell from reporting compound words as misspelled.
- Use alternate personal dictionary
  You can give a filename to a personal dictionary here.
- Extra characters allowed in words
  Allows you to add nonstandard characters to what `ispell` considers words, e.g. German umlauts. This should not normally be needed.

6.3.2 Problems

If you can't spellcheck international characters, try the Input Encoding switch to ispell option.

6.3.3 Limitations

Even though your `ispell` program may correctly check contractions, it doesn't seem to when used through LyX. In addition, it is not at all clear that the options to allow runtogether words, or to include extra characters, or to allow extra affixes, work as they are supposed to.

Some users have expressed a wish to be able to globally change the spelling of a particular word, rather than having to change the spelling separately for each occurrence of the word. That feature will be considered for later versions of LyX.

6.4 International Support

This section describes how to use LyX with any language you want. LyX comes with a default configuration which supports the English language on a U.S.-style keyboard, with a standard U.S. paper size and the spell checker set to U.S. English. You can change any or all of these settings as desired, and you can make the changes apply to the current session only, or use them as your new default configuration.

If you have a keyboard suited to the language you are using (for example, a german keyboard for writing in German), and you have correctly configured your X environment, all you need to do for LyX is tell it your language, the character encoding, and desired paper size. Refer to 6.4.1 for more information.

If, however, you have a U.S.-style keyboard and want to write in a different language than English, you can use an alternate keymap. For example, if you have a U.S.-style keyboard but want to write in Italian, you should configure LyX to use an Italian keymap. Refer to 6.4.2 for details.

Finally, you may just want to change a few key mappings or create an entirely different keymap (for Vulcan, for instance). You may, for example, normally write in Italian on a U.S. keyboard but want to include an occasional quotation in German. In such a case, you can write your own keyboard mapping or modify an existing one to support the characters you want.

The details of how to customize LyX to your own language is *way* beyond the scope of this manual. You can not only alter the keyboard layout, you can also change the names of the menus buttons, etc., to reflect your language. If you want to learn more about writing keymap files and tailoring LyX to your native tongue, please see the *Customization* manual for details.

6.4.1 Layout->Document Language Options

The Layout->Document menu lets you set the language, papersize, and character encoding for your language.

Choose your language by clicking on the arrow in the Language box of the Layout->Document menu. The default is U.S. English. Scroll to find the language you want and then click on your choice. The language name appears in the window.[4]

Click on the Papersize popup to see your choices in paper size. The default is the U.S. Letter paper size. Click on the one you want and it appears in the window.

The Encoding box lets you choose the character encoding map you want to use. The default is the `ASCII` encoding which is typically sufficient for U.S. English. A superset of the `ASCII` encoding is the `Latin1` encoding, which includes the characters required by the various Western European languages. The third choice, `Latin2`, is for support of Eastern European languages. Click on the popup and then click on the encoding you want to use, and it appears in the window. (Refer to 6.4.3 for the character encodings.)

To use any language, papersize, or encoding change you made, click on the OK button. Your new configuration will now be used as long as you are in the current LyX session.

6.4.2 The Options->Keyboard Keymap Configuration

The Options->Keyboard menu allows you to choose the keymap of your choice for your U.S.-style keyboard. You can choose primary and secondary keyboard languages and then select which one you want to use.

Click on the down arrow for Primary and choose the language keymap you want by clicking on it. The name them appears in the Primary window. Do the

[4] In LaTeX terms, selecting a language other than default adds Babel support. If you do not have Babel installed, refer to the different LaTeX distributions for it.

same in the Secondary window for a secondary language if you want one. You can then select either your primary or secondary keymap in the Mapping section of the menu, or select No key mapping if you do not want to use an alternate keymap.

The Character set window allows you to use different character sets if your language uses more than one. Greek, for example, uses two, and a Greek user can enter `iso-8859-7` in this window and the appropriate character map (a `.cdef` file), if available, is loaded.

Note that one of the choices for both primary and secondary keymaps is Other. You can use this to select a custom keymap which you've created yourself. For example, current distributions of LyX provide an `american-2` keymap file in the `$LYX_DIR/kbd` directory. This keymapping supports some accented characters for other languages in addition to the U.S. keymap. To use the `american-2` keymap, select the Options-Keyboard menu, select other in the primary selection box, enter the name of the keymap (`american-2`) and click on OK. You should now be able to enter accent characters using the new keymap.

