## NAME

textedit – XView-based text editor with mouse support

## SYNOPSIS

**textedit** [ *generic-tool-arguments* ] [ **–Ei on | off** ] [ **–auto_indent** ] [ **–Eo on | off** ]
[ **–okay_to_overwrite** ] [ **–Er on | off** ] [ **–read_only** ] [ **–Ec** *N* ]
[ **–checkpoint** *count* ] [ **–EL** *lines* ] [ **–lower_context** *lines* ] [ **–Em** *pixels* ]
[ **–margin** *pixels* ] [ **–En** *N* ] [ **–number_of_lines** *lines* ] [ **–ES** *N* ]
[ **–multi_click_space** *radius* ] [ **–Et** *N* ] [ **–tab_width** *tabstop* ] [ **–ET** *N* ]
[ **–multi_click_timeout** *intrvl* ] [ **–Eu** *N* ] [ **–history_limit** *max* ] [ **–EU** *N* ]
[ **–upper_context** *lines* ] *filename*

## DESCRIPTION

**textedit** is a window-based XView application that provides both mouse and pointer operations and keyboard accelerators for the editing of text. In general, text editing throughout the OpenWindows user environment uses **textedit** conventions, both in text display areas such as mail message windows and command panel text fields.

**textedit** operates via a set of command panel buttons and text fields and a set of menus and submenus accessible from the main editing window.

## OPTIONS

*generic-tool-arguments*
: **textedit** accepts the XView generic tool arguments described in the **xview**(7) man page with the exception of the arguments for setting the frame label.

**–Ei on | off**

**–auto_indent**
: Choose whether or not to automatically indent newly-opened lines. The default is off. Corresponds to the **auto_indent** Text default.

**–Eo on | off**
**–okay_to_overwrite**
: Set behavior to the **Store as New File** menu item. If **on** a **Store as New File** to the current file is treated as a **Save Current File**. If **off** (the standard default), **Store as New File** operations using the current filename results in an error message. Corresponds to **Store_self_is_save**.

**–Er on | off**
**–read_only**
: Turn read-only mode on or off. When on, text cannot be modified.

**–Ec** *N*
**–checkpoint** *count*
: Checkpoint after every *count* editing operation. If *count* is 0 (the standard default), no checkpointing takes place. Each character typed, each **Paste**, and each **Cut** counts as an editing operation. Corresponds to **checkpoint_frequency**.

**–EL** *lines*
**–lower_context** *lines*
: Specify the minimum number of lines to keep between the caret and the bottom of the text subwindow. The default is 2. Corresponds to **lower_context**.

–**Em** *pixels*
–**margin** *pixels*
Set the scrollbar margin width in pixels. The default is 4. Corresponds to **left_margin**.

–**En** *N*
–**number_of_lines** *lines*
Set the number of lines in the bottom subwindow. The default is 45.

–**ES** *N*
–**multi_click_space** *radius*
Set the radius in pixels, within which clicks must occur to be treated as a multi-click selection. The default is 3 pixels. Corresponds to **multi_click_space**.

–**Et** *N*
–**tab_width** *tabstop*
Set the number of SPACE characters displayed per TAB stop. The default is 8. This option has no effect on the characters in the file. Corresponds to **tab_width**.

–**ET** *N*
–**multi_click_timeout** *intrvl*
Set the interval, in milliseconds, within which any two clicks must occur to be treated as a multi-click selection. The default is 390 milliseconds. Corresponds to **multi_click_timeout**.

–**Eu** *N*
–**history_limit** *max*
Set the maximum number of editing operations that can be undone or replayed. The default is 50. Corresponds to **history_limit**.

–**EU N**
–**upper_context** *lines*
Set the minimum number of lines to keep between the caret and the top of the text subwindow. The default is 2. Corresponds to **upper_context**.

