

The KWrite Handbook

**Thad McGinnis, Anne-Marie Mahfouf, Anders
Lund, Christoph Cullmann, and Lauri Watts**



The KWrite Handbook

Contents

1	Introduction	1
2	Some Fundamentals	2
2.1	Drag and Drop	2
2.2	Command Line Options	2
2.2.1	Specify a File	2
2.2.2	Specify a file on the internet	3
2.2.3	Other Command Line Options	3
2.3	Shortcuts	3
3	The Menu Entries	6
3.1	The File Menu	6
3.2	The Edit Menu	7
3.3	The View menu	8
3.4	The Bookmarks Menu	9
3.5	The Tools Menu	10
3.6	The Settings Menu	12
3.7	The Help Menu	13
4	Configure KWrite	14
4.1	Appearance	14
4.2	Fonts & Colors	15
4.2.1	Colors	15
4.2.2	Fonts	16
4.2.3	Normal Text Styles	16
4.2.4	Highlighting Text Styles	16
4.3	Cursor & Selection	17

The KWrite Handbook

4.4	Editing	17
4.5	Indentation	19
4.6	Open & Save	20
4.7	Highlighting	21
4.8	Filetypes	21
4.9	Shortcuts	22
4.10	Plugins	22
5	Credits and Licenses	23

Abstract

KWrite is a text editor for KDE

Chapter 1

Introduction

KWrite is more than a text editor for the KDE Desktop. It is meant to be a programmer's editor, and could be considered as at least a partial alternative to more powerful editors. It may be best used in conjunction with Konqueror for source file browsing for different languages. KWrite also works very well as a simple text editor. One of KWrite's main features is the colorized syntax, customized for many different programming languages such as: C/C++, JavaTM, Python, Perl, Bash, Modula 2, HTML, and Ada.

Chapter 2

Some Fundamentals

KWrite is very simple to use. Anyone that has used a text editor should have no problems.

2.1 Drag and Drop

KWrite uses the KDE Drag and Drop protocol. Files may be dragged and dropped onto KWrite from the Desktop, Konqueror or some remote FTP site opened in one of Konqueror's windows.

2.2 Command Line Options

Though KWrite may most often be started from the KDE program menu, or a desktop icon, it can also be opened at the command line prompt of a terminal window. There are a few useful options that are available when doing this.

2.2.1 Specify a File

By specifying the path and name of a particular file the user can have KWrite open (or create) that file immediately upon startup. This option might look something like the following:

```
% kwrite  
/home/myhome/docs/myfile.txt
```

2.2.2 Specify a file on the internet

The above-mentioned method could even be used to open files on the internet (if the user has an active connection at the time.) An example of this might look like the following:

```
% kwrite  
ftp://ftp.kde.org/pub/kde/Welcome.msg
```

2.2.3 Other Command Line Options

The following command line help options are available

kwrite --help This lists the most basic options available at the command line.

kwrite --help-qt This lists the options available for changing the way KWrite interacts with Qt™.

kwrite --help-kde This lists the options available for changing the way KWrite interacts with KDE.

kwrite --help-all This lists all of the command line options.

kwrite --author Lists KWrite's authors in the terminal window

kwrite --version Lists version information for Qt™, KDE, and KWrite. Also available through **kwrite -v**

kwrite --stdin Causes KWrite to read the document content from STDIN. This is similar to the common option - used in many command line programs, and allows you to pipe command output into KWrite.

kwrite --encoding *encoding* *URL* Causes KWrite to use the specified encoding for the document.

kwrite --line *line* *URL* Navigate to the specified line after opening the document.

kwrite --column *column* *URL* Navigate to the specified column after opening the document.

2.3 Shortcuts

Many of the shortcuts are configurable by way of the [Settings](#) menu. By default KWrite honors the following shortcuts:

The KWrite Handbook

Insert	Toggle between Insert and Overwrite mode. When in insert mode the editor will add any typed characters to the text while pushing along any data to the right of the text cursor. Overwrite mode causes the entry of each character to eliminate the character immediately to the right of the text cursor.
Left Arrow	Move the cursor one character to the left
Right Arrow	Move the cursor one character to the right
Up Arrow	Move the cursor up one line
Down Arrow	Move the cursor down one line
Page Up	Move the cursor up one page
Alt+Page Up	Previous Bookmark
Page Down	Move the cursor down one page
Alt+Page Down	Next Bookmark
Backspace	Delete the character to the left of the cursor
Home	Move the cursor to the beginning of the line
End	Move the cursor to the end of the line
Delete	Delete the character to the right of the cursor (or any selected text)
Shift+Left Arrow	Mark text one character to the left
Shift+Right Arrow	Mark text one character to the right
F1	Help
Shift+F1	What's this?
F3	Find again
Shift+F3	Find Previous
Ctrl+A	Select All
Ctrl+Shift+A	Deselect
Ctrl+Shift+B	Block Selection Mode
Ctrl+B	Set a Bookmark
Ctrl+C	Copy the marked text to the clipboard.
Ctrl+D	Comment
Ctrl+Shift+D	Uncomment
Ctrl+F	Find
Ctrl+G	Go to line...