6.4.3 Character Tables

Here is a table with all the characters in the `Latin1` character set. You should be able to print all these characters directly from the keyboard without using too many modifier keys (if your keyboard is set up correctly, that is). Note that you must set your font encoding in Layout->Document->Encoding for latin1 to use this keyset, and for latin2 to use the `Latin2` keyset.

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
00				0	@	P	’	p				°	À	Đ	à	∂
01			!	1	A	Q	a	q			¡	±	Á	Ñ	á	ñ
02			“	2	B	R	b	r				²	Â	Ò	â	ò
03			#	3	C	S	c	s			£	³	Ã	Ó	ã	ó
04			$	4	D	T	d	t				´	Ä	Ô	ä	ô
05			%	5	E	U	e	u			¥	μ	Å	Õ	å	õ
06			&	6	F	V	f	v				¶		Ö		ö
07			‘	7	G	W	g	w			§	·	Ç	×	ç	÷
08			(	8	H	X	h	x			¨	¸	È		è	
09			)	9	I	Y	i	y			©	¹	É	Ù	é	ù
0A			*	:	J	Z	j	z			ª	º	Ê	Ú	ê	ú
0B			+	;	K	[	k	{			«	»	Ë	Û	ë	û
0C			,	<	L	\	l	|			¬	¼	Ì	Ü	ì	ü
0D			-	=	M	]	m	}			–	½	Í	Ý	í	ý
0E			.	>	N	ˆ	n	˜			®	¾	Î	Þ	î	þ
0F			/	?	O	_	o				¯	¿	Ï	ß	ï	ÿ

There are a few things you need to know about this table. This manual is set up — by hand, mind you — to print all of these characters. That ain't the default. Nowhere near, in fact. Here are some of the detail you'll need to bear in mind when using characters from the `Latin1` character set:

- The characters at entries A2, A4, A5, A6 and AD – the cent, the yen, the generic-currency-symbol, the broken vertical bar, and the short dash – are just plain missing in the default encodings. We don't know where they are or why this is the case.

- Even if you've selected latin1 in the Document Layout popup, users who have only the `OT1`-fonts for LaTeX [or who have the `T1`-fonts but aren't using them] will still be missing a few characters: D0, F0, DE, FE, AB, and BB – the uppercase and lowercase eth and thorn, and the french quotes – won't show up.

- Users of `OT1`-fonts can, however, get the french quotes [characters AB and BB] if they include the either the package `umlaute.sty` or `german.sty` in their documents.[5]

[5]This only holds when you want to input these quotes by yourself. The automatic quote feature described in Section 3.7.2.2, will generate automatically LaTeX code adapted to available fonts and packages.

The following is a full list of all of the accented characters LyX can display directly. It includes not only the accented characters from the previous table, but also the characters from `ISO8859--2` through `4`.

- From `ISO8859--1`:

¨ Ä Ë Ï Ö Ü ä ë ï ö ü ÿ	diaeresis
ˆ Â Ê Î Ô Û â ê î ô û	circumflex
‘ À È Ì Ò Ù à è ì ò ù	grave
´ Á É Í Ó Ú Ý á é í ó ú ý	acute
˜ Ã Ñ Õ ã ñ õ	tilde
¸Çç	cedilla
¯	macron[6]

- From `ISO8859--2` through `4`:

ĤĴĥĵĈĜŜĉĝŝ	circumflex
ŚŹśźŔĹĆŃŕĺćń	acute
ĨĩŨũ	tilde
ŞşŢţŖĻĢŗļģŅĶņķ	cedilla[7]
ĒēĀĪŌŪāīōū	macron
ŐŰőű	hungarian umlaut

All the characters above are actively supported by TeX fonts. In addition TeX allows diacritical marks on almost all characters . This is also supported in LyX, by enabling Options->LaTeX->Allow accents on ALL chars. Also make sure you're using the `T1` font-encoding and have the package `umlaute.sty` with the definition file `iso.def` installed.

6.4.4 International Spellcheck Support

LyX uses the `ispell` spelling checker. You should configure `ispell` to work with your system if it does not already. To get the appropriate language dictionary, refer to the `Where` file included with the `ispell` package. Note that some dictionaries do not support the `Latin1` encoding. If you have selected Layout->Document->Encoding as `Latin1` with one of these dictionaries, the spellchecker will not work for some people. Refer to section 6.3 for more details about international spellchecking.

