LyX Reference Manual

Contents

Chapter 1

Introduction

1.1 About this Document

This is the LyX Reference Manual.

If you've looked at some of the other documentation, you might be expecting this section to be identical to all of the other opening sections in the other documentation files. The Reference Manual isn't like the other documents, and requires a soemwhat different approach.

First, this file is huge. No, I mean *really* huge! It contains ... well, everything. To keep that from being too intimidating or overwhelming, we're going to begin with examples of the entries and an explanation of how this monster is organized.

1.2 Organization

Each chapter is grouped around a central theme or topic. Chapter 2 describes the user interface. Included are the tool bar, the various popups and which menu items and keys they are bound to. Chapter 4 lists all of the bindable functions. You'll need this information if you want to alter the key bindings or create your own tool bar. Lastly, chapter 3 is the Dreaded Miscellaneous! Basically, anything we couldn't figure out where to put ended up here.

The entries themselves are the subsections of each chapter. They are in alphabetical order for ease of access. The sections of each chapter have single purpose: to split up the chapter into more managable chunks. They simply group the entries by letter, so, for example, there will be a section labelled

"A-C," another called "S," and so on.

1.3 A Sample Entry

This is a sample entry to the Reference Manual. It should help readers figure out how to use this manual, and show developers how to create entries.

1.3.1 Example #1: Entries

Default Bindings:

Menu - Where in the menus you can find this thing. If the menu item just brings up a dialog box, put its name in as a *Description* list. Also put in where on the panel the item in bound. Put the names of menus, popup buttons, etc. in Sans Serif to make it stand out from any other text. Example:

MenuName -> SubMenuName -> ItemName

Ding a Ling Panel: ThisButton

Toolbar - Should describe the location on the bar and what the icon looks like. Keep it brief. Example:

Fourth button from the left.

Hand with six fingers.

Keyboard - There will be either one or two paragraphs here. They should just contain a key sequence, and the name of the binding file. Put the name of the binding file in `Typewriter` font, and the key sequence in Sans Serif to make it stand out. If no binding file name is present, then that binding is standard in all of the binding files. Example:

C-M-p in `bindfile.bind`.

M-9

Purpose: The purpose of this entry is to describe these entries.

Within an entry, the different parts are labelled "Default Bindings:", "Purpose", "Usage", "Examples", and "See Also". These are each paragraph headings. The "Menu-", "Toolbar", and "Keyboard" parts of "Default Bindings:" are subparagraph headings. None of these should be numbered.

In a real entry, this will tell you about the item and what it's for.

Usage: This entry will tell you how to use this item.

There won't always be one of these. In fact, any of the"standard" sections of an entry may be missing, especially if it's empty. For example, there are only 15 or so toolbar buttons, so most things won't have a default toolbar binding. Or, the author of an entry may prefer to put some of the "Usage" stuff under "Purpose" or "Examples".

Examples: This is an example example for an example entry.

Are you confused? Good. You came to the right place!

Unfortunately, this example entry has been kinda abstract. So, we'll give you a few more example entries.

See Also: Example #2; Example #3.

1.3.2 Example #2: Bogosify

Default Bindings:

Menu - Bogus -> MuHaHa

Toolbar - First button from the left.
Bogus ball.

Keyboard - C-F50 in `bogus.bind`.
C-W-Help in `PCbogus.bind`.

Purpose: This item puts extra bogosity in your documents by activating the bogofilter in BoGoTEX. Everyone needs a little bogosity in their writings. Otherwise, you risk boring your readers into a deep sleep. In fact, some science professors, scoffing at the usefulness of the BoGoTEX extensions, have written and even published articles in reputable journals that have been known to put readers into a coma! Don't believe me? Try reading just about any scientific paper!

Usage: Just select the text to bogosify with the mouse, then use the keyboard bindings, toolbar button, or menu item for this function.

Examples: Why, this entire entry is an example! It's totally bogus!

See Also: BogoMIPS

1.3.3 Example #3: Decat

Default Bindings:

Menu - Cuddly -> Feline to bring up the Meow dialog box.

Meow Panel: Decat button.

Toolbar - Button smack dab in the middle.
Kitty-cat.

Keyboard - W-c W-a W-t in `fuzzy.bind`.

Purpose: The decat function fixes "additions" made by overly-helpful paws.

We all know how much cats *love* computers, especially the laps that sit in front of them. However, a cat, being feline, must add its opinions to whatever you're typing. Or, your cat may simply decide to help you finish your work so you can get to more important matters, such as brushing and petting. The decat function is designed for just such moments.

Usage: Select the text to decat while holding down all five mouse buttons, then use the keyboard bindings, toolbar button, or menu item for this function.

Examples: A particular black and white cat once added the following:

> sfd4rcxy45bb q43 tfd t43revcx

...to a document. Run through decat, it became:

> Hey, bozo! Pay attention to me!

Another frisky kitty named Horatio added:

> ';; msdam m-009243m laasd;mlk

to some text. Decat changed it to:

> Can I beat up another dog?

[And if you believe that, I have a bridge in Brooklyn for sale....]

See Also: ASPCA; PETA; ECC; Any cat owner.

1.3.4 Template

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left (or right). Brief description of button icon.
Keyboard -	C-S-M-W-key in `file1.bind`. C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

Chapter 2

User Interface

Ed. Note: The chapter with a blow-by-blow description of what each popup does, which menu its bound to, and so on. -jw

2.1 A-E

2.2 F-M

2.3 N-P

2.4 Q-T

2.5 U-Z

Chapter 3

General Information

Ed. Note: This is otherwise known as The Dreaded Miscellaneous! It'll be a pain to maintain, since I'll probably be splitting it every few months! -jw

Chapter 4

Bindable Functions

We have some quick, last minute notes for you about this chapter.

If a particular piece of an entry is missing, it's equivalent to saying, "There is none." For example, if, under "**Default Bindings**" there was no "**Menu**" item, it means there is no menu binding. Likewise, the guy who wrote the entry may have decided that the function's purpose is self-explanatory. So, there is no "**Purpose**" section in the entry.

Hopefully, all of that is clear.

If you haven't noticed already, this is a *huge* chapter. Your best bet for getting around in it is to open up the Table of Contents browser. Select Table of Contents... from the Edit menu.

Lastly, some credits:

Alejandro Aguilar-Sierra wrote the math entries.

Rich Fields wrote most of the entries in here.

John Weiss edited this chapter.

4.1 A

4.1.1 accent-acute

Default Bindings: None.

Purpose: To produce accented characters from the ISO-Latin-1 character set.

All of the commands of the form "`accent-xxxxxx`," are of the same type. Rather than repeat the same entry 17 times, we'll explain how all of the "`accent-*`" commands work.

Usage: All of the "`accent-*`" commands work like a "Compose" key. They add a particular accent to the next character typed. There are two ways to use these commands:

1. Direct Execution:

   (a) Use the "`command-execute`" key to enter the desired "`accent-*`" command in the minibuffer.

   [See the entry for "`command-execute`" if you don't know how to execute LyX commands directly.]

   (b) Next, type the letter you wish to accent.

2. Dead Keys

   (a) Bind the desired "`accent-*`" command to a keyboard key. Typically, you bind the command to the key that looks like the accent.

   For example, you'd bind "`accent-acute`" to the apostrophe key (the ') and "`accent-circumflex`" to the "ˆ" key (Shift-6 on a US keyboard).

   [See the manual `Customization.lyx` for information about binding keys.][1]

   (b) Restart LyX if it's already running, so that the new bindings can take effect.

   (c) Type the dead-key, followed by the letter you wish to accent.

   Using our earlier example, if you type "ˆ" followed by the letter "e", you'll produce the character "".

To produce the accent by itself, hit the Space key instead of a character.

You can only accent a character if the result is a standard ISO-Latin-1 character. Otherwise the function will have no effect and LyX will beep in protest.

[1] *Editor's Note: I may change this and put an entry to the bind command in this manual. -jw*

Examples: This command produces the following accent:
Valid letters with an acute accent:
ś ź Ś Ź ŕ ĺ ć ń Ŕ Ĺ Ć Ń

See Also: command-execute;
the key binding section in `Customization.lyx` ;
accent-breve; accent-caron; accent-cedilla; accent-circle; accent-circumflex; accent-dot; accent-grave; accent-hungarian-umlaut; accent-macron; accent-special-caron; accent-tie; accent-tilde; accent-umlaut; accent-underbar; accent-underdot; accent-vector.

4.1.2 accent-breve

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a breve accent:
ă ğ ŭ Ă Ğ Ŭ

See Also: accent-acute.

4.1.3 accent-caron

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a caron accent:
č š ž Č Š Ž ď ľ ň ř ť Ď Ľ Ň Ř Ť ě Ě

See Also: accent-acute.

4.1.4 accent-cedilla

Default Bindings: None.

Examples: This command produces the following accent:
¸Valid letters with a cedilla accent:
ģ ķ ļ ņ ŗ ş ţ Ģ Ķ Ļ Ņ Ŗ Ş Ţ

See Also: accent-acute.

4.1.5 accent-circle

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a circle accent:
å ů Å Ů

See Also: accent-acute.

4.1.6 accent-circumflex

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a circumflex accent:
ĉ ĝ ĥ ĵŝ Ĉ Ĝ Ĥ Ĵ Ŝ

See Also: accent-acute.

4.1.7 accent-dot

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a dot accent:
ė iĖ İ ċ ġ ż Ċ Ġ Ż

See Also: accent-acute.

4.1.8 accent-grave

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a grave accent:

See Also: accent-acute.

4.1.9 accent-hungarian-umlaut

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a hungarian umlaut accent:
ő ű Ő Ű

See Also: accent-acute.

4.1.10 accent-macron

Default Bindings: None.

Examples: This command produces the following accent:
Valid letters with a macron accent:
ā ē ī̄ō ū Ā Ē Ī Ō Ū

See Also: accent-acute.

4.1.11 accent-special-caron

Default Bindings: None.

Examples: WARNING! THIS ACCENT COMMAND CURRENTLY CRASHES LyX! DO NOT USE!

This command produces the following accent:

...

Some letters with a special caron accent:

...

See Also: accent-acute.

4.1.12 accent-tie

Default Bindings: None.

Examples: This command produces the following accent:

Valid letters with a tie accent: None. Just the accent itself.

See Also: accent-acute.

4.1.13 accent-tilde

Default Bindings: None.

Examples: This command produces the following accent:

Valid letters with a tilde accent:

ĩ ũ Ĩ Ũ

See Also: accent-acute.

4.1.14 accent-umlaut

Default Bindings: None.