The KWrite Handbook

Ctrl+I	Indent selection
Ctrl+Shift+I	Unindent selection
Ctrl+J	Join Lines
Ctrl+N	New document
Ctrl+O	Open a document
Ctrl+P	Print
Ctrl+Q	Quit - close active copy of editor
Ctrl+R	Replace
Ctrl+S	Invokes the Save command.
Ctrl+U	Uppercase
Ctrl+Shift+U	Lowercase
Alt+Ctrl+U	Capitalize
Ctrl+V	Paste the clipboard text into line edit.
Ctrl+W	Close
Ctrl+X	Delete the marked text and copy it to the clipboard.
Ctrl+Z	Undo
Ctrl+Shift+Z	Redo
Ctrl+-	Collapse One Local Level
Ctrl+Shift+-	Collapse Toplevel
Ctrl++	Expand One Local Level
Ctrl+Shift++	Expand Toplevel
F5	Reload
F6	Show/Hide Icon Border
F7	Switch to Command Line
F9	Show/Hide Folding Markers
F10	Dynamic Word Wrap
F11	Show/Hide Line Numbers

Chapter 3

The Menu Entries

3.1 The File Menu

- File** → **New (Ctrl+N)** This starts a new Document in the editor. If there is a current document with unsaved changes the user is given a chance to save it.
- File** → **Open... (Ctrl+O)** Displays a standard KDE Open File dialog. Use the file view to select the file you want to open, and click on Open to open it. You can find more information about the KDE Open File dialog in the KDE User Guide.
- File** → **Open Recent** This is a shortcut to open recently saved documents. Clicking on this item opens a list to the side of the menu with several of the most recently saved files. Clicking on a specific file will open it in KWrite - if the file still resides at the same location.
- File** → **Save (Ctrl+S)** This saves the current document. If there has already been a save of the document then this will overwrite the previously saved file without asking for the user's consent. If it is the first save of a new document the save as dialog (described below) will be invoked.
- File** → **Save As...** This allows a document to be saved with a new file name. This is done by means of the file dialog box described above in the [Open](#) section of this help file.
- File** → **Reload (F5)** Reloads the active file from disk. This command is useful if another program or process has changed the file while you have it open in KWrite.
- File** → **Print... (Ctrl+P)** Opens a simple print dialog allowing the user to specify what, where, and how to print.
- File** → **Export as HTML...** Export your file in HTML format so your document can be viewed as a web page.

File → **Close (Ctrl+W)** Close the active file with this command. If you have made unsaved changes, you will be prompted to save the file before KWrite closes it.

File → **Quit (Ctrl+Q)** This will close the editor window, if you have more than one instance of KWrite running, through the New View or New Window menu items, those instances will not be closed.

3.2 The Edit Menu

Edit → **Undo (Ctrl+Z)** This is used to eliminate or reverse the most recent user action or operation.

Edit → **Redo (Ctrl+Shift+Z)** This will reverse the most recent change (if any) made using Undo

Edit → **Cut (Ctrl+X)** This command deletes the current selection and places it on the clipboard. The clipboard is a feature of KDE that works invisibly to provide a way to transfer data between applications.

Edit → **Copy (Ctrl+C)** This copies the currently selected text to the clipboard so that it may be pasted elsewhere. The clipboard is a feature of KDE that works invisibly to provide a way to transfer data between applications.

Edit → **Copy as HTML** This copies the currently selected text to the clipboard as HTML.

Edit → **Paste (Ctrl+V)** This will insert the contents of the clipboard at the cursor position. The clipboard is feature of KDE that works invisibly to provide a way to transfer data between applications.

Edit → **Select All (Ctrl+A)** This will select the entire document. This could be very useful for copying the entire file to another application.

Edit → **Deselect (Ctrl+Shift+A)** Deselects the selected text in the editor if any.