---

[6] The dead macron in usually not needed, as you will use a non–dead key for this instead. For example, S-M-minus, or if `.Xmodmap` is correct, S-M-macron.

[7] These characters might not look very nice on screen, but they will be just fine when run through LaTeX and printed.

Chapter 7

LyX and LaTeX

7.1 How LyX Uses LaTeX

This chapter is for both TeX-nicians and the LaTeX-curious. In it, we'll explain how LyX and LaTeX work together to produce printable output. This is the only place in any of the manuals where we assume you know something about LaTeX.

At one time, we called LyX a "WYSIWYM frontend to LaTeX," but that's no longer true. There are frontends to LaTeX out there. They are basically editors with the ability to run LaTeX and mark any errors in the file you're editing. Although LyX *is* an editor, and it *does* run LaTeX, and it also marks errors in the file, it also does much, much more. Thanks to the WYSIWYM concept,[1] you don't need LaTeX to use LyX effectively. LyX has also added a few extensions to LaTeX.[2] Try the following sometime: select Export->as LaTeX... from the File menu, then look at the preamble of the resulting `.tex` file. You'll notice a variety of new macros defined specifically by LyX.[3] This is all transparent to the user, however.

There are five commands that automatically invoke LaTeX. They are:

- View dvi
- View PostScript
- Update dvi
- Update Postscript

[1] What You See Is What You Mean

[2] The Team did this for parsing/coding reasons. The new macros allow LyX to more easily produce LaTeX code and ensure that the resulting code is correct. Nothing that you, the user, need worry about.

[3] A little bit of history: at one time, all of these macros were in a `lyx.sty` file. The Team has since decided that `.tex` files generated by LyX should be more portable. So, now the macros are in the preamble.

- Print...

All of these are in the File menu. They will only invoke LaTeX if the file has changed since the last time LaTeX was run.

When you run LaTeX on the file you're editing, LyX performs these steps:

1. Convert the document to LaTeX and save to a file with the extension `.tex` in place of `.lyx`.
2. Run LaTeX on the `.tex` file (maybe several times).
3. If there are any errors, mark them in the document.

If you've run LaTeX using View dvi, LyX then executes `xdvi` on the Dvi file. If you've used View PostScript or Print... , LyX performs two more steps:

- Run `dvips` to convert the Dvi file to PostScript®:
  - For View PostScript, the output file has the extension `.ps_tmp`
  - For Print... , the output file has the extension `.ps`, as expected.
- Execute `ghostview` or send the PostScript® file to the printer.

7.2 "Help! LyX generated an unreadable .tex file!"

Die-hard LaTeX users will scream and howl this into the night, then declare LyX useless, simply because they didn't RTFM.[4]

We're going to set the record straight. LyX produces two kinds of LaTeX files. One is human readable. The other is LyX readable. Every time LyX executes LaTeX, it produces a LaTeX file that it can easily scan for errors. The resulting `.tex` file is not human readable. Don't even try to read it. If you want a `.tex` file that you can send to a colleague, select Export->as LaTeX... from the File menu.

We'll say that again, specially separated for the particularly clue-impaired:

> If you want a `.tex` file that you can send to a colleague, select Export->as LaTeX... from the File menu.

[4]Note from John Weiss: I know a certain faculty member who actually did complain to me that LyX doesn't generate LaTeX files. Ain't it funny how, sometimes, faculty are the worst at reading instructions?

7.3 Translating LATEX files into LYX

You can import a LATEX file into LYX by using the File->Import->LATEX command in LYX. This will call a Perl script call `reLyX`—which will create a file `foo.lyx` from the file `foo.tex`—and then open that file. If the translation doesn't work, you can try calling `reLyX` from the command line[5], possibly using fancier options.

`reLyX` will translate most legal LATEX, but not everything. It will leave things it doesn't understand in TEX mode, so after translating a file with `reLyX`, you can look for red text and hand-edit it to look right.

`reLyX` has its own manpage, which you should read to find out about what LATEX isn't supported, bugs (and how to get around them), and how to use the various options.

`reLyX` is relatively new, and still under development. The bad news is, there are still bugs and some missing features. The good news is that these may have been fixed by the time you read this. New versions of `reLyX` (which can be installed without reinstalling LYX) will be made available at the developers' web site[6] as they are written.

If you can't get `reLyX` to work, or you just want to put a piece of LATEX code into a LYX file, see Section7.4.