Examples: This command produces the following accent:

Some letters with an umlaut accent:

See Also: accent-acute.

4.1.15 accent-underbar

Default Bindings: None.

Examples: This command produces the following accent:
_ Some letters with an underbar accent: None. Just the accent itself.

See Also: accent-acute.

4.1.16 accent-underdot

Default Bindings: None.

Examples: This command produces the following accent:
. Valid letters with an underdot accent: None. Just the accent itself.

See Also: accent-acute.

4.1.17 accent-vector

Default Bindings: None.

Examples: WARNING! THIS ACCENT COMMAND CURRENTLY CRASHES LyX! DO NOT USE!

This command produces the following accent:

...

Some letters with a vector accent:

...

See Also: accent-acute.

4.2 B

4.2.1 backward-select

Default Bindings:

Keyboard - S-Left

Purpose: Highlights a single character to the left of the current cursor location. If the cursor is at the left screen margin the cursor moves to the end of the previous line, and the whitespace in-between is selected. The selected text will be in reverse-video.

See Also: char-backward;
forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.2.2 break-line

Default Bindings:

Menu - Insert -> Special Character -> Linebreak

Keyboard - C-Return

Purpose: To forcibly split the current line without ending the current paragraph.

Usage: This command has several purposes, depending on context. Its primary function is to manually end a line without ending the paragraph. This should only be done when you need some fine-tuning.

In math-mode, the `break-line` command automagically turns a one-line equation into a multiline expression, inserting the new equation line as well as all alignment delimiters.

See Also: break-paragraph; break-paragraph-keep-layout.

4.2.3 break-paragraph

Default Bindings:

Keyboard - Return

Purpose: To terminate the current paragraph.

Usage: This command does slightly different things, depending on context. Typically, it ends the current paragraph, breaks the current line, and starts a new paragraph in the Standard paragraph environment on the next line. It also typically removes all envrionment nesting.

If you are in the middle of a paragraph, `break-paragraph` splits it in two. If the paragraph is a list item, it creates a new list item in the process.

If you are at the end of a paragraph, `break-paragraph` starts the new paragraph after the current one. If you are at the beginning of a paragraph, the new paragraph starts *before* the current one.

`break-paragraph` may start the new paragraph in an environment other than Standard, but the behavior is not consistent. For example, if you move to the middle of an Itemize list, to the end of one of the items or the beginning of the next item, `break-paragraph` will not change the paragraph environment. It will still mess up the envrionment nesting, however.

Examples: None. Your best bet is to just play with hitting return until you build an intuition for how this works.

See Also: break-paragraph-keep-layout; break-line;
depth-decrement; depth-increment.

4.2.4 break-paragraph-keep-layout

Default Bindings:

Keyboard - M-Return

Purpose: To terminate the current paragraph without changing the paragraph environment or nesting depth.

Usage: Its usage is very much the same as `break-paragraph`, with one important difference. The new paragraph *always* has the same environment and nesting depth as the old one.

See Also: break-paragraph; break-line;
depth-decrement; depth-increment.

4.2.5 buffer-auto-save

Default Bindings:

Keyboard - C-x a or C-x C-a in `emacs.bind` only.

Purpose: Forces an autosave on the current document.

Usage: Backward compatibility. This function used to be the only way to undo any editing. With the advent of a true undo/redo function in LyX, forcibly autosaving a document doesn't have much use.

See Also: buffer-load-auto-save

4.2.6 buffer-begin

Default Bindings:

Keyboard - C-Home in `pc-modern.bind`.
Home in `emacs.bind`.

Purpose: Moves the cursor location to the top of the current buffer.

See Also: char-forward; char-backward; word-forward; word-backward; line-begin; line-end; tab-forward; up; down; screen-up; screen-down; buffer-end.

4.2.7 buffer-begin-select

Default Bindings:

Keyboard - S-C-Home

Purpose: Highlights text from the current cursor location to the top of the current buffer. The selected text is shown in reverse-video.

See Also: buffer-begin;
backward-select; forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-end-select.

4.2.8 buffer-close

Default Bindings:

Menu - File -> Close

Keyboard - M-f c (standard).
C-F4 in `pc-modern.bind`.
C-x k or C-x C-k in `emacs.bind`.

Purpose: Closes the current buffer. If changes have been made since it was first opened, or since the last `buffer-save` or `buffer-save-as` command, you will be asked if you want to save the buffer to its file before closing the buffer.

See Also: buffer-open.

4.2.9 buffer-end

Default Bindings:

Keyboard - C-End in `pc-modern.bind`.
End in `emacs.bind`

Purpose: Moves both the screen display and the current cursor location to the end of the current buffer.

See Also: char-forward; char-backward; word-forward; word-backward; line-begin; line-end; tab-forward; up; down; screen-up; screen-down; buffer-begin.

4.2.10 buffer-end-select

Default Bindings:

Keyboard - S-C-End

Purpose: Selects text from the current cursor location to the bottom of the current buffer. The selected text is shown in reverse-video.

See Also: buffer-end-select;
backward-select; forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select.

4.2.11 buffer-load-auto-save

Default Bindings: None

Purpose: Loads the autosave file.

Usage: To invoke this function, you must use the `command-execute` key to enter `buffer-load-autosave` in the minibuffer.

Invoking this command once loads the autosave file (which has the suffix "`.bak`") in place of the current document. Invoking it two times in a row loads the autosave backup file (which has the "`.bak~`" suffix) in place of the current document.

You cannot change the time interval between autosaves on the fly. The interval, in seconds, is set in the files `system.lyxrc` or `${HOME}/.lyxrc`.

See Also: command-execute;
buffer-auto-save.

4.2.12 buffer-new

Default Bindings:

Menu - File -> New

Keyboard - M-f n (standard).
C-x d or C-x C-d in emacs.bind.

Purpose: Opens a new buffer for a new LyX document.

Usage: This function first opens a File Browser window. You must now specify a new filename for the new document. You can use the File Browser to choose the directory for the new file, but the file name itself must be unique. The file cannot already exist.

After you enter the filename and choose Ok, the File Browser window opens again. This time, it has the title "Choose Template." If you wish to use a LyX template file for this new document, use the File Browser to choose one. Otherwise, hit the Cancel button.

See Also: File Browser;
LyX Template Files;
Editor's Note: These first two will be added at a later date. - jw
buffer-open; buffer-close; buffer-write; buffer-write-as.

4.2.13 buffer-open

Default Bindings:

Menu - File -> Open

Toolbar - First button from the left.
File folder with an arrow.

Keyboard - M-f o (standard).
F3 in pc-modern.bind.
C-x f or C-x C-f in emacs.bind.

Purpose: Opens an existing file and places it in a new buffer.

Usage: This function opens a File Browser window with the title "Document to Open." Choose the file to open and hit Ok.

See Also: File Browser; [*Editor's Note: To be added. - jw*]
buffer-open; buffer-close; buffer-write; buffer-write-as.

4.2.14 buffer-previous

Default Bindings:

Keyboard -	M-d p (standard). C-x b or C-x C-b in `emacs.bind`.

Purpose: Changes from the current document to another open and previously viewed document.

Usage: Self-explanatory.

If only one document is open, `buffer-previous` has no effect.

If more than two documents are open, `buffer-previous` switches to whichever document you were last working on.

You can also switch between open documents using the Documents menu.

Using `buffer-previous` several times in a row switches back and forth between two documents. This is useful for cutting and pasting bewteen two different documents.

See Also: Documents Menu. [*Ed. Note - to be added. -jw*]

4.2.15 buffer-print

Default Bindings:

Menu -	File -> Print
Toolbar -	Button #3 from the left. A stylized printer.
Keyboard -	M-f p

Purpose: Prints the current document.

Usage: This function opens the Print panel, which allows print control via buttons and typed inputs; a buffer can be printed either to a file or to a spooled print device. When printing to a printer, the printer name you specify in the Printer box must be configured for `dvips`.[2] Most systems have "`ps`" as their default PostScript printer, for which `dvips` comes preconfigured.

See Also: The section on dvips in `UserGuide.lyx`;
Print panel; [*Ed. Note - To be added. - jw.*]
buffer-view; buffer-view-ps.

4.2.16 buffer-typeset

Default Bindings:

Menu -	File -> Run LaTeX
Keyboard -	M-f t or M-f r (standard).
	C-x r, C-x t, C-x C-r, or C-x C-t in `emacs.bind`.

Purpose: Runs LaTeX on the current buffer.

Usage: A LaTeX scratch file is created, based on the contents of the current buffer, and the file is processed once with LaTeX. LaTeX messages are posted to the screen of the parent window, and approximate LaTeX error locations are marked within the current buffer with a red error box; the text of the actual LaTeX error can be displayed by double-clicking on the error-box. [Editor's Note: the M-f r keystroke combination is described as obsolete in documentation within the `emacs.bind` file-<ref>]

See Also: error-next.

4.2.17 buffer-view

Default Bindings:

Menu -	File -> View (xdvi)

[2]This is explained in greater detail in `UserGuide.lyx`.

Keyboard - M-f x (standard).
C-x p or C-x C-p in `emacs.bind`.

Purpose: Views the current buffer with the `xdvi` program.

Usage: A LaTeX scratch file is created and processed, and a `xdvi` process window is then automatically created with which you can view the resulting `dvi` file. If any of these programs are not installed on your system, the process stops at that point and control is returned to LyX.

Examples: None necessary.

See Also: buffer-print; buffer-view-ps.

4.2.18 buffer-view-ps

Default Bindings:

Menu - File -> View (ghostview)

Keyboard - M-f g (standard).
C-x g or C-x C-g in `emacs.bind`.

Purpose: Views the current buffer with the `ghostview` program.

Usage: A LaTeX scratch file is created and processed, and `dvips` is run on the resulting `dvi` file to create a scratch PostScript file. A `ghostview` process window is then automatically created in which you can view your work essentially in the same form as if it were printed. If any of these programs are not installed on your system, the process stops at that point and control is returned to LyX.

See Also: buffer-print; buffer-view.

4.2.19 buffer-write

Default Bindings:

Menu -	File -> Save
Keyboard -	M-f s (standard). F2 in `pc-modern.bind`. C-x s or C-x C-x in `emacs.bind`.

Purpose: Writes (saves) the current document to the original filename.

See Also: buffer-write-as.

4.2.20 buffer-write-as

Default Bindings:

Menu -	File -> Save As
Keyboard -	M-f a (standard). C-x w or C-x C-w in `emacs.bind`.

Purpose: Saves the current buffer to a different filename than that of the file originally opened.

Usage: This function opens an input window that allows you to specify a new filename via input to a window box. Currently (0.10.x), a file browser is not part of the window, and you may specify an existing filename.