Edit → **Bloc Selection Mode (Ctrl+Shift+B)** Toggles Selection Mode. When the Selection Mode is "BLOCK", you can make vertical selections, ie select column 5 to 10 in lines 9 to 15. The status bar shows the current state of the Selection Mode, either "NORM" or "BLK".

Edit → **Overwrite Mode (Insert)** Toggles the Insert/Overwrite modes. When the mode is "INS", you insert characters where the cursor is. When the mode is "OVR", writing characters will replace the current characters if your cursor is positioned before any character. The status bar shows the current state of the Overwrite Mode, either "INS" or "OVR".

Edit → **Find (Ctrl+F)** This opens the find dialog which is used to specify the Text to Find in the document. There is small text box for entering the search pattern which also doubles as a dropdown box. Clicking on the dropdown arrow at the side of the box makes available other recent search

patterns. Other parameters are included to make the search more efficient. Selecting Case Sensitive will limit finds to entries that match the case (upper or lower) of each of the characters in the search pattern. Find Backwards directs the search to proceed in an upwardly direction. The Selected Text option keeps the search within currently selected text. Checking Whole Words Only prevents the search from stopping on words that contain the searched for pattern. The Search from Cursor option begins the search from the current position of the cursor within the document rather than from the beginning.

Edit → **Find Next (F3)** This repeats the last find operation, if any, without calling the find dialog box.

Edit → **Find Previous (Shift+F3)** This repeats the last find operation, if any, without calling the find dialog box, and searching backwards instead of forwards through the document.

Edit → **Replace... (Ctrl+R)** This command opens the replace dialog box. The replace dialog is almost identical to the above-mentioned find dialog. In addition to the features in the find dialog it contains a Replace With: text entry/dropdown box. Using this dialog the user can specify both the text to be found and text with which to replace it. The additional Prompt On Replace option allows the user to have KWrite ask for confirmation before each replacement.

Edit → **Go to line (Ctrl+G)** This opens the goto line dialog box which is used to have the cursor jump to a particular line (specified by number) in the document. The line number may be entered directly into the text box or graphically by clicking on the up or down arrow spin controls at the side of the text box. The little up arrow will increase the line number and the down arrow decrease it. There is also a slide control to the right of the text box which allows the user to move the goto point in the document in an analog manner.

3.3 The View menu

The View menu allows you to manage settings specific to the active editor, and to manage frames.

View → **New Window** Opens a new window with the same text.

View → **Switch to Command Line (F7)** Displays the Katepart command line at the bottom of the window. In the command line, type "help" to get help and "help list" to get a list of commands.

View → **Schema** Select a font schema.

View → **Dynamic Word Wrap (F10)** The text lines will be wrapped at the view border on the screen.

View → Dynamic Word Wrap Indicators (F10) Choose when and how the dynamic word wrap indicators should be displayed. This is only available if the Dynamic Word Wrap option is checked.

View → Show Static Word Wrap Marker If this option is checked, a vertical line will be drawn at the word wrap column as defined in the Settings → Configure Editor... in the Editing tab. Please note that the word wrap marker is only drawn if you use a fixed pitch font.

View → Show Icon Border (F6) This is a toggle item. Setting it on checked will make the side of the active editor, and vice versa.

View → Show Line Numbers (F11) This is a toggle Item. Setting it on checked will make a pane displaying the line numbers of the document visible in the left border of the active editor, and vice versa.

View → Show Scrollbar Marks If this option is checked, the view will show marks on the vertical scrollbar.

Hide Folding Markers (F9) If this option is checked, the marks for code folding will be hidden.

Code Folding

Collapse Toplevel (Ctrl+Shift+-) Collapse all toplevel regions in the document.

Expand Toplevel (Ctrl+Shift++) Expand all toplevel regions in the document.

Collapse One Local Level (Ctrl+-) Collapse the region closest to the cursor.

Expand One Local Level (Ctrl++) Expand the region closest to the cursor.

3.4 The Bookmarks Menu

Bookmarks → Set Bookmark (Ctrl+B) Sets or removes a bookmark in the current line of the active document. (If it's there, it is removed, otherwise one is set).

Bookmarks → Clear All Bookmarks This command will remove all the markers from the document as well as the list of markers which is appended at the bottom of this menu item.

Bookmarks → Previous (Alt+Page Up) This will move the cursor to beginning of the first above line with a bookmark. The menuitem text will include the line number and the first piece of text on the line. This item is only available when there is a bookmark in a line above the cursor.