7.4 Inserting LATEX Code into LYX Documents

This is a rather important point: You can always insert LATEX code into any LYX document. LYX simply cannot, and will probably never be able to, display every possible LATEX construction. If ever you need to insert LATEX commands into your LYX document, you have two options:

Inline: Mark a block of text as LATEX anywhere, anytime.

- There are three ways to access this command:
  - Menu: Layout->TEX Style
  - Toolbar Button: The button with the red word "TEX" on it.
  - Keybinding: M-c t

  It works just like a character style. You can toggle TEX Style on or off. When on, everything you type will be in TEX mode. You can also select a block of text and [for example] hit the TEX button to mark it as TEX code.
- Best used for brief or small bits of LATEX code.

  We've already encountered some of the uses of TEX Style in sections 3.7.2.1 and 3.7.3.

[5] When LYX is installed, a separate executable called `reLyX` will be placed in the same folder as the `lyx` executable (e.g., `/usr/local/bin/reLyX`). `reLyX` requires Perl (version 5.002 as of this writing).

[6] `http://www.devel.lyx.org`

LATEX Environment: We've already encountered this one, too, in section 3.3.9.4.

- The LATEX paragraph environment is for large blocks of LATEX code. Using it is pretty simple if you follow a guideline or two:
  - Break lines using C-Return. A plain Return will reset the paragraph environment to Standard.
  - Don't begin lines with a Space. LYX has a bug that gags on such lines. [I know, that means you can't indent your LATEX code to make it look pretty.]
  - You *CAN* nest the LATEX environment inside of others. For example, suppose you wanted to do the following:

    ```
    \begin{itemize}
    \item .....
    .
    .
    .
    %Your LaTeX Code to do something
    %
    .
    .
    .
    \end{itemize}
    ```

    You can just nest a LATEX environment inside of an Itemize environment to achieve the same thing.
- Here's an example of code in the LATEX paragraph environment. The code looks like this:

  ```
  \begin{tabular}{ll}
  \begin{minipage}{5cm}
  This is an example for a minipage environment. You
  can put nearly everything in it, even (non-floating)
  figures and tables.
  \end{minipage}
  &
  \begin{minipage}{5cm}
  \begin{verbatim}
  \begin{minipage}{5cm}
  This ...
  \end{minipage}
  \end{verbatim}
  \end{minipage}
  \end{tabular}
  ```

  What follows is the LATEX environment proper. Those of you reading the manual online will only see a bunch of funky text in red. Those reading a printed version of the manuals will see the actual results:

This is an example for a minipage environment. You can put nearly everything in it, even (non-floating) figures and tables.	`\begin{minipage}{5cm}`
	`This ...`
	`\end{minipage}`

In addition to these two methods, you can also create a separate file containing some complex LaTeX structure. You can then use either the TeX Style or the LaTeX paragraph environment to put the LaTeX command "`\input{myfile}`" into your document. LaTeX will then include the file "`myfile.tex`" [note the addition of the "`.tex`" suffix] into your document. We recommend that you only do this if you have a `.tex` file which you *know* works already. Otherwise, you'll have a big job tracking down LaTeX errors...

There are a few last notes to emphasize:

- Inside of LyX, LaTeX code appears *in red.*
- LyX *does not* check if your LaTeX code is correct.
- Beware reinventing the wheel.

That last note refers to two things. First, LyX does have quite a few features tucked into it, and more are coming. Be sure to check the manuals to make sure that LyX doesn't have such-and-such feature before you go off merrily coding LaTeX.[7] Second, there are numerous LaTeX packages out there to do all sorts of things, from labels to envelopes to fancy multipage tables. Check out a CTAN site for details.

If you do need to do some wild and fancy things within your document, be sure to check out a good LaTeX book for assistance. There are a number of them listed in the bibliography of this manual.

There are a number of LaTeX commands which have to be placed before the beginning of the actual text. They go into the preamble, and this is explained in the next section.

7.5 LyX and the LaTeX Preamble

7.5.1 About the LaTeX Preamble

If you already know LaTeX, there is no need to explain here what the preamble is good for. If you don't, the following will give you some ideas — we recommend again that you consult a LaTeX book for further information. In any case, you should read the points below, because they explain what you can do and what you don't need to do in the LaTeX preamble of a LyX document.

The LaTeX preamble comes at the very beginning of a document, *before* the text. It serves to:

[7] Note from John Weiss: I seem to do this an awful lot. Sat down and merrily began coding something to print out labels, only to learn that there were already 2 different LaTeX packages to do this. Worse yet - I had them already!