## USAGE

**textedit** is part of the OpenWindows user environment.

### Signal Processing

If **textedit** hangs, for whatever reason, you can send a **SIGHUP** signal to its process ID, which forces it to write any changes (if possible):

**kill** –**HUP** *pid*

The edits are written to the file **textedit.***pid* in its working directory. If that fails, **textedit** successively tries to write to a file by that name in **/var/tmp**, and then **/tmp**. In addition, whenever **textedit** catches a fatal signal, such as **SIGILL**, it tries to write out the edits before aborting.

### Defaults Options

There are several dozen user-specified defaults that affect the behavior of the text-based facilities. See **xview**(7) for a complete description. Important defaults entries in the **Text** category are:

**Checkpoint_frequency**

If set to **0** (the standard default) no checkpointing is done. For any value greater than zero, a checkpoint is made each time the indicated number of editing operations has been performed since the last checkpoint. Each character typed, each **Paste**, and each **Cut** counts as an editing operation. The checkpoint file has a name of the form: *filename*%%, where *filename* is the name of the file being edited.

**Selections**

Selections in **textedit** are defined as selected portions of text to which editing operations can be applied. For example, a selection can be *deleted, moved, copied,* etc.

**textedit** provides two types of selections: primary and secondary. Primary selections allow you to select a set of text on which to perform an editing function. Secondary selections allow you to define a *second* block of text without undefining your primary text selection or repositioning your cursor. Being able to define two sets of text at once allows you to take advantage of the advanced editing functions described below in the section called **Function Keys.**

**Using a Mouse and Pointer:**

Single characters can be selected using the SELECT mouse button.

Blocks of text can be selected by selecting a starting point with the SELECT mouse button and selecting an ending point with the ADJUST button.

Or blocks of text can be selected using OPEN LOOK's *wipe through* feature by pointing at a beginning character and depressing the SELECT button while moving the pointer to the end of a block of text.

Selections can also be made by *clicking* (rapidly pressing) the SELECT button. Click once to select a single letter; click twice to select a word; click three times to select a complete line of text; click four times to select the entire document being edited.

**Visual Feedback**

All primary selections are indicated visually by inverse video of the text selected and are *pending delete.* Pending delete selections are replaced if you type or paste while the text is selected.

Secondary selections that are not *pending delete* are indicated by underlining of the text.

Secondary selections pending delete are indicated by underlining of the text and strike through of the individual characters.

**Secondary Selections**

Secondary selections are made using any of the selection methods described above in addition to holding down one of the four function keys corresponding to the commands **Cut**, **Find**, **Paste**, or **Copy**.

Secondary selections are made pending delete by holding the CTRL key when making the secondary selection. If a secondary selection is pending-delete, it is deleted when the function key is released, except in the case of the **Find**, which deselects the secondary selection.

Commands that operate on the primary selection do so even if the primary selection is not in the window that issued the command.

**Inserting Text and Command Characters**

For the most part, typing any of the standard keys either inserts the corresponding character at the insertion point, or erases characters. However, certain key combinations are treated as commands. Some of the most useful are:

| Command | Character | Description |
|---|---|---|
| **Cut-Primary** | META-X | Erases the primary selection and moves it to the Clipboard. |
| **Find-Primary** | META-F | Searches the text for the pattern specified by the primary selection or by the Clipboard, if there is no primary selection. |
| **Copy-to-Clipboard** | META-C | Copies the primary selection to the Clipboard. |
| **Paste-Clipboard** | META-V | Inserts the Clipboard contents at the insertion point. |
| **Copy-then-Paste** | META-P | Copies the primary selection to the insertion point (through the Clipboard). |
| **Go-to-EOF** | CTRL-RETURN | Moves the insertion point to the end of the text and positions the text so the insertion point is visible. |