See Also: buffer-write.

4.3 C-E

4.3.1 cancel

Default Bindings:

Keyboard -	Escape in `pc-modern.bind`. C-g in `emacs.bind`.

Purpose: General cancel. Cancels the selection of a meta-key selection, a drop-down menu, or a pop-up window. Cancels entry of commands in the minibuffer. Also leaves math-mode.

Usage: `Cancel` is the primary method of leaving math mode. Also, if a meta-key selection has been started, (for instance, M-c), pressing the `cancel` key will abort the selection. Similiarly, if a `find-replace` window is open, placing the cursor in the window and executing the `cancel` function will close the window, or if a drop-down menu has been selected, `cancel` will close the menu. Lastly, if you are entering a command in the minibuffer, `cancel` will abort entry and leave the minibuffer.

See Also: command-execute; find-replace; math-mode.

4.3.2 char-backward

Default Bindings:

Keyboard - Left (standard).
C-b in `emacs.bind`.

Purpose: Moves the cursor to the left by one character. At the beginning of the line the cursor will move to the end of the previous line.

See Also: char-forward; word-forward; word-backward; line-begin; line-end; tab-forward; up; down; paragraph-up; paragraph-down; screen-up; screen-down; buffer-begin; buffer-end.

4.3.3 char-forward

Default Bindings:

Keyboard - Right (standard).
C-f in `emacs.bind`.

Purpose: Moves the cursor to the right by one character. At the end of the line the cursor will move to the beginning of the next line.

See Also: char-backward; word-forward; word-backward; line-begin; line-end; tab-forward; up; down; paragraph-up; paragraph-down; screen-up; screen-down; buffer-begin; buffer-end.

4.3.4 command-execute

Default Bindings:

> **Keyboard -** M-x in `emacs.bind`.

Purpose: Executes a LyX function.

Usage: There are two ways to invoke this command. You can use the key binding, or you can click on the minibuffer with the mouse. [The minibuffer is the single line of text in the grey box below the editing area.]

After you invoke `command-execute`, a cursor will appear in the minibuffer. Type the name of a LyX function. If the function takes any arguments, you can type them in after the function name, using a space to separate the two. Then hit "Return."

Examples:

- Example #1:

  Click on the minibuffer with the mouse. Type "accent-acute" followed by the "Return" key. Now type the letter "u". You should get the character "".

- Example #2: Commands with Arguments:

  Type "M-x". [`emacs.bind` users only. Those using `pc-modern.bind` should click on the minibuffer with the mouse.] Now type "layout LyX-Code" followed by the "Return" key. The current paragraph will now be in the LyX-Code environment. You could have specified any other valid paragraph environment as an argument to the `layout` command.

See Also: accent-acute; layout.

4.3.5 command-prefix

Default Bindings:

Menu -	?
Toolbar -	?
Keyboard -	?

Purpose: ?

Usage: ?

Examples: ?

See Also: ?

4.3.6 copy

Default Bindings:

Menu -	Edit -> Copy
Toolbar -	Fifth button from the left. Two pages with a + sign.
Keyboard -	M-e o (standard). C-c or C-Insert in `pc-modern.bind`. M-w in `emacs.bind`.

Purpose: Places a copy of the previously selected text into the paste buffer.

Usage: Select text, either with the mouse cursor or by motion selection commands, then either execute `cut` by either following the menu commands or pressing the appropriate keystrokes.

See Also: cut; paste.

4.3.7 cut

Default Bindings:

Menu -	Edit -> Cut
Toolbar -	Sixth button from the left. An open pair of scissors.
Keyboard -	M-e c (standard). C-x or S-Delete in `pc-modern.bind`. C-w in `emacs.bind`.

Purpose: Removes previously selected text and places the removed text into the copy paste buffer.

Usage: Select text to be removed, either with the mouse cursor or by motion selection commands, then execute `copy` by either following the menu commands, selecting the toolbar icon, or pressing the appropriate keystrokes.

See Also: copy; paste.

4.3.8 delete-backward

Default Bindings:

Keyboard -	Backspace .

Purpose: Deletes the character to the left of the cursor.

Usage: Self-explanatory. If the cursor is at the beginning of a paragraph, two things may happen. First, if the previous and current paragraph have the same type of paragraph environment, LyX joins the two paragraphs. If the two have different environment types, LyX will most likely not[3] join them and simply move the cursor to the end of the preceding line.

[3] "most likely not" means that the action depends on the specific paragraph environments involved. Play around and see what happens.

See Also: delete-forward; word-delete-forward; word-delete-backward; line-delete-forward; line-delete-backward.

4.3.9 delete-forward

Default Bindings:

Keyboard - Delete (standard).
C-d in `emacs.bind`.

Purpose: Deletes the character to the right of the cursor.

Usage: Self-explanatory. If the cursor is at the end of a paragraph, two things may happen. First, if the next and current paragraph have the same type of paragraph environment, LyX joins the two paragraphs. If the two have different environment types, LyX will most likely not[4] join them and simply move the cursor to the end of the preceding line.

See Also: delete-backward; word-delete-forward; word-delete-backward; line-delete-forward; line-delete-backward.

4.3.10 depth-decrement

Default Bindings:

Keyboard - M-p Left (standard).
C-S-d in `pc-modern.bind`.

Purpose: To decrease the paragraph environment depth.
To un-nest two nested paragraph environments.

[4] "most likely not" means that the action depends on the specific paragraph environments involved. Play around and see what happens.

Usage: To change the depth of a single paragraph, simply move the cursor into that paragraph and invoke `depth-decrement`. If you want to change the depth of a group of paragraphs, select the group with the mouse and then invoke `depth-decrement`.

This command "wraps." Namely, if you reach the outermost environment depth, `depth-decrement` move the paragraph to the innermost environment depth.

Examples: Examples.

See Also: depth-next; depth-increment.

4.3.11 depth-increment

Default Bindings:

Menu -	Layout -> Change Environment Depth
Toolbar -	Fifth button from the right. Icon indicates doubly indented text bounded by arrows.
Keyboard -	M-p Right (standard). C-d in `pc-modern.bind`.

Purpose: To increase the paragraph environment depth.
To nest two paragraph environments.

Usage: To change the depth of a single paragraph, simply move the cursor into that paragraph and invoke `depth-increment`. If you want to change the depth of a group of paragraphs, select the group with the mouse and then invoke `depth-increment`.

This command "wraps." Namely, if you reach the innermost environment depth, `depth-increment` move the paragraph to the outermost environment depth.

Examples: Examples.

See Also: depth-decrement; depth-next.

4.3.12 depth-next

Default Bindings:

Keyboard - M-l v .

Purpose: This command is simply an alias for `depth-increment`.

See Also: depth-increment.

4.3.13 down

Default Bindings:

Keyboard - Down (standard).
C-n in `emacs.bind`.

Purpose: Moves the cursor down one line. If the cursor starting-position is at the bottom of the screen-display, the buffer scrolls upward to display the new current line about 1/4 of the screen-height from the bottom of the screen.

See Also: char-forward; char-backward; word-forward; word-backward; line-begin; line-end; tab-forward; up; screen-up; screen-down; buffer-begin; buffer-end.

4.3.14 down-select

Default Bindings:

Keyboard - S-Down

Purpose: To select text from the current cursor position to the same position one line down. If the next line is shorter than the current line, the cursor simply moves to the end of the next line.

See Also: down;

backward-select; forward-select; up-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.3.15 drop-layouts-choice

Default Bindings:

Keyboard - M-p c (standard).

Purpose: Opens up the combobox for selecting paragraph environments, which is to the left of the toolbar.

Usage: Once the combobox is open, you can use the cursor keys to select a new paragraph environment. Hitting "Return" chooses the paragraph environment currently highlighted. Hitting "Escape" cancels the operation and closes the combobox.

See Also: layout.

4.3.16 error-next

Default Bindings:

Menu - Edit -> Go to Error

Keyboard - M-e e (standard).
C-g in `pc-modern.bind`.

Purpose: Find the next LaTeX error in the buffer.

Usage: You can only use this command under two conditions:

1. You've recently invoked `buffer-typset`, `buffer-view`, `buffer-view-ps`, or `print` to run LaTeX on your document.
2. LaTeX found errors in the document.

If this happens, LyX will find the errors and mark them in the document with a box containing the word "Error".

Examples: It's not easy to provide an example for this one. If you don't know LaTeX, and never intend to insert any LaTeX code into your LyX documents, you might not need to worry about this command. It's here for those users who *do* know LaTeX and might enter incorrect LaTeX code into their documents.

See Also: buffer-typeset; buffer-view; buffer-view-ps; print.

4.4 F

4.4.1 figure-insert

Default Bindings:

Menu -	Insert -> Figure
Toolbar -	Second button from the right. A picture embedded in text.
Keyboard -	M-i g

Purpose: Insert graphics in the current document.

Usage: This function opens the Figure panel, which contains three options:

1. Encapsulated Postscript
2. Inlined EPS
3. LaTeX or TeX

You'll get slightly different behavior depending on which type you select.

In actuality, LyX only recognizes two graphics formats, LaTeX[5] graphics and ecapsulated PostScript files. The first two options in the Figure panel, "Encapsulated Postscript" and "Inlined EPS," import encapsulated PostScript. The other option, "LaTeX or TeX," obviously imports LaTeX graphics.

Placement is the next difference in the three options. "Inlined EPS" inserts the graphics within the current line, as if it were a normal character. This can

[5]This includes pure TeX graphics commands.

have remarkably bizarre results on the line spacing if the imported graphics is very tall. The other two options, "Encapsulated Postscript" and "LaTeX or TeX," breaks the current line on either side of the figure, separating it from the text.

Note that this function merely creates a box for the appropriate graphics. You must still choose the file to import. For PostScript files, double click on the box, which will initially contain only the letter "F", to open the EPS Figure panel. You can now use the options in the EPS Panel to select and even display an encapsulated PostScript file.

Editor's Note: For files containing LaTeX format graphics, I'm not too sure what to do. You'll get a box with the LaTeX command `\input` *in it, but I really don't know what you're supposed to do with it. -jw*

Examples: Currently, none provided. This will change at some point in the future.

See Also: EPS Figure panel. [*Ed. Note - To be added. - jw.*]

4.4.2 find-replace

Default Bindings:

Menu -	Edit -> Find & Replace
Keyboard -	M-e f (standard) C-f in `pc-modern.bind` or C-s in `emacs.bind`.

Purpose: To search for text and, if desired, replace it with other text.