- declare the document class. LyX already does this for you.

  If you're a seasoned LaTeX-nicians, and you have some custom document class you want to use, check out the *Customization Manual* for info how to make LyX interface to it. Be sure to submit your efforts to the LyX Team for inclusion in future versions!

- declare the usage of packages. LaTeX packages provide special commands, which are only available within a document when the package has been declared in the preamble. For example, the package `indentfirst` forces all paragraphs to be indented. There are other packages for labels, envelopes, margins, etc.

- set counters, variables, lengths and widths. There are several LaTeX counters and variables which *must* be set globally from within the preamble in order to have the desired effect. [There are other variables which you can set and reset inside the document, too.] Margins are a good example of something which must be set in the preamble. Another example is the label format for lists. You can actually set these just about anywhere, but it's best to do it just once, inside the preamble.

- declare user defined commands [with `\newcommand` or `\renewcommand`], mostly abbreviations for LaTeX commands which appear very often inside a document. Although the preamble is a good place to declare such commands, they *can* be declared anywhere else [but *before* they are used for the first time, of course...]. This can be useful if there is a lot of raw LaTeX code in your document, which normally should not be the case.

LyX adds its own set of definitions to the preamble of the `.tex` file it produces. This makes LaTeX files generated by LyX portable.

7.5.2 Changing the Preamble

The commands which LyX adds to the preamble of a LaTeX file are fixed; you can't change them without patching LyX itself. You can, however, add your own stuff to the preamble. There are two ways to do this:

1. Select LaTeX preamble from the Layout menu. This opens an editing popup in which you can put your favorite commands.

   The editing behavior of this popup is toolkit-dependent, so don't expect the LyX keybindings to work in it.

2. Set this in your default template (see 2.3), so that it will be the default preamble for any file you create.

LyX adds anything in the LaTeX preamble popup to its own built-in preamble. Before adding your own declarations in the preamble, you should make sure that LyX doesn't already support what you want to do [remember what we said about reinventing the wheel?]. Also *make sure your preamble code is correct.* LyX doesn't check it.

7.5.3 Examples

Here are some examples of what you can add to a preamble, and what they do:

7.5.3.1 Example #1: Offsets

There are two variables under LaTeX that control page position: `\hoffset` and `\voffset`. Their names should be self-explanatory. These variables are useful if you think for a moment about computer labels. Sometimes, the size of a print medium and the area of the medium that you can actually print on aren't the same. This is where `\hoffset` and `\voffset` come in.

The default values for `\hoffset` and `\voffset` are both 0 pt., i. e. the page isn't shifted.

Unfortunately, some DVI drivers always seem to shift the page. We have no idea why, or why the sysadmin hasn't fixed such behavior. If you're using LyX on a system that you don't personally maintain, and your sysadmin is a doofus, `\hoffset` and `\voffset` can save the day. Suppose you're left and top margins are always 0.5 inches too big. You can add this to the preamble:

```
\setlength{\hoffset}{-0.5 in}
\setlength{\voffset}{-0.5 in}
```

... and your margins should now be correct.

7.5.3.2 Example #2: Labels

Speaking of labels, suppose you wanted to print out a bunch of address labels. There's a rather nice package, available at your nearest CTAN archive, for printing sheets of labels, called `labels.sty`. Now, your system may not have this package installed by default. We leave that up to you to check. You'll also want to read the documentation for it; we're not going to do that for you. Since this is an example, however, we'll give you an example of how you use this package.

First, make sure you're using the article document class. Next, you need to put the following in your preamble:

```
\usepackage{labels}
\LabelCols=3
\LabelRows=7
\LeftBorder=8mm
\RightBorder=8mm
\TopBorder=9mm
\BottomBorder=2mm
```

This sets things up for Avery® label sheets, stock #5360. You're now ready to print labels, but you'll need to insert LaTeX code, placing the commands `\begin{labels}` and `\end{labels}` around each label text. This and other special features of `labels.sty` are explained in its documentation.

Someday, someone may write a LyX layout file to support this package directly. Maybe that someone is you.