**Keyboard Functions**

The commands indicated by use of the function keys are:

| Command | SPARC | x86 | Description |
|---|---|---|---|
| **Stop** | L1 | Esc | Aborts the current command. |
| **Again** | L2 | Meta-a | Repeats the previous editing sequence since a primary selection was made. |
| **Undo** | L4 | Meta-z | Undoes a prior editing sequence. |
| **Front** | L5 | Meta-r | Makes the window completely visible (or hides it, if it is already exposed). |
| **Copy** | L6 | Meta-c | Copies the primary selection, either to the Clipboard or at the closest end of the secondary selection. |
| **Open** | L7 | Meta-w | Makes the window iconic (or normal, if it is already iconic). |
| **Paste** | L8 | Meta-v | Copies either the secondary selection or the Clipboard at the insertion point. |
| **Find** | L9 | Meta-f | Searches for the pattern specified by, in order, the secondary selection, the primary selection, or the Clipboard. |
| **Cut** | L10 | Meta-x | Erases either the primary or the secondary selection and moves it to the Clipboard. |
| **Help** | F1 | F1 | Produces help text. |

**Find** usually searches the text forwards, towards the end. Holding down the SHIFT key while invoking **Find** searches backward through the text, towards the beginning. If the pattern is not found before the search encounters either extreme, it “wraps around” and continues from the other extreme. **Find** starts the search at the appropriate end of the primary selection, if the primary selection is in the subwindow that the search is made in; otherwise it starts at the insertion point, unless the subwindow cannot be edited, in which case it starts at the beginning of the text.

CTRL-Find invokes the **Find and Replace** pop-up frame.

**Menu Items**

**File** A pull-right menu item for file operations.

**Edit** A pull-right menu item equivalent of the editing function keys. The **Edit** submenu provides **Again**, **Undo**, **Copy**, **Paste**, and **Cut** (same as function keys L2, L4, L6, L8, and L10).

**Display** A pull-right menu item for controlling the way text is displayed and line display format.

**Find** A pull-right menu item for find and delimiter matching operations.

**Extras** A user definable pull-right menu item. The **Extras** standard submenu is controlled by **$OPENWINHOME/lib/locale/<locale>/xview/.text_extras_menu** This file has the same syntax as **.openwin-menu** file. See the **xview** man page.

Only those items that are active appear as normal text in the menu; inactive items (which are inappropriate at the time) are “grayed out”.

**User Defined Commands**

The file **$OPENWINHOME/lib/locale/<locale>/xview/.text_extras_menu** specifies filter programs that are included in the text subwindow **Extras** pull-right menu item. The file **~/.textswrc** specifies filter programs that are assigned to (available) function keys. These filters are applied to the contents of the primary selection. Their output is entered at the caret.

The file **$OPENWINHOME/share/src/xview/examples/textsw/textswrc** is a sample containing a set of useful filters. It is not read automatically.

## FILES

**~/.textswrc** specifies bindings of filters to function keys

**$OPENWINHOME/lib/locale/<locale>/xview/.text_extras_menu**
specifies bindings of filters for the extras menu pull-right items

**$OPENWINHOME/share/src/xview/examples/textsw/textswrc**
contains useful filters, including **shift_lines** and **capitalize**.

*filename***%** prior version of *filename* is available here after a **Save Current File** menu operation

**textedit.***pid* edited version of *filename*; generated in response to fatal internal errors

**/tmp/Text∗** editing session logs

## SEE ALSO

**kill**(1), **xview**(7)
*Solaris User’s Guide*
"About Text Editor" in the Help Handbook available through the Help option on the Workspace menu.

## DIAGNOSTICS

**Cannot open file '*filename*', aborting!** *filename* does not exist or cannot be read.

**textedit** produces the following exit status codes:

| | |
|---|---|
| 0 | normal termination |
| 1 | standard OpenWindows help message was printed |
| 2 | help message was requested and printed |
| 3 | abnormal termination in response to a signal, usually due to an internal error |
| 4 | abnormal termination during initialization, usually due to a missing file or running out of swap space |

## BUGS

Multi-click to change the current selection does not work for **Adjust Selection**.

Handling of long lines is incorrect in certain scrolling situations