Usage: Invoking this command opens the Find & Replace panel. All searches start at the current cursor position. The target text is placed in the "find" box. The two arrow buttons choose the search direction. If LyX finds the text, the cursor moves to the text and remains at the new position. The target text may be optionally replaced with alternate text placed in the "replace" box. In all instances the text is case sensitive.

See Also: TBD.

4.4.3 font-bold

Default Bindings:

Menu - Layout -> Bold Style and
Layout -> Character... to open the "Character Layout" popup

CharacterLayout: option "Bold" from the "Series" list

Keyboard - M-l c [to open the "Character Layout" popup]
M-l b, or M-c b (standard)
C-b in `pc-modern.bind`.

Purpose: Changes the font to an emboldened series or, if the font is already emboldened, reverts to the default series.

Usage: Rather than repeat the same thing for all 9 font selection functions, we'll describe the generic behavior of the font functions here. Some of the font commands have a toolbar button, others have a menu selection. All of them are in the Character Layout popup.

All of the font commands do the following to words:

1. If the cursor is in the middle of a word [between two whitespaces], the entire word changes to the new font.
2. If the cursor is at the beginning or end of a word, the font commands only affect any subsequently typed text.
3. You can also tack text in a different font onto a word. Just move the cursor out of the word to shut off the font, otherwise, you'll change the whole word into a single font.

You can also change the font of a block of text by selecting it with the mouse [or with the keyboard] and then invoking the font command.

Eight of the font commands, `font-code`, `font-emph`, `font-roman`, `font-sans`, `font-size`, `font-smallcaps`, `font-underline`, and `font-bold`, are toggles.

They will remove a font if text is already in it and deactivate a font if it's active.

LyX automatically uses the font of the word next to the cursor. If the cursor is in the middle of a word, it uses the font of the text to the right of the cursor.

One last thing: we've provided this function because LaTeX has a bold font series. We know that many users are used to using boldface for emphasis, and you can make this the default emphasized font by putting the correct command in the LaTeX preamble. However, we suggest that you use `font-emph` to emphasize any text and only use `font-bold` when appropriate.[6]

Examples: This is an example of a bold font series:

> **This is bold text.**

Suppose you have this sentence:

> Here is some text.

If you put the cursor in the word "some" and invoke `font-bold`, you get:

> Here is **some** text.

If you take the first example, select the words "This is," and invoke `font-bold`, you get:

> This is **bold text.**

Now put the cursor at the end of the word "text." Delete the period, invoke `font-bold`, and type in some text. You'll get:

> This is **bold text** and this is not.

If you take the sentence, "Here is some text," put the cursor before the "t" in the word "text", invoke `font-bold`, and type in some text, you'll get:

[6] For example, to make reference to a vector in the body of the text, many mathematician use a bold series.

Here is some **newly added** text.

Now put the cursor after the "d" in "added", and type " funky " [it's important that you type the spaces, too]. You'll get:

Here is some **newly added funky** text.

Lastly, we'll demonstrate mixed fonts. Again, start with the sentence, "Here is some text." We can do the following:

Here is some**thing bold**text.

...by 1) placing the cursor after the "e" in "some"; 2) invoking `font-bold`; 3) typing "thing"; 4) placing the cursor before the "t" in "text"; 5) invoking `font-bold`; 6) typing "bold"; and 7) moving the cursor out of the word.

The last step automatically shuts off the bold font. LyX automatically uses the font of the current word [the word next to the cursor]. If you had not moved the cursor, but instead tried to turn off bold font by invoking `font-bold` again,

Here is some**thing boldtext**.

...would have been the result. If you'd tried to turn off bold font by invoking `font-default`,

Here is some**thing** boldtext.

...would have been the result. Now let's return to:

Here is some**thing bold**text.

We'll try adding text in the middle of the words. Put the cursor bewteen the "e" and the "t" in "some**thing**", type "ly", move the cursor between the "d" and "t" in "**bold**text", and type "old". This is what you'll get:

Here is some**lything bold**oldtext.

Hopefully, these examples helped you understand how the font commands work. All 9 commands listed below in "*See Also*" work the same way, so the above examples hold for them, too.

See Also: Character Layout popup; [*Ed. note - to be added. -jw*]
font-bold; font-code; font-default; font-emph; font-roman; font-sans; font-size; font-smallcaps; font-underline.

4.4.4 font-code

Default Bindings:

Menu - Layout -> Character... to open the "Character Layout" popup

CharacterLayout: option "Typewriter" from the "Family" list

Keyboard - M-l c [to open the "Character Layout" popup]

M-c p (standard)

C-p in `pc-modern.bind`.

Purpose: Changes the font to the typewriter family, or, if the font is already in this family, reverts to the default family.

Usage: A lot of the usage for `font-code` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-code`.

Note that `font-code` shuts off `font-roman` and `font-sans` if either of these families is active.

Examples: Here is an example of the typewriter font family, which `font-code` selects:

`This is Typewriter font.`

See Also: Character Layout popup; [*Ed. note - to be added. -jw*]
font-bold; font-roman; font-sans.

4.4.5 font-default

Default Bindings:

Keyboard - M-c Space

Purpose: Changes the font to the default font.

Usage: A lot of the usage for `font-default` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-default`.

Examples: Suppose you had this mess in your text:

> Hello I'*m a* `really wie`**rd sent**ten*ce and* MY NA**me** is **Del***erium*.

Select this text and invoke `font-default`. You'll change it to:

> Hello I'm a really wierd senttence and my name is Delerium.

See Also: Character Layout popup; [*Ed. note - to be added. -jw*] font-bold.

4.4.6 font-emph

Default Bindings:

Menu -	Layout -> Emphasize Style and
	Layout -> Character... to open the "Character Layout" popup
	CharacterLayout: option "Italic" from the "Shape" list
Toolbar -	Button #7 from the left.
	An exclamation mark.
Keyboard -	M-l c [to open the "Character Layout" popup]
	M-c i, M-c e, and M-l e (standard)
	C-e in `pc-modern.bind`.

Purpose: Changes the font shape to the italic shape, or, if the font is already in this shape, reverts to the default font shape.

Usage: A lot of the usage for `font-emph` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-emph`.

Note that `font-emph` shuts off `font-smallcaps` if that shape is active.

Many new users wonder why this isn't called "font-italic," and why boldface and underlined text don't have their own menu entries or toolbar buttons. LyX has adopted a convention from LaTeX: specify *what* you want to do and let the computer take care of how to do it. The italic font shape is the standard way to emphasize text in typesetting. LyX [and LaTeX] has also adopted this convention. You can, of course, change which font is the emphasized font by making the appropriate changes to the LaTeX preamble, although this will only affect the final output. There are also some interesting things that LaTeX will do to emphasized text that LyX can't [yet] display. It will still be in the final output. For example, LaTeX does certain things for emphasized emphasized text automatically. If you used a specific font, you'd need to do the same thing manually, using specific font changes.

In short, the LyX team *strongly recommends* that you use `font-emph` to emphasize any text [this includes book titles] and save any specific font command for those few special cases where you need to specify exactly how the text is typeset.

Examples: Here's an example of emphasized text:

> *Always use `font-emph` instead of specific font names!*

See Also: Character Layout popup; [*Ed. note - to be added. -jw*]
font-bold; font-smallcaps.

4.4.7 font-free

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left. Brief description of button icon.
Keyboard -	None.

Purpose: [I don't know what this does.<ref>]

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.4.8 font-roman

Default Bindings:

Menu - Layout -> Character... to open the "Character Layout" popup

CharacterLayout: option "Roman" from the "Family" list

Keyboard - M-l c [to open the "Character Layout" popup]
M-c r (standard).

Purpose: Changes the font to the roman family, or, if the font is already of this family, reverts to the default font family.

Usage: A lot of the usage for `font-roman` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-roman`.

Note that `font-roman` shuts off `font-code` and `font-sans` if either of these families is active.

Examples: Here is an explicit use of the roman font family:

This is Roman font.

Note that roman is the default font family.

See Also: Character Layout popup; [*Ed. note - to be added. -jw*]
font-bold; font-code; font-roman; font-sans.

4.4.9 font-sans

Default Bindings:

Menu - Layout -> Character... to open the "Character Layout" popup

CharacterLayout: option "Sans Serif" from the "Family" list

Keyboard - M-l c [to open the "Character Layout" popup]

M-c s (standard).

Purpose: Changes the font shape to the sans serif family, or, if the font is already of this family, reverts to the default font family.

Usage: A lot of the usage for `font-sans` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-sans`.

Note that `font-sans` shuts off `font-code` and `font-roman` if either of these families is active.

Examples: Here is an example of the Sans Serif font family:

This is the Sans Serif font family.

See Also: Character Layout popup; [*Ed. note - to be added. -jw*] font-bold; font-code; font-roman.

4.4.10 font-size

Default Bindings:

Menu - Layout -> Character... to open the "Character Layout" popup

CharacterLayout: Any item from the "Size" list

Keyboard - The M-s prefix, plus one of the following keys:

M-s t

M-s 1 "`font-size tiny`"

M-s s	
M-s 2	"font-size small"
M-s n	
M-s 3	"font-size normal"
M-s l	
M-s 4	"font-size large"
M-s S-L	
M-s 5	"font-size larger"
M-s S-A	
M-s 6	"font-size largest"
M-s h	
M-s 7	"font-size huge"
M-s g	
M-s S-H	
M-s 8	"font-size giant"

Also:

M-l c [to open the "Character Layout" popup]

Purpose: Change the font size.

Usage: This command requires an argument, which is the font size. You must provide the argument in the "`\bind`" command if you use `font-size` in a keybinding file, or in the minibuffer if you use `command-execute` to invoke it.[7]

LaTeX describes font sizes in relative values. LyX uses the following terms: *tiny, small*, *normal*, *large*, *larger*, *largest*, *huge*, *giant*. The *normal* size can be set to `10pt`, `11pt`, or `12pt`, with the other sizes set proportionally.

A lot of the usage for `font-size` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-size`.

You should avoid using `font-size` unless you need to do some fine-tuning. *Never* use `font-size` to make titles or section headings; LyX already does this for you and in a much better fashion.

[7] *Editor's note: Don't do this - there is a bug. For some reason,* `font-size` *crashes LyX at the moment. - jw 10/6/96*

Examples: Here are all eight font sizes:

tiny, small, normal, large, larger, largest, huge, giant.

See Also: Character Layout popup; [*Ed. note - to be added. -jw*]
command-execute;
font-bold.

4.4.11 font-smallcaps

Default Bindings:

Menu - Layout -> Noun Style and

Layout -> Character... to open the "Character Layout" popup

CharacterLayout: option "Small caps" from the "Shape" list

Toolbar - Button #8 from the left.