7.5.3.3 Example #3: Paragraph Indentation

Americans are trained to indent the first line of *every* paragraph. As with all of their other weird quirks, most Americans will whine and moan until they can have their way and indent the first line of all paragraphs.[8]

Of course, this behavior isn't standard typography. In books, you typically only indent the first line of a paragraph *if* it follows another one. The idea behind indenting the first line of a paragraph is to distinguish neighboring paragraphs from one another. If there is no previous paragraph, for example, it follows a figure, or is the first paragraph in a section, then there is no special indentation.

If you're a typical American, though, you don't care about such esoteric things; you want your indentation! Add this to the preamble:

```
\usepackage{indentfirst}
```

If your TeX distribution isn't a braindead one, you'll have this package, and all of your paragraphs will get the indentation you think they deserve.

7.5.3.4 Example #4: This Document

You can also check out the preamble of this document to get an idea of some of the advanced things you can do. You'll probably need to make the LaTeX preamble popup full-screen to see most of it. Also, there are more examples and an assortment of LaTeX "dirty tricks" given in the *Extended Features* manual.

7.6 LyX and LaTeX Errors

When LyX calls LaTeX, it tells LaTeX to blithely ignore any errors and keep going. It then uses the log-file from the LaTeX run to do a post-mortem. As we stated earlier in the chapter, LyX generates two kinds of `.tex` files, one of which it uses to locate errors in the document. If there was an error someplace, LyX will put a box with the word "Error" at the appropriate place in the document.[9] It will also display a message popup alerting you to the fact that there were errors.

You can navigate through the errors by using Go to Error in the Edit menu. You can "open" the error-boxes and view the error message LaTeX produced by clicking on it.

Some folks also like to look at the log file directly, accessible from Edit->View LaTeX log file. There are some fairly common error messages an warnings. We'll cover those here. You should look at a good LaTeX book for a complete listing.

[8] Note from John Weiss: This was written by an American — *me*! It's my perception of my fellow countrymen. Tough if you don't like it. Thpbpbpbpbpbpbpbp!

[9] LyX will occasionally misguess where the error was. This will typically happen with tables, figures, math, and the preamble.

- “`LaTeX Warning:`“

  Anything beginning with these word is a warning message for the purpose of “debugging” the LaTeX code itself. You’ll get messages like this if you added or changed cross-references or bibliography entries, in which case, LaTeX is trying to tell you that you need to make another run.

  You can by-and-large ignore these.

- “`LaTeX Font Warning:`”

  Another warning message, this time about fonts which LaTeX couldn’t find. The rest of the message will often say something about a replacement font that LaTeX used.

  You can safely ignore these.

- “`Overfull \hbox`”

  LaTeX absolutely *loves* to spew these out. They are warning you about lines that were too long and run past the right margin. Almost always, this is unnoticeable in the final output. Or, only one or two characters extend past the margin. LaTeX seems to generate at least one of these messages for just about any document you write.

  You can ignore these stupid messages. Your eyes will tell you if there’s a problem with something that’s too wide; just look at the output.

- “`Underfull \hbox`”

  Not quite as common as its cousin. LaTeX seems to like to print lines that are a bit too wide as opposed to ones that are a bit too narrow. We have no idea why.

  You can ignore these, too.

- “`Overfull \vbox`” and “`Underfull \vbox`”

  Warnings about troubles breaking the page. Once again, just look at the output. Your eyes will tell you where something has gone wrong.

- “ `LaTeX Error: File ‘Xxxx’ not found`”

  The file “Xxxx” isn’t installed on this system. This usually appears because some package your document needs isn’t installed. If you didn’t touch the preamble or didn’t use the `\usepackage{}` command, then one of the packages LyX tried to load is missing. Use Help->LaTeX Configuration, to get a list of packages that LyX knows about. This file is updated whenever you reconfigure LyX (using Option->Reconfigure) and tells you which packages have been detected and what they do.

  If you *did* use the `\usepackage{}` command, and the package in question isn’t installed, you’ll need to install it yourself.

- "`LaTeX Error: Unknown option`"

  Error messages beginning with this are trying to tell you that you specified a bad or undefined option to a package. Check the package's documentation.

- "`Undefined control sequence`"

  If you've inserted LATEX code into your document, but made a typo, you'll get one of these. You may have forgotten to load a package. In any case, this error message usually means that you used an undefined command.

There are other error and warning messages. Some are self-explanatory. These are usually LATEX messages. Others are downright cryptic. These are actually TEX error messages, and we really have *no clue* what they mean or how to decipher them.