Bookmarks → Next (Alt+Page Down) This will move the cursor to beginning of the next line with a bookmark. The menuitem text will include the line number and the first piece of text on the line. This item is only available when there is a bookmark in a line below the cursor.

At the bottom of this menu, a list of markers appears if any markers are available for this window.

3.5 The Tools Menu

Tools → **Read Only Mode** Set the current document to Read Only mode. This prevents any text addition and any changes in the document formatting.

Tools → **Filertype** Choose the filetype scheme you prefer for the active document. This overwrites the global filetype mode set in Settings → Configure Editor... in the Filetypes tab for your current document only.

Tools → **Highlighting** Choose the Highlighting scheme you prefer for the active document. This overwrites the global highlighting mode set in Settings → Configure Editor... for your current document only.

Tools → **Indentation** Choose the style of indentation you want for your active document. This overwrites the global indentation mode set in Settings → Configure Editor... for your current document only.

Tools → **Encoding** You can overwrite the default encoding set in Settings → Configure Editor... in the Open/Save tab to set a different encoding for your current document. The encoding you set here will be only valid for your current document.

Tools → **End of Line** Choose your preferred end of line mode for your active document. This overwrites the global end of line mode set in Settings → Configure Editor... for your current document only.

Tools → **Spelling...** This initiates the spellchecking program - a program designed to help the user catch and correct any spelling errors. Clicking on this entry will start the checker and bring up the speller dialog box through which the user can control the process. There are four settings lined up vertically in the center of the dialog with their corresponding labels just to the left. Starting at the top they are:

Unknown word: Here, the spellchecker indicates the word currently under consideration. This happens when the checker encounters a word not in its dictionary - a file containing a list of correctly spelled words against which it compares each word in the editor.

Replace with: If the checker has any similar words in its dictionary the first one will be listed here. The user can accept the suggestion, type in his or her own correction, or choose a different suggestion from the next box.

Suggested Words: The checker may list here a number of possible replacements for the word under consideration. Clicking on any one of the suggestions will cause that word to be entered in the Replacement: box, above.

Language: If you have installed multiple dictionaries, here you can select which dictionary/language should be used.

On the right side of the dialog box are 5 buttons that allow the user to control the spellcheck process. They are:

The KWrite Handbook

Add to Dictionary Pressing this button adds the word in the Misspelled Word: box to the checker's dictionary. This means that in the future the checker will always consider this word to be correctly spelled.

Replace This button has the checker replace the word under consideration in the document with the word in the Replacement: box.

Replace All This button causes the checker to replace not only the current Unknown word: but to automatically make the same substitution for any other occurrences of this Misspelled Word: in the document.

Ignore Activating this button will have the checker move on without making any changes.

Ignore All This button tells the checker to do nothing with the current Unknown word: and to pass over any other instances of the same word.

NOTE

This only applies to the current spellcheck run. If the checker is run again later it will stop on this same word.

Three more buttons are located horizontally along the bottom of the spellcheck dialog. They are:

Help This invokes the KDE help system starting at the KWrite help pages (this document).

Finished This button ends the spellcheck process, and returns to the document.

Cancel This button cancels the spellcheck process, all modifications are reverted, and you will return to your document.

Tools → **Spelling (from cursor)...** This initiates the spellchecking program but it starts where your cursor is instead of at the beginning of the document.

Tools → **Spellcheck Selection...** Spellchecks the current selection.

Tools → **Indent (Ctrl+I)** This increases the paragraph's indentation by one step. The size of the step depends on the [indentation settings](#).

Tools → **Unindent (Ctrl+Shift+I)** This reduces the paragraph's indentation by one step. The size of the step depends on the [indentation settings](#).

Tools → **Clean Indentation** This cleans the indentation for the current selection or for the line the cursor is currently in. Cleaning the indentation ensures that all your selected text follows the indentation mode you choose.

Tools → **Align** Causes a realign of the current line or selected lines using the indentation mode and indentation settings in the document.

Tools → **Comment (Ctrl+D)** This adds one space to the beginning of the line where the text cursor is located or to the beginning of any selected lines.