A person.

Keyboard - M-l c [to open the "Character Layout" popup]

M-l n and M-c c (standard)

C-s in `pc-modern.bind`.

Purpose: Changes the font shape to the small caps shape, or, if the font is already of this shape, reverts to the default font shape.

Usage: A lot of the usage for `font-smallcaps` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-smallcaps`.

Note that `font-smallcaps` shuts off `font-emph` if that shape is active.

The small caps shape is typically used for proper names. Some countries use this convention more frequently than the US does, hence its prominence in LyX.

Examples: Here's what the small caps shape looks like:
THIS IS ALL IN THE SMALL CAPS FONT SHAPE.

See Also: Character Layout popup; [*Ed. note - to be added. -jw*]
font-bold; font-emph .

4.4.12 font-state

Default Bindings:

Menu - Main -> Item

Toolbar - Button # from the left.
Brief description of button icon.

Keyboard - None.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.4.13 font-underline

Default Bindings:

Menu - Layout -> Character... to open the "Character Layout" popup

CharacterLayout: option "Underbar" from the "Bar" list

Keyboard - M-l c [to open the "Character Layout" popup]
C-u or M-c u (standard).

Purpose: Adds an underbar to subsequent or selected text.

Usage: A lot of the usage for `font-underline` works the same way as `font-bold`, so see the entry for `font-bold` to get an idea for how to use `font-underline`.

The LyX team has provided this function only for compatibility with LaTeX. We know that many users are used to underlining text for emphasis or for book titles. While you might need to do this on a typewriter, LyX is not a typewriter, nor should you use it like one. Now, you could make this the default emphasized font by putting the correct command in the LaTeX preamble. However, we suggest that you use `font-emph` to emphasize any text, including book titles, and avoid using `font-underline` at all.[8]

Examples: Here's an example of the underbar font attribute:

> Avoid using underlined text!!!

See Also: Character Layout popup; [*Ed. note - to be added. -jw*] font-bold.

4.4.14 footnote-insert

Default Bindings:

Menu -	Insert -> Footnote
Toolbar -	Button # 7 from the right. A page with an arrow pointing to lines at the bottom of the page.
Keyboard -	M-i f

Purpose: To insert a footnote.

[8] There is one - and only one - case that we can think of where you might need to use `font-underline`. Suppose you're writing the bibliography of a journal article. The journal you're submitting this article to typesets all volume numbers in an underlined font, and wishes all submissions to do the same. This is perhaps the only time you'd need to use `font-underline`.

Usage: Use the keyboard, menu, or toolbar to insert a footnote. A red box appears on the next line. If you're in the middle of a line, the line appears to break around the box [it doesn't really]. Anything you enter in the box will be in the footnote.

To the left of the red box is another, smaller box with the word "foot" written in red on a grey background. Click on this box once with the left mouse button to close the footnote. A closed footnote looks like the word "foot" written as a superscript in red.

You can also open and close footnotes, as well as other types of notes and labels, with the `open-stuff` command. See its entry for keybindings.

One last note: even though footnotes in LyX appear unnumbered, they are. LaTeX does the numbering for you, as well as putting the footnote at the bottom of the correct page, when it processes your file.

Examples: This[9] is a footnote.

See Also: open-stuff; marginpar-insert.

4.4.15 forward-select

Default Bindings:

Keyboard - S-Right

Purpose: Highlight a single character to the right of the cursor. If the cursor is at the end of a line, the cursor moves to the beginning of the next line, and the whitespace in between is selected. The selected text is shown in reverse-video.

See Also: char-forward;

backward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

[9] Hi! I'm a footnote!

4.5 G-K

4.5.1 hfill-insert

Default Bindings:

Menu - Insert -> Special Character -> HFill
Keyboard - C-i or M-i h

Purpose: Insert a LaTeX `\hfill` command.

Usage: The LaTeX `\hfill` stands for "horizontal fill." As the name implies, an *HFill* fills in the current line with blank space. You can put it at the beginning or end of a line, which will force the line against the right or left margin, respectively. However, an *HFill* is most useful in the middle of a line. The example explains it best. The *HFill* has pushed the text on either side of it to the left and right margins.

Important Note: by a "margin," we do *not* mean the page margins. We mean the environment margin. For example, if your document has two columns per page, an *HFill* fills in the line to the left and right margin *of the column*, not the entire page. If you're in a table cell, *HFill* fills to the cell borders, and so on.

You should only use `hfill-insert` for fine tuning. LyX has better ways to justify and format text.

Examples: An `\hfill` command here has pushed this text apart.

See Also: *Editor's Note: Eventually, vfills or the Paragraph popup. -jw*

4.5.2 hyphenation-point-insert

Default Bindings:

Menu - Insert -> Special Character -> Hyphenation Point
Keyboard - C-h or M-i Minus

Purpose: Insert a manual hyphenation point via the LaTeX `\-` command.

Usage: While LaTeX is normally able to hyphenate most words, it has problems with other words, especially non-English words and technical terms. This command allows you to tell LaTeX to hyphenate a single instance of a word that LaTex would otherwise be unable to hyphenate.

To allow LaTeX to properly hyphenate *gnomon*, you type it as `gno\-mon`, where you insert the "`\-`" by means of the hyphenation command. Use this command to do the same in LyX [see the example].

Note: To tell LaTeX how to hyphenate *all* instances of a word in a document, insert a LaTeX `\hyphenation{}` command in the preamble,[10] where the argument is a space-delimited list of the words to be hyphenated, with the hyphenation points noted by dashes, as in `\hyphenation{gno-mon gno-mons gno-mon-ly}`.

Examples: This is what *gnomon* looks like in LyX when hyphenated.

See Also: ?

4.5.3 inset-formula-latex-deletable-insert

Default Bindings:

None.

Purpose: I don't know what this is used for, though it seems to go into a kind of math-mode.

Usage: ?

Examples: ?

See Also: ?

4.5.4 inset-formula-latex-insert

Default Bindings:

None.

[10] Edit the preamble from the menu by selecting Layout, then LaTeX Preamble .

Purpose: I don't know what this is used for, though it seems to go into a kind of math-mode.

Usage: ?

Examples: ?

See Also: ?

4.5.5 inset-latex-deletable-insert

Default Bindings:

None.

Purpose: ?

Usage: ?

Examples: ?

See Also: ?

4.5.6 inset-latex-insert

Default Bindings:

None.

Purpose: ?

Usage: ?

Examples: ?

See Also: ?

4.5.7 keymap-off

Default Bindings:

Menu - Options -> Keyboard

Key Mappings: No key mapping

Keyboard - M-k o or M-k x

Purpose: Turn off keyboard character keymapping.

Usage: In addition to the default keymap, the keyboard can have a primary keymap, a secondary keymap, or both. If you have previously defined and selected either the primary or secondary keymaps, you can turn-off this selection by issuing the `keymap-off` command.

Editor's Note - We probably need to add some more info. - jw

Examples: ?

See Also: keymap-primary; keymap-secondary.

Keyboard keymapping is described in more detail in ...

4.5.8 keymap-primary

Default Bindings:

Menu - Options -> Keyboard

Key Mappings: Primary key mapping

Keyboard - M-k 1

Purpose: Select the primary keymap.

Usage: Self-explanatory. See `keymap-off` for more details about using multiple keymaps.

See Also: keymap-off

4.5.9 keymap-secondary

Default Bindings:

Menu -	Options -> Keyboard
	Key Mappings: Secondary key mapping
Keyboard -	M-k 2

Purpose: Select the secondary keymap.

Usage: Self-explanatory. See `keymap-off` for more details about using multiple keymaps.

See Also: keymap-off

4.5.10 keymap-toggle

Default Bindings:

Menu -	Options -> Keyboard
	Key Mappings: Mapping Switch
Keyboard -	M-k t

Purpose: Allows you to toggle back and forth between the primary and secondary keymaps (either of which can be set to a value of `default`).

Usage: Self-explanatory. See `keymap-off` for more details about using multiple keymaps.

See Also: keymap-off

4.6 L

4.6.1 label-insert

Default Bindings:

Menu - Insert -> Label

Input: (enter label name in input box)

Keyboard - M-i l

Purpose: Insert a label (and create a character string label key for it) for use in a cross-reference to either the current environment entity or the printed page number of the label location.

Usage: This command inserts a LaTeX `\label{key}` command into the document, which contains a case-sensitive alphanumeric label `key` that you define. This command allows LaTeX to track the current environment and printed page number for subsequent[11] cross-reference via the `ref-insert` command. The current environment is the section, table, figure, equation, enumerated item, *etc.*, that the label is within. Note that the actual printed label is not displayed until you view or print your document; LyX displays instead on the screen the symbolic label, which is the character string label that you defined.

Note that you can't define labels for math equations using this command. LyX labels equations using a combination of the `math-number` command and the actual LaTeX command `\label{key}`. See the User's Guide for more information about labelling and numbering equations.

Examples: If the current environment is a section, as it is here at this point, you can insert a label with the following result:. This label can then be referenced, and the result will be the section number for this section, as Section 4.6.1.

[11] Actually, it doesn't need to be a subsequent reference. You can reference a label *before* it is defined, but you must then run LaTeX multiple times to allow LaTeX to resolve these types of references.

See Also: ref-insert; math-number.

You can find further description of the LaTeX labeling and cross-referencing method in the LyX User's Guide or any LaTeX user's guide.

4.6.2 layout

Default Bindings:

Toolbar - Pull-down box at the left end of the toolbar.

Keyboard - M-p prefix plus one of the following:

M-p 1	"`layout Chapter`"
M-p 2	"`layout Section`"
M-p 3	"`layout Subsection`"
M-p 4	"`layout Subsubsection`"
M-p 5	"`layout Paragraph`"
M-p 6	"`layout Subparagraph`"
M-p b	"`layout Itemize`"
M-p d	"`layout Description`"
M-p e	"`layout Enumerate`"
M-p i	"`layout Itemize`"
M-p l	"`layout List`"
M-p n	"`layout Enumerate`"
M-p p	"`layout Lyx-Code`"
M-p q	"`layout Quote`"
M-p S-Q	"`layout Quotation`"
M-p s	"`layout Standard`"
M-p t	"`layout Title`"
M-p S-at	"`layout Section*`"
M-p S-numbersign	"`layout Subsection*`"
M-p S-dollar	"`layout Subsubsection*`"

Purpose: To select a paragraph environment.

Usage: This function requires an argument, which is the name of the paragraph environment to use. Yes, we realize the nomenclature is confusing. Unfortunately, we've changed our terminology but not the function names.