There's a general sequence you should follow if you get error messages:

1. Look at the LATEX code you inserted for typos.

2. If there are no typos, check and see that you used the command(s) correctly.

3. If you get a bunch of error boxes piled up at the very top of the document, it means that there are errors in the preamble. Start debugging your preamble.

4. If you didn't add anything to the preamble and didn't add any LATEX code to the document, the first suspect is your LATEX distribution itself. Check for missing packages and install them.

5. Okay, so there are no missing packages. Did you use any of the fine-tuning options in LYX? Specifically, did you *misuse* any of them, like trying to manually insert lots of Protected Blanks, Linebreaks, or Pagebreaks? Did you try to kludge something together with these instead of using the appropriate paragraph environment?

6. All right, you didn't use any of the fine-tuning options, you played by the rules. Did you try to pull a fancy maneuver? Did you do something funky inside a table or an equation, like inserting a graphic into a table cell?

7. Did you go overboard with the nesting? LYX [currently] doesn't check to make sure you're in the limits for nesting environments. If you nested a bunch of environments to the 17^{th} level, that's the problem.

8. Okay, you didn't get any error messages, but your output looks whacked. If you have a table or figure that's too wide or long for the page, you need to:

   (a) rescale the figure so it fits.

(b) trim down the table so it fits.

If something else is wrong with the output, and you didn't try to pull anything fancy or kludge the fine-tuning options, we're not sure what's wrong.

If all this doesn't help — well, then *perhaps* you might have found a bug in LYX…

Chapter 8

Credits

The documentation is something of an exercise in stone-soup. There are lots of hands that have contributed something to the pot.

First, we need to give due credit to those who came before us. They gave us the base upon which the new manuals are built, and some continue to provide information:

- Matthias Ettrich wrote the original documentation, from which this manual is built, as well as the introduction to this manual [or the "LyX Manifesto," as some of us call it].
  - Lars Gullik Bjnnes wrote several minidocs, including some of the information about international support in LyX.
  - Ivan Schreter also wrote a minidoc about international support, specifically about international keyboard maps and customization.
  - Pascal André originally documented the LinuxDoc SGML interface.
  - Alejandro Aguilar Sierra originally documented math mode and provided the entries for the math functions in `Reference.lyx`
  - Special thanks to the LyX Team [1] for help and answers to questions.

Next, it's time to give credit to the "LyX Documentation Team," all of the people who helped rewrite the old documentation into the form it had after ver. 0.10:

- David Johnson:
  - Contributor to the FAQ and the old "`HowDoI-.lyx`" [now defunct].
  - General editing assistance.
  - Documentation of:
    - math mode

    * tables
    * spellchecking

  ...in `UserGuide.lyx`

- Rich Fields:
  - Primary contributor to `Reference.lyx`
  - Documentation of the basic LyX interface in `UserGuide.lyx`
- Paul Evans:
  - Maintainer of the FAQ and the old "`HowDoI-.lyx`" [now defunct].
  - Documentation of LinuxDoc in `UserGuide.lyx`
- Paul Russel:
  - Documentation of figures and imported graphics in `UserGuide.lyx`
- John Raithel:
  - Documentation of internationalization features in `UserGuide.lyx`
- Robin Socha:
  - Documentation of:
    * footnotes
    * margin notes
    * table of contents
    * cross-references

  ...in `UserGuide.lyx`
- Amir Karger
  - Primary contributor to `Tutorial.lyx`
- Matthias Zenker:
  - Documentation of
    * manual fine-tuning
    * using LaTeX from within LyX

  ...in `UserGuide.lyx`
- John Weiss:
  - General organization and format of the documents.

- Documentation of :
  - LyX setup
  - paragraph environments, document layout, nesting, typography notes, fonts

  ...with some minor changes to Matthias Ettrich's introduction section in `UserGuide.lyx`.
- Also responsible for Introduction in `Tutorial.lyx`
- Editor of the documents. [from 6/96-fall 1997]

After fall of 1997, the LyX Team as a whole took over maintenance of the documentation.

Bibliography

[1] The LyX source tree. File: CREDITS.

[2] Leslie Lamport. *LaTeX: A Document Preparation System.* Addison-Wesley, second edition, 1994

[3] Michel Goossens, Frank Mittelbach and Alexander Samarin: *The LaTeX Companion.* Addison-Wesley, 1994

[4] A Guide to LaTeX 2_ε, Kopka and Daly.

[5] Donald E. Knuth. *The TeXbook*