- Tools** → **Uncomment (Ctrl+Shift+D)** This removes one space (if any exist) from the beginning of the line where the text cursor is located or from the beginning of any selected lines.
- Tools** → **Uppercase (Ctrl+U)** Put the selected text or the letter after the cursor in uppercase.
- Tools** → **Lowercase (Ctrl+Shift+U)** Put the selected text or the letter after the cursor in lowercase.
- Tools** → **Capitalize (Alt+Ctrl+U)** Capitalize the selected text or the current word.
- Tools** → **Join Lines (Ctrl+J)** Joins the selected lines, or the current line and the line below with one white space character as a separator. Leading/trailing white space on joined lines is removed in the affected ends.
- Tools** → **Word Wrap Document** Apply static word wrapping on all the document. That means that a new line of text will automatically start when the current line exceeds the length specified by the Wrap words at: option in the Editing tab in Settings → Configure Editor...

3.6 The Settings Menu

- Settings** → **Show Toolbar** When checked, this displays a movable toolbar containing buttons used to initiate frequently used commands. When unchecked the toolbar is hidden.
- Settings** → **Show Statusbar** When checked, this displays a small bar at the bottom of the editor containing information about the status of the current document. When unchecked the statusbar is hidden.
- Settings** → **Show Path** When selected, this displays in the titlebar the path (its location in the file system) of the current document. When unchecked the path is hidden.
- Settings** → **Configure Editor...** This menu item opens a dialog whereby several different [settings](#) may be adjusted.
- Settings** → **Choose Editor...** Choose the default text editing component that you wish to use in KWrite. You can choose System Default, Embedded Advanced Text Editor or Qt™ Designer Based Text Editor (note that the Qt™ Designer Based Text Editor is only available if you have KDevelop installed on your system). If you choose System Default, KWrite will honor your changes in KDE Control Center. All other choices will override that setting.
- Settings** → **Configure Shortcuts...** This command opens a dialog box whereby the [shortcuts](#) may be changed. A display window at the top of the dialog box shows the list of commands (actions) that can have keyboard shortcuts. Below the display are three radio buttons. The user may choose

between No Key, Default Key, and Custom Key. (Note that a set of radio buttons only allows the selection of one of the offered items - in the way that buttons on a car radio only offer the selection of one preset station. Also, the Default Key selection is only available for those commands that actually have a 'default' shortcut.) Selecting the Custom Key option activates the three check boxes and key button at the bottom of the dialog. The user may then select a key combination for the command in question by means of the check boxes and key button. For example, with the About KDE command selected in the display window, the user could select **Ctrl** and **Alt**, click on the key button, and then press the **K** key on the keyboard. This would mean that anytime he or she held down the **Ctrl** and **Alt** buttons and pressed **K** (while using KWrite) the About KDE display box would be called.

Settings → **Configure Toolbars...** This will open the dialog whereby the toolbar configuration may be changed. The user can choose which shortcut buttons should appear on the toolbar. A display window on the left lists the commands available to be placed on the toolbar. A display on the right lists those commands already on the toolbar. A set of four arrow buttons between the two displays manipulates the selections. The right pointing arrow places any command selected in the left pane onto the right pane, i.e., it is added to the toolbar. The left arrow does just the opposite, removing any action selected in the right window from the toolbar. The up and down pointing arrows change the position of an action selected in the right window which changes the position of its button in the toolbar.

3.7 The Help Menu

Help → **KWrite Handbook (F1)** Invokes the KDE Help system starting at the KWrite help pages. (this document).

Help → **What's This? (Shift+F1)** Changes the mouse cursor to a combination arrow and question mark. Clicking on items within KWrite will open a help window (if one exists for the particular item) explaining the item's function.

Help → **Report Bug...** Opens the Bug report dialog where you can report a bug or request a 'wishlist' feature.

Help → **About KWrite** This will display version and author information.

Help → **About KDE** This displays the KDE version and other basic information.

Chapter 4

Configure KWrite

Selecting Settings → Configure Editor... from the menu brings up the Configure Editor dialog box. This dialog can be used to alter a number of different settings. The settings available for change vary according to which category the user chooses from a vertical list on the left side of the dialog. By means of three buttons along the bottom of the box the user can control the process.

She or he may invoke the Help system, accept the current settings and close the dialog by means of the OK button, or Cancel the process. The categories Colors, Fonts Indent, Select, Edit, Spelling and Highlighting are detailed below.

4.1 Appearance

Word Wrap

Dynamic word wrap If this option is checked, the text lines will be wrapped at the view border on the screen.

Dynamic word wrap indicators Choose when the Dynamic word wrap indicators should be displayed.

Vertically align dynamically wrapped lines to indentation depth: Enables the start of dynamically wrapped lines to be aligned vertically to the indentation level of the first line. This can help to make code and markup more readable.