There are more possible arguments than those listed in the keybindings. In fact, the possible arguments change depending on what document class you're using. In any case, there are so many different paragraph environments that we couldn't go into detail about each one here. [*Editor's Note- Maybe we need another chapter, just for the layouts? - jw*]

See Also: The appropriate section of UserGuide.lyx.

4.6.3 layout-character

Default Bindings:

Menu - Layout -> Character

Character Layout: (controls for character font settings)

Keyboard - M-l c

Purpose: Activates the Character Layout pop-up, which allows you to control the character family, series, shape, size, and underlining font details.

Usage: You can change the character appearance just once, for selected text, or for all subsequently typed text.

Yes, we realize the nomenclature is confusing. Unfortunately, we've changed our terminology but not the function names.

See Also: layout-paragraph; layout-document.

4.6.4 layout-document

Default Bindings:

Menu - Layout -> Document

Document Layout: (controls for document format settings)

Keyboard - M-l d

Purpose: Activates the Document Layout pop-up, which allows you to control parameters affecting the entire document, including: class, pagestyle, default font type and size, language, paper size, number of paper sides and orientation, number of columns, paragraph separation method and spacing, section numbering and table of contents depth, and other parameters.

See Also: layout-character; layout-paragraph.

4.6.5 layout-paragraph

Default Bindings:

Menu - Layout-> Paragraph

Paragraph Layout: (controls for paragraph format settings)

Keyboard - M-l p

Purpose: Activates the Paragraph Layout pop-up, which allows you to control parameters affecting the current paragraph, including: alignment, vertical spacing above and below the paragraph, line spacing, page breaking, indention, and label width (for appropriate paragraphs).

Usage: You may make changes either to the current paragraph or to a group of selected paragraphs. Changes made to the current paragraph [or the selected group] do not affect other paragraphs.

Yes, we realize the nomenclature is confusing. Unfortunately, we've changed our terminology but not the function names.

See Also: layout-character; layout-document.

4.6.6 layout-preamble

Default Bindings:

Menu - Layout -> LaTeX Preamble

LaTeX Preamble: (edit window for direct modification of the LaTeX preamble)

Keyboard - M-l l

Purpose: Activates the LaTeX Preamble pop-up, which allows you to directly make additions to the LaTeX preamble that LyX inserts in the LaTeX version of the document.

Usage: In the LaTeX preamble, you can insert non-standard LaTeX parameter settings or LaTeX commands.

Each document has its own preamble. LyX currently adds certain commands to the preamble based on settings you've chosen from within LyX. Some things LyX just can't do yet, but LaTeX can. Or, you may want to use some custom LaTeX macros. The LaTeX Preamble popup is where to do it.

Example: The following is a typical preamble that changes default settings for twocolumn mode, creates a simple custom LaTeX command for later use in this document, and then reads in some other personal settings and/or macros from an external file. Remember, if you want your documents to be portable, you need to bundle any such external files with the document file(s):

```
\columnsep 10pt
\columnseprule 2pt
\newcommand{\astm}[2]{#1\,#2}
%
\input{\home\homenode\myname\latex\macros\my_macros.def}
```

See Also: The appropriate chapter in LyX User's Guide.

4.6.7 layout-quotes

Default Bindings:

Menu - Layout -> Quotes

Quotes: (controls for setting desired quotation marks)

Keyboard - M-l q

Purpose: Activates the Quotes pop-up, which allows you to set the type of quote marks used by LyX.

Usage: Self explanatory. Different languages use different symbols for the default quotes. For example, French uses , not “”. Selecting a different type of quote changes the behavior of the " and ' keys. Currently, LyX only supports English, French, and German quotes.

See Also: ?

4.6.8 line-begin

Default Bindings:

Keyboard - Home in `pc-modern.bind`.
C-a in `emacs.bind`.

Purpose: Move the cursor to the beginning of the current line.

See Also: char-forward; char-backward; word-forward; word-backward; line-end; tab-forward; up; down; screen-up; screen-down; buffer-begin; buffer-end.

4.6.9 line-begin-select

Default Bindings:

Keyboard - S-Home

Purpose: Select (highlight) text from the current cursor location to the beginning of the current line. The selected text is shown in reverse video.

See Also: line-begin;

backward-select; forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.6.10 line-delete-forward

Default Bindings:

Keyboard - C-k or M-e k

Purpose: Delete text from the current cursor location to the end of the screen line.

Usage: If the cursor is at the end of the line there is no action. If the cursor is at the beginning of the line, the text of the entire line is deleted but the empty line is retained for insertion of new text.

See Also: ?

4.6.11 line-end

Default Bindings:

Keyboard - End in `pc-modern.bind`.
C-e in `emacs.bind`.

Purpose: Move the cursor to the end of the current line.

See Also: char-forward; char-backward; word-forward; word-backward; line-end; tab-forward; up; down; paragraph-up; paragraph-down; screen-up; screen-down; buffer-begin; buffer-end.

4.6.12 line-end-select

Default Bindings:

Keyboard - S-End

Purpose: Select (highlight) text from the current cursor location right to the end of the current line, in the current buffer. The selected text is shown in reverse video.

See Also: line-end;

backward-select; forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.6.13 lyx-quit

Default Bindings:

Menu - File -> Quit

Keyboard - M-f q
M-F4 in `pc-modern.bind`.
C-x c or C-x C-c in `emacs.bind`.

Purpose: Quit LyX and close all buffers and windows.

Usage: After entering the command, a pop-up for each open and modified buffer appears, asking you if you want to save that buffer. This only happens if any open documents need saving.

At this point, one of two things happen. If you executed `lyx-quit` from the keyboard, LyX simply quits. If you used Quit from the File menu, however, LyX asks you if you really want to quit.

If you decide to customize the keybinding for this one, choose a complicated key sequence.

See Also: ?

4.7 M

4.7.1 marginpar-insert

Default Bindings:

Menu - Main -> Item

Toolbar - Button # from the left.
Brief description of button icon.

Keyboard - C-S-M-W-key in `file1.bind`.
C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.7.2 mark-off

Default Bindings:

Menu - Main -> Item

Toolbar - Button # from the left.
Brief description of button icon.

Keyboard - C-S-M-W-key in `file1.bind`.
C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.7.3 mark-on

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left. Brief description of button icon.
Keyboard -	C-S-M-W-key in `file1.bind`. C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.7.4 mark-toggle

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left. Brief description of button icon.
Keyboard -	C-S-M-W-key in `file1.bind`. C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.7.5 math-delim

Default Bindings:

Menu - Math -> Math Panel

MathPanel: Third button from left to right.
Two square brackets enclosing a blue rectangle.

Purpose: To insert math delimiters [i. e. - parentheses, brackets, etc.].

Usage: Requires two arguments, the left and right delimiters. Each delimiter can be specified by either a LaTeX name or a valid character . Use only one space between items.

If you use this function from the Math Panel, you'll get another popup that allows you to choose the left and right delimiters.

Examples: "`math-delim { rangle`" result in {⟩ with the cursor at the small rectangle.

See Also: UserGuide, chapter Math.

4.7.6 math-display

Default Bindings:

Menu - Math -> Display

MathPanel: Rightmost button.
A small blue rectangle with two horizontal lines below and above.

Purpose: In text mode, this is used to create a new displayed equation. In math mode it changes the display status of a formula from inlined to display and vice-versa.

Usage: Self-Explanatory

Examples: This is a displayed equation:

$$a = b + c$$

See Also: UserGuide, chapter Math.

4.7.7 math-greek

Default Bindings:

Keyboard - M-m g

Purpose: Enter a single Greek character.

Usage: After invoking this function, the next character you type will be translated to a Greek character. Anything typed after that character will be a normal Latin character.

Examples: $\Gamma\Omega\alpha\beta$.

See Also: UserGuide, chapter Math.

4.7.8 math-greek-toggle

Default Bindings:

Keyboard - M-m S-G

Purpose: Turn on the math Greek keyboard.

Usage: Every typed Latin character is mapped to Greek until this command is invoked again. If you leave math mode, the Greek keyboard remains active.

See Also: UserGuide, chapter Math.

4.7.9 math-insert

Default Bindings:

Keyboard - The M-m prefix, plus one of the following keys:

M-m i	"`math-insert int`"
M-m o	"`math-insert oint`"
M-m p	"`math-insert partial`"
M-m s	"`math-insert sqrt`"
M-m S-S	"`math-insert sum`"

Purpose: To insert math objects or symbols.

Usage: Requires an argument, which can be the LaTeX name of either a symbol or an object. The possible arguments are:

- `frac`
- `int`
- `oint`
- `partial`
- `sqrt`
- `sum`

Examples: "`math-insert int`" or "`math-insert frac`".

See Also: UserGuide, chapter Math.

4.7.10 math-limits

Default Bindings:

Keyboard - M-m l

Purpose: To change the appearance of the limits above and below an integral symbol, a limit, a summation, etc.

Usage: Put the cursor *before* [i.e. to the left] of the symbol with the limits, then invoke `math-limits`.

Examples: This is a normal integral $\int_0^\infty$ and this was changed with limits $\int\limits_0^\infty$.

See Also: UserGuide, chapter Math.

4.7.11 math-matrix

Default Bindings:

Menu - Math -> Math Panel

MathPanel: A grid of 3×3 small blue rectangles.

Purpose: To insert a matrix.

Usage: This command requires as arguments the dimensions of the matrix (two integer numbers separated by an space) and, optionally, the alignment. The alignment is a word composed of the vertical alignament (`b`, `c` or `t`) and the horizontal alignments (`l`, `c` or `r`).

Examples: `math-matrix 3 3 bccc` results in , a 3×3 matrix with its bottom at the baseline and all columns centered.

See Also: UserGuide, chapter Math.

4.7.12 math-mode

Default Bindings:

Menu -	Math -> Math mode
Toolbar -	Second button from the right. A blue equation.
Keyboard -	M-c m

Purpose: Set math mode.

Usage: In text mode you can insert a math expression.

Examples: Most of the entries for math functions have examples of math mode in them. Here's another example:

Here is some text. Now suppose I wanted an inlined equation, such as $\frac{A \cdot e^x}{x+1}$, in my text. I'd use math mode.

See Also: UserGuide, chapter Math.

4.7.13 math-mode-old (OBSOLETE)

Default Bindings: None

Purpose: Old math mode. Only for backward compatibility. Support for this will probably disappear at some point.

Usage: Should not be used.

4.7.14 math-nonumber

Default Bindings: None

Purpose: To supress the number of a line in a numbered equation array.

Usage: The cursor position must be at the line whose number will be supressed.

Examples: This example contains a two-line, numbered equation. Ordinarily, both lines get numbered, which makes no sense for this equation:

$$\begin{aligned}(x+y)(x-y) &= x^2 - xy + xy - y^2 \\ &= x^2 - y^2 \qquad (4.1)\end{aligned}$$

Notice that only the bottom line has a number. The top line had its number suppressed using "`math-nonumber`".

See Also: UserGuide, chapter Math.

4.7.15 math-number

Default Bindings:

Keyboard - M-m n

Purpose: Toggles the numbering status of an equation. Changes the LaTeX environment automatically.

Usage: Self-explanatory

Examples: This is a numbered equation

$$x = 2 - y \qquad (4.2)$$

And this is not

$$x = y - 2$$

See Also: UserGuide, chapter Math.

4.7.16 math-size

Default Bindings:

Keyboard - M-m ?

Editor's Note - there may be no default binding for this command. - jw

Purpose: To change arbitrarily the size used by math fonts inside a context. Provides an interface to the LaTeX math mode font size commands.

Usage: Requires one argument. Valid arguments are `displaystyle`, `textstyle`, `scriptstyle` and `scriptscriptstyle`.

As for how to use these arbitrary font sizes, see a good LaTeX book. You should use this for fine-tuning only. LyX [and LaTeX] will ordinarily set an appropriate font size for you.

Examples: This is the normal size of a fraction inside text $\frac{1}{2}$, and this other $\displaystyle\frac{1}{2}$ is the result of using `math-size displaystyle.` [There's no visible difference between the two inside LyX, but there will be a difference once you print.]

See Also: UserGuide, chapter Math;
any good LaTeX guide.

4.7.17 melt

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left. Brief description of button icon.
Keyboard -	C-S-M-W-key in `file1.bind`. C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.7.18 menu-open

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left. Brief description of button icon.
Keyboard -	C-S-M-W-key in `file1.bind`. C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a “;” or place in multiple paragraphs.

4.7.19 meta-prefix

Default Bindings:

Menu -	Main -> Item
Toolbar -	Button # from the left. Brief description of button icon.
Keyboard -	C-S-M-W-key in `file1.bind`. C-S-M-W-key in `file2.bind`.

Purpose: Description.

Usage: Description.

Examples: Examples.

See Also: Other entries or documents. Separate many references by either a ";" or place in multiple paragraphs.

4.8 N-R

4.8.1 note-insert

Default Bindings:

Menu - Insert -> Note

Keyboard - M-i n

Purpose: Creates at the current cursor position, and for screen interactive use only, a LyX Note, which is visible by the word "Note" within a bright-yellow box. The box is tied to a pop-up, within which you can insert a message.

Usage: After inserting the note at the desired text position, double-click on the note to see bring-up the pop-up, and type-in your message. Neither the box nor the message appear in the printed output.

Examples: To the immediate left is an inserted note. Do a left-double-click on the box with your mouse to see the message.

See Also: note-next.

4.8.2 note-next

Default Bindings:

Menu - Edit -> Go to Note

Keyboard - M-e n

Purpose: Move the current cursor and screen location to the next LyX Note in the current buffer. There is no action if there are no more notes in the document (or no notes at all).

See Also: note-insert.

4.8.3 open-stuff

Default Bindings:

Menu - Edit -> Floats -> Open/Close

Keyboard - C-o
(or use the appropriate mouse button action)

Purpose: Opens a LyX inset for editing or parameter modification.

Usage: Place the cursor to the left of the entity and execute the command. Or, click on the center of the object with the mouse. open-stuff will open a math-mode block (left mouse button, single click), figure, note, footnote or margin note (left mouse button, double-click).

Typically, the object you're opening is some sort of inset or float. You can use `melt` to remove the float and merge the text back into the body of the document.

See Also: melt.

4.8.4 paste

Default Bindings:

Menu - Edit -> Paste

Toolbar - Button #6 from the left.
Schematic of a clipboard with arrow to a document.

Keyboard - M-e p
C-v or S-Insert in `pc-modern.bind`.
C-y in `emacs.bind`.

Purpose: Paste the contents of the paste buffer into the current buffer, at the current cursor location.

Usage: Upon starting LyX, the paste buffer is initially empty. During a `cut` or `copy` operation, the selected text will be stored in the paste buffer, overwriting anything previously stored in the buffer.

If a selection is highlighted at the current cursor location, LyX `paste` inserts the paste buffer text *after* the selection. It does not overwrite; it only operates in *insert* mode.

See Also: cut; copy.

4.8.5 prefix-arg

Default Bindings:

Menu - ?

Toolbar - ?
?

Keyboard - ?

Purpose: ?

Usage: ?

Examples: ?

See Also: ?

4.8.6 protected-space-insert

Default Bindings:

Menu - Insert -> Special Character -> Protected Blank

Keyboard - C-Space

Purpose: Insert an interword space at which a line break will not be allowed either on the LyX screen or in the LaTeX output.

Usage: The protected space is represented on the screen as a small, magenta, "square cup" symbol, like this , which if you are reading a printed copy looks something like ⊔.

This is sometimes also called a "hard space" or a "fixed space" because it is not resized like other whitespace.

Use the protected space for phrases that you don't want broken at the end of a line, or to insert more than one word into certain kinds of headings. For example, the Description list always treats the first word of the paragraph as an item label and boldfaces it. If you wanted to use more than one word as an item label, use protected space between the words.

Never use `protected-space` as a subtitute tab-stop! LyX has far, far better ways of formatting text, so use them.

Examples: This phrase, "the whole nine yards," will not be broken at the end of a line. And if we had a description list with multi-word labels:

entry #1 This is one.

entry #2 This is two. Notice the protected space between the words in the item label.

See Also: The User's Guide

4.8.7 quote-insert

Default Bindings:

Keyboard - C-q, S-C-", M-i ', or M-i S-"

Purpose: Insert a typewriter-style double quote symbol, ", at the current cursor location.

Usage: On the default keyboard, the grave and apostrophe keys produce matching single quote characters, and the double-quote key produces context-sensitive anti-symmetric double quotes. While it should normally be used only in rare situations, the `quote-insert` command allows you to easily produce the less esthetically pleasing typewriter-style double quote symbol.

This command also allows you to create a doube-quote character when you've changed the behavior of the " key using the `layout-quotes` command.

See Also: layout-quotes.

4.8.8 redo

Default Bindings:

Menu - Edit -> Redo

Keyboard - M-e r
C-y in `pc-modern.bind`.

Purpose: Attempts to redo the last major editing command that was undone with the `undo` command.

Usage: Self-explanatory. `redo` may not always succeed, or it may redo more editing than you expected. The only limit to how many times you can perform a `redo` is the beginning of the undo buffer.

See Also: undo.

4.8.9 ref-insert

Default Bindings:

Menu - Insert -> Cross-Reference
Insert Reference: (cross-reference control panel)

Keyboard - M-i r

Purpose: Insert a cross-reference (using a character string label key), based on a previously defined label.

Usage: This command inserts either a LaTeX `\ref{key}` or a `\pageref{key}` command, as appropriate, into the document. The `key` parameter is the case-sensitive alphanumeric label that you previously defined using `label-insert`. Executing the `ref-insert` command brings up the Insert Reference pop-up control panel, which shows you a selection list of previously defined references. Also on this panel, you are given the option of referencing either the environment reference related to that label (like the section number) or the page number at which the label is located.

The `\ref(key)` command allows LaTeX to track the specified key and substitute the appropriate cross-reference. The environment referenced can be a section, table, figure, equation number, enumerated item, *etc.* Note that the actual printed label is not displayed until you view or print your document; LyX displays instead on the screen the symbolic label, which is the character string label key that you defined.

Note that labels defined in math mode don't become visible to the Insert Reference popup until after you close and reopen the document.

Examples: See `label-ref` for an example of referencing a section number. That same label from that example can also be used to reference the page number at that location; for example, the page number of that location is Page 64.

See Also: label-insert; math-number.

You can find further description of the LaTeX labeling and cross-referencing method in the LyX User's Guide or any good LaTeX user's guide.

4.9 S

4.9.1 screen-down

Default Bindings:

Keyboard - PageDown
C-v or C-x] in `emacs.bind`.

Purpose: Moves the cursor down one screenful of text while maintaining the current relative cursor screen position.

Usage: Self-explanatory. Note that repeated `screen-down` and `screen-up` commands will not necessarily return the cursor to its previous location.

See Also: char-forward; char-backward; word-forward; word-backward; line-begin; line-end; tab-forward; up; down; screen-up; buffer-begin; buffer-end.

4.9.2 screen-down-select

Default Bindings:

Keyboard - S-PageDown

Purpose: Highlights (selects) the text between the starting and ending cursor positions during a `screen-down` move. The highlighted text is shown in reverse video.

See Also: screen-down;
backward-select; forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; buffer-begin-select; buffer-end-select.

4.9.3 screen-recenter

Default Bindings:

Keyboard - C-l
M-x c in `pc-modern.bind`.

Purpose: Repositions the screen so that it is centered on the current cursor location. The position of the cursor in the text does not change.

See Also: ?

4.9.4 screen-up

Default Bindings:

Keyboard - PageUp
C-x [in `emacs.bind`.

Purpose: Moves the cursor up one screenful of text while maintaining the current relative cursor screen position.

Usage: Self-explanatory. Note that repeated `screen-down` and `screen-up` commands will not necessarily return the cursor to its previous location.

See Also: char-forward; char-backward; word-forward; word-backward; line-begin; line-end; tab-forward; up; down; screen-down; buffer-begin; buffer-end.

4.9.5 screen-up-select

Default Bindings:

Keyboard - S-PageUp

Purpose: Highlights (selects) the text between the starting and ending cursor positions during a `screen-up` move. The highlighted text is shown in reverse video.

See Also: screen-up;
backward-select; forward-select; up-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.9.6 self-insert

Default Bindings:

Menu - Main -> Item

Toolbar - Button # from the left.
Brief description of button icon.

Keyboard - None.

Purpose: Don't know what this does<ref>.

Usage: ?

Examples: ?

See Also: ?

4.9.7 server-char-after

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

Examples: Examples.

See Also: server-get-font; server-get-latex; server-get-layout; server-get-name; server-get-xy; server-notify; server-set-xy.

4.9.8 server-get-font

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

See Also: server-char-after; server-get-latex; server-get-layout; server-get-name; server-get-xy; server-notify; server-set-xy.

4.9.9 server-get-latex

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

See Also: server-char-after; server-get-font; server-get-layout; server-get-name; server-get-xy; server-notify; server-set-xy.