Additionally, this allows you to set a maximum width of the screen, as a percentage, after which dynamically wrapped lines will no longer be vertically aligned. For example, at 50%, lines whose indentation levels are deeper than 50% of the width of the screen will not have vertical alignment applied to subsequent wrapped lines.

Code Folding

Show folding markers (if available) If this option is checked, the current view will display marks for code folding, if code folding is available.

Borders

Show icon border If this is checked, you will see an icon border on the left hand side. The icon border shows bookmark signs for instance.

Show line numbers If this is checked, you will see line numbers on the left hand side.

Show scrollbar marks If this option is checked the current view will show marks on the vertical scrollbar. These marks will for instance show bookmarks.

Sort Bookmarks Menu

By position The bookmarks will be ordered by the line numbers they are placed at.

By creation Each new bookmark will be added to the bottom, independently from where it is placed in the document.

Show indentation lines If this is checked, the editor will display vertical lines to help identifying indent lines.

4.2 Fonts & Colors

This section of the dialog lets you configure all fonts and colors in any color scheme you have, as well creating new schemes or deleting existing ones. Each scheme has settings for colors, fonts and normal and highlight text styles.

KWrite will preselect the currently active scheme for you, if you want to work on a different scheme start by selecting that from the Schema combobox.

4.2.1 Colors

Text Area Background

Normal text This is the default background for the editor area, it will be the dominant color on the editor area.

Selected Text This is the background for selected text. The default is the global selection color, as set in your KDE color preferences.

Current Line Set the color for the current line. Setting this a bit different from the Normal text background helps to keep focus on the current line.

Bookmark This combo lets you set overlay colors for various mark types. The color is mixed into the background color of a marked line, so that a line with more marks or a marked line that is current has a background that is a mix of more colors. The mark colors are also used if you enable display of scrollbar marks.

Other Elements

Left Border Background This color is used for the marks, line numbers and folding marker borders in the left side of the editor view when they are displayed.

Line Numbers This color is used to draw the line numbers on the left side of the view when displayed.

Bracket Highlight This color is used to draw the background of matching brackets.

Word Wrap Markers This color is used to draw a pattern to the left of dynamically wrapped lines when those are aligned vertically, as well as for the static word wrap marker.

Tab Markers This color is used to draw white space indicators when enabled.

4.2.2 Fonts

Here you can choose the font for the schema. You can choose from any font available on your system, and set a default size. A sample text displays at the bottom of the dialog, so you can see the effect of your choices.

4.2.3 Normal Text Styles

The normal text styles are inherited by the highlight text styles, allowing the editor to present text in a very consistent way, for example comment text is using the same style in almost all of the text formats that kate can highlight.

The name in the list of styles is using the style configured for the item, providing you with an immediate preview when configuring a style.

Each style lets you select common attributes as well as foreground and background colors. To unset a background color, rightclick to use the context menu.

4.2.4 Highlighting Text Styles

Here you can edit the text styles used by a specific highlight definition. The editor preselects the highlight used by your current document. To work on a different highlight, select one in the Highlight combobox above the style list.

The name in the list of styles is using the style configured for the item, providing you with an immediate preview when configuring a style.

Each style lets you select common attributes as well as foreground and background colors. To unset a background color, rightclick to use the context menu. In addition you can see if a style is equal to the default style used for the item, and set it to that if not.

You will notice that many highlights contain other highlights represented by groups in the style list. For example most highlights import the Alert highlight, and many source code formats imports the Doxygen highlight. Editing colors in those groups only affects the styles when used in the edited highlight format.

4.3 Cursor & Selection

Text Cursor Movement

Smart home When selected, pressing the home key will cause the cursor to skip white space and go to the start of a line's text.

Wrap cursor When on, moving the insertion cursor using the **Left** and **Right** keys will go on to previous/next line at beginning/end of the line, similar to most editors.

When off, the insertion cursor cannot be moved left of the line start, but it can be moved off the line end, which can be very handy for programmers. When this option is chosen, moving the cursor with the arrow keys off the end of a line (to the right) causes it to jump down to the beginning of the next line. Likewise when the cursor is moved past the beginning of a line (to the left) it jumps up to the end of the preceding line. When this option is not selected, moving the cursor right past the end of a line merely causes it to continue horizontally in the same line and trying to move it left past the beginning does nothing.

Page Up/Page Down moves cursor This option changes the behavior of the cursor when the user presses the **Page Up** or **Page Down** key. If unselected the text cursor will maintain its relative position within the visible text in KWrite as new text becomes visible as a result of the operation. So if the cursor is in the middle of the visible text when the operation occurs it will remain there (except when one reaches the beginning or end.) With this option selected, the first key press will cause the cursor to move to either the top or bottom of the visible text as a new page of text is displayed.

Autocenter cursor (lines): Sets the number of lines to maintain visible above and below the cursor when possible.

Selection Mode

Normal Selections will be overwritten by typed text and will be lost on cursor movement.

Persistent Selections will stay even after cursor movement and typing.

4.4 Editing

Tabulators

Insert spaces instead of tabulators When this is enabled the editor will insert a calculated number of spaces according to the position in the text and the `tab-width` setting when you press the **TAB** key.

Show tabulators When this is enabled KWrite will display a small dot as a visual representation of tabulator characters.

NOTE

This also causes dots to be drawn to indicate trailing white space. This will be fixed in a future version of KWrite

Tab Width If the [Replace Tabs By Spaces](#) option is selected this entry determines the number of spaces with which the editor will automatically replace tabs.

Static Word Wrap Word wrap is a feature that causes the editor to automatically start a new line of text and move (wrap) the cursor to the beginning of that new line. KWrite will automatically start a new line of text when the current line reaches the length specified by the [Wrap Words At:](#) option.

Enable static word wrap Turns static word wrap on or off.

Show static word wrap markers (if applicable) If this option is checked, a vertical line will be drawn at the word wrap column as defined in the Settings → Configure Editor... in the Editing tab. Please note that the word wrap marker is only drawn if you use a fixed pitch font.

Wrap words at: If the [Word Wrap](#) option is selected this entry determines the length (in characters) at which the editor will automatically start a new line.

Remove Trailing Spaces KWrite will automatically eliminate extra spaces at the ends of lines of text.

Auto Brackets When the user types a left bracket ([, (, or {) KWrite automatically enters the right bracket (],), or }) to the right of the cursor.

Maximum undo steps: Here the user may specify the number of steps KWrite will retain in memory for purposes of undoing entries and actions. This means that the higher the number of steps set the more memory KWrite will use for this. Setting this entry to 10 would mean that the user would be able to reverse the last ten operations, i.e. click the undo button 10 times and obtain results.

Smart search text from: This determines where KWrite will get the search text from (this will be automatically entered into the Find Text dialog):

- **Nowhere:** Don't guess the search text.
- **Selection Only:** Use the current text selection, if available.
- **Selection, then Current Word:** Use the current selection if available, otherwise use the current word.
- **Current Word Only:** Use the word that the cursor is currently resting on, if available.

- **Current Word, then Selection:** Use the current word if available, otherwise use the current selection.

Note that, in all the above modes, if a search string has not been or cannot be determined, then the Find Text Dialog will fall back to the last search text.

4.5 Indentation

Automatic indentation

Indentation mode: Select the automatic indentation mode you want to use as default. It is strongly recommended to use `None` or `Normal` here, and use filetype configurations to set other indentation modes for text formats like C/C++ code or XML.

Insert leading Doxygen "*" when typing Automatically insert a leading "*" while typing within a doxygen style comment. This setting is only enabled when applicable.

Indentation with Spaces

Use spaces instead of tabs to indent This replaces tabs with the number of spaces set in Number of spaces: below.

Emacs style mixed mode Use a mix of tabs and space characters for indentation.

Number of spaces: Set the number of spaces you want to use for indentation when you check Use spaces instead of tabs to indent above.

Keep Indent Profile When this is enabled, the editor will not unindent lines in a selection further when the line with the least indentation becomes unindented. If you sometimes unindent blocks of indented code, this may be helpful.

Keep Extra Spaces Indentations of more than the selected number of spaces will not be shortened.

Keys to use

Tab key indents This allows the tab key to be used to indent.

Backspace key indent This allows the backspace key to be used to indent.

Tab Key Mode if Nothing Selected

Insert indent characters This allows the Tab key insert indent characters.

Insert tab character This allows the Tab key insert a tab.

Indent current line This allows the Tab key indent the current line.

4.6 Open & Save

File Format

Encoding: This sets the default character encoding for your files.

End of line: Choose your preferred end of line mode for your active document. You have the choice between UNIX®, DOS/Windows® or Macintosh.

Automatic end of line detection Check this if you want the editor to autodetect the end of line type. The first found end of line type will be used for the whole file.

Memory Usage

Maximum loaded blocks per file: The editor will load given number of blocks (of around 2048 lines) of text into memory; if the filesize is bigger than this the other blocks are swapped to disk and loaded transparently as-needed.