4.9.10 server-get-layout

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage:

See Also: server-char-after; server-get-font; server-get-latex; server-get-name; server-get-xy; server-notify; server-set-xy.

4.9.11 server-get-name

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

See Also: server-char-after; server-get-font; server-get-latex; server-get-layout; server-get-xy; server-notify; server-set-xy.

4.9.12 server-get-xy

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

See Also: server-char-after; server-get-font; server-get-latex; server-get-layout; server-get-name; server-notify; server-set-xy.

4.9.13 server-notify

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

See Also: server-char-after; server-get-font; server-get-latex; server-get-layout; server-get-name; server-get-xy; server-set-xy.

4.9.14 server-set-xy

Default Bindings: None.

Purpose: One of the functions for the LyX server.

Usage: ?

See Also: server-char-after; server-get-font; server-get-latex; server-get-layout; server-get-name; server-get-xy; server-notify.

4.9.15 spellchecker

Default Bindings:

Menu - Edit -> Spellchecker

Spellchecker: (control panel for the spellchecker)
or

Options -> Spellchecker Options

Spellchecker Options: Start Spellchecker

Keyboard - M-e s

Purpose: Start spellchecker.
[*Author's Note: Need more detail here. -<ref>*]

See Also: spellcheck popup. [*Editor's Note: To be added later - jw.*]

4.9.16 symbol-insert

Default Bindings:

Keyboard - M-c m

Purpose: This command is simply an alias for the command `math-mode`.

See Also: math-mode.

4.10 T-U

4.10.1 tab-forward

Default Bindings:

Keyboard - Tab

Purpose: Move the cursor position forward to the left margin at the start of the next paragraph or list item.

Usage: Note that although other motion commands have a text selection parallel, `tab-forward` does not; there is currently no "`tab-forward-select`" command in LyX.

See Also: char-forward; char-backward; word-forward; word-backward; line-begin; line-end; up; down; screen-up; screen-down; buffer-begin; buffer-end.

4.10.2 table-insert

Default Bindings:

Menu - Edit -> Table -> Insert table

Table: (table size and insertion control panel)

Toolbar - First button on the right.
Table icon showing a four by four table with gridlines.

Purpose: Inserts a LaTeX table (`tabular` environment) into the text.

Usage: Executing the command creates a pop-up table control panel that allows you to set the initial number of rows and columns. Hitting either the control panel OK or Apply buttons inserts the table at the current cursor location. Text can then be inserted into any cell of a table by selecting the cell and typing. The table can be modified by placing the cursor anywhere in the table and clicking the right mouse button, which brings up a table control menu.

Examples: Executing the `table-insert` function at the end of this sentence, and accepting the default values, results in the following:

1	2	3	4	5
A	x			y
B		x	y	
C		y	x	
D	y			x

We took the liberty of adding a little text to the table.

See Also: TBD. We'll think of something, to be sure...

4.10.3 tex-mode

Default Bindings:

Menu - Layout -> Tex Style

Toolbar - Fourth button from the right.
Button with TEX on the face

Keyboard - M-l t or M-c t

Purpose: Enter LATEX code.

Usage: The `tex-mode` command is a toggle function. When first executed, `tex-mode` begins accepting characters as a LATEX command rather than text; when `tex-mode` is executed again this action stops. The result must be a valid and appropriate LATEX command, or an error will result when the file is processed with LATEX.

Examples: The characters accepted as a LATEX command are shown on the screen in red, as this example of the LATEX `\relax` command shows: “”.

See Also: math-mode.

4.10.4 toc-update

Default Bindings:

Menu - Edit -> Table of Contents
Table of Contents: (table of contents control panel)

Keyboard - M-e t

Purpose: This function brings up the Table of Contents pop-up.

Usage: Despite having “update” as part of the name, `toc-update` merely opens the Table of Contents pop-up. Once again, the purpose of the function changed, but hte name didn’t. Sorry for the confusion.

You can update the table of contents using the Update button on the control panel, but the primary purpose of the Table of Contents pop-up is browsing.

See Also: Table of Contents popup. [*Ed. Note: Needs to be documented - jw*]

4.10.5 toolbar-add-to

Default Bindings: None.

Purpose: Used in the `${HOME}.lyxrc` and `system.lyxrc` files to add a toolbar button.

Usage: ?

See Also: toolbar-push.

4.10.6 toolbar-push

Default Bindings: None.

Purpose: Used in the `${HOME}.lyxrc` and `system.lyxrc` files to add a toolbar button.

Usage: ?

See Also: toolbar-add-to.

4.10.7 undo

Default Bindings:

Menu - Edit -> Undo

Keyboard - M-e u

C-z in `pc-modern.bind`.
C-x u or C-~S-slash in `emacs.bind`.

Purpose: Undo the last major editing command.

Usage: Self explanatory.

Two things: first, `undo` has unlimited depth. You can keep invoking `undo` until you run out of changes and the document is back in the state it was in when you loaded it.

Second, `undo` does not work everywhere. You cannot use it inside of math-mode, for example.

See Also: redo.

4.10.8 up

Default Bindings:

Keyboard - `Up`
`C-p` in `emacs.bind`.

Purpose: Moves the cursor down up line. If the cursor starting-position is at the top of the screen-display, the buffer scrolls downward to display the new current line about 1/4 of the screen-height from the top of the screen.

See Also: char-forward; char-backward; word-forward; word-backward; tab-forward; line-begin; line-end; down; screen-up; screen-down; buffer-begin; buffer-end.

4.10.9 up-select

Default Bindings:

Keyboard - `S-Up`

Purpose: To select text from the current cursor position to the same position one line up. If the previous line is shorter than the current line, the cursor simply moves to the end of the previous line.

See Also: up;

backward-select; forward-select; down-select; word-forward-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.11 W-Z

4.11.1 word-backward

Default Bindings:

Keyboard - C-Left

Purpose: Moves the cursor backward by one word.

Usage: Self explanatory. If you are already at the beginning of a word, the cursor moves to the first letter of the previous word. If you are in the middle or at the end of a word, the cursor moves to the first letter of that word.

See Also: char-forward; char-backward; word-forward; tab-forward; line-begin; line-end; up; down; screen-up; screen-down; buffer-begin; buffer-end.

4.11.2 word-backward-select

Default Bindings:

Keyboard - S-C-Left

Purpose: Highlights (selects) the text between the starting and ending positions during a `word-backward` move. The highlighted text is shown in reverse video.

See Also: word-backward;
backward-select; forward-select; up-select; down-select; word-forward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.11.3 word-capitalize

Default Bindings:

Keyboard - M-c Right

Purpose: Capitalizes the next character to the right of the current cursor, and moves the cursor position to the beginning of the next word.

Usage: No case change occurs for preselected text.

Examples: If your current cursor location is immediately to the left of the "p" in the word *description* and you execute the `word-capitalize` function, *description* will change to *descriPtion.*

See Also: word-lowcase; word-upcase.

4.11.4 word-delete-backward

Default Bindings:

Keyboard - C-BackSpace

Purpose: Deletes the previous word.

Usage: If the cursor is in between words (in whitespace), then the previous word is deleted.

If the cursor is in the middle of a word, then the beginning of the word is deleted to the cursor location.

If the cursor is at the beginning of a paragraph, then the last word of the previous paragraph is deleted. Also in this case, if both paragraph have the same environment, LyX will join the two paragraphs.

See Also: word-delete-forward.

4.11.5 word-delete-forward

Default Bindings:

Keyboard - C-Delete

Purpose: Deletes the next word.

Usage: If the cursor in between words (in whitespace), then the next word is deleted.

If the cursor is in the middle of a word, then the end of the word is deleted to the cursor location.

If the cursor is at the end of a paragraph, one of two things can happen. If the current and next paragraph have the same paragraph environment, LyX merges to the two paragraphs, but doesn't delete any words. If the two paragraphs have different environments, nothing happens. Note that this is different from the behavior of `word-delete-backward`.

See Also: word-delete-backward.

4.11.6 word-forward

Default Bindings:

Keyboard - C-Right

Purpose: Moves the cursor forward by one word.

Usage: Unlike `word-backward`, this function always displays the same behavior. It always moves the cursor to the beginning of the next word, no matter where the starting cursor location may be.

See Also: char-forward; char-backward; word-forward; word-backward; tab-forward; line-begin; line-end; up; down; screen-up; screen-down; buffer-begin; buffer-end.

4.11.7 word-forward-select

Default Bindings:

Keyboard - S-C-Right

Purpose: Highlights (selects) the text between the starting and ending positions during a `word-forward` move. The highlighted text is shown in reverse video.

See Also: word-forward;
backward-select; forward-select; up-select; down-select; word-backward-select; line-begin-select; line-end-select; screen-up-select; screen-down-select; buffer-begin-select; buffer-end-select.

4.11.8 word-lowcase

Default Bindings:

Keyboard - M-c Down

Purpose: Forces a word to all lowercase, then moves the cursor to the start of the next word.

Usage: Only the portion of the word to the right of the cursor is affected. No case change occurs for preselected text.

Examples: If your current cursor location is immediately to the left of the "p" in the word *DESCRIPTION* and you execute the `word-lowcase` function, *DESCRIPTION* will change to *DESCRIption*.

See Also: word-capitalize; word-upcase.

4.11.9 word-upcase

Default Bindings:

Keyboard - M-c Up

Purpose: Forces a word to all uppercase, then moves the cursor to the start of the next word.

Usage: Only the portion of the word to the right of the cursor is affected. No case change occurs for preselected text.

Examples: If your current cursor location is immediately to the left of the "p" in the word *description* and you execute the word-upcase function, *description* will change to *descriPTION*.

See Also: word-capitalize; word-lowcase.

Chapter 5

Bindings

5.1 Introduction

This chapter is really just a huge cross-reference. Orignally, we wanted to list all of the key bindings followed by the function it's bound to. That way, a user can look up a key to find out what it does. We then decided, what the hey, why not include the default toolbar and menu bindings, too.

If you haven't guessed by now, this chapter won't have the standard entries which you've seen elsewhere in this manual. The form is really self-explanatory.

5.2 Toolbar

5.3 Menu

5.3.1 File

5.3.2 Edit

5.3.3 Layout

5.3.4 Insert

5.3.5 Math

5.3.6 Options

5.3.7 Documents

5.3.8 Help

5.4 Keyboard

5.4.1 Specific to `emacs.bind`

5.4.2 Specific to `pc-modern.bind`

5.4.3 Standard motion bindings

5.4.4 Standard selection bindings

5.4.5 Menu Accelerators

5.4.6 Generic Accelerators

Remainder of Document is Incomplete.