This can cause little delays while navigating in the document; a larger block count increases the editing speed at the cost of memory.

For normal usage, just choose the highest possible block count: limit it only if you have problems with the memory usage.

Automatic Cleanups on Load/Save

Remove trailing spaces The editor will automatically eliminate extra spaces at the ends of lines of text while loading/saving the file.

Folder Config File

Search depth for config file: The editor will search the given number of folder levels upwards for KWrite config file and load the settings line from it.

Backup on Save Backing up on save will cause KWrite to copy the disk file to <prefix><filename><suffix>' before saving changes. The suffix defaults to `~` and prefix is empty by default.

Local files Check this if you want backups of local files when saving.

Remote files Check this if you want backups of remote files when saving.

Prefix Enter the prefix to prepend to the backup file names.

Suffix Enter the suffix to add to the backup file names.

4.7 Highlighting

This group of options is used to customize the highlighting styles for each programming language type. Any changes you made in other areas of this dialog apply only to this type.

Highlight: This is used to choose the language type to configure.

Informations View the properties of the chosen language highlighting rules: author name and license.

Properties

File extensions: This is the list of file extensions used to determine which files to highlight using the current syntax highlight mode.

MIME types: Clicking the wizard button will display a dialog with a list of all available mime types to choose from.

The **File Extensions** entry will automatically be edited as well.

Priority: Set the priority of the highlight rule.

Download... Click this button to download new or updated syntax highlight descriptions from the Kate website.

4.8 Filetypes

This page allows you to override the default configuration for documents of specified mimetypes. When the editor loads a document, it will try if it matches the file masks or mimetypes for one of the defined filetypes, and if so apply the variables defined. If more filetypes match, the one with the highest priority will be used.

Filetype: The filetype with the highest priority is the one displayed in the first drop down box. If more filetypes were found, they are also listed.

New This is used to create a new filetype. After you click on this button, the fields below get empty and you can fill the properties you want for the new filetype.

Delete To remove an existing filetype, select it from the drop down box and press the Delete button.

Properties of current filetype The filetype with the highest priority is the one displayed in the first drop down box. If more filetypes were found, they are also listed.

Name: The name of the filetype will be the text of the corresponding menu item. This name is displayed in the Tools → Filetypes

Section: The section name is used to organize the file types in menus. This is also used in the Tools → Filetypes menu.

Variables: This string allows you to configure KWrite's settings for the files selected by this mimetype using KWrite variables. You can set almost any configuration option, such as highlight, indent-mode, encoding, etc.

For a full list of known variables, see the manual.

File extensions: The wildcards mask allows you to select files by file-name. A typical mask uses an asterisk and the file extension, for example *.txt; *.text. The string is a semicolon-separated list of masks.

MIME types: Displays a wizard that helps you easily select mimetypes.

Priority: Sets a priority for this file type. If more than one file type selects the same file, the one with the highest priority will be used.

4.9 Shortcuts

You can change here the shortcut keys configuration. Select an action and click on Custom if you want a different shortcut for this action.

The search line allows you to look for a specific action and see its associated shortcut.

4.10 Plugins

This tab lists all available plugins and you can check those you want to use. Once a plugin is checked, the Configure button is enabled and you can click it in order to configure the highlighted plugin.

Chapter 5

Credits and Licenses

KWrite Copyright 2001 by the Kate team.

Based on the original KWrite, which was Copyright 2000 by Jochen Wilhelmy
digisnap@cs.tu-berlin.de

Contributions:

- Christoph Cullmann cullmann@kde.org
- Michael Bartl michael.bartl1@chello.at
- Phlip phlip_cpp@my-deja.com
- Anders Lund anders@alweb.dk
- Matt Newell newellm@proaxis.com
- Joseph Wenninger kde@jowenn.at
- Jochen Wilhely digisnap@cs.tu-berlin.de
- Michael Koch koch@kde.org
- Christian Gebauer gebauer@kde.org
- Simon Hausmann hausmann@kde.org
- Glen Parker glenebob@nwlinc.com
- Scott Manson sdmanson@altel.net
- John Firebaugh jfirebaugh@kde.org

Original documentation by Thad McGinnis ctmcginnis@compuserve.com

Updated by Anne-Marie Mahfouf annma@kde.org and Anders Lund anders@alweb.dk

This version of the KWrite Handbook is based on the original by Christian
Tibirna tibirna@kde.org

The KWrite Handbook

Converted to docbook/proofreading by Lauri Watts lauri@kde.org

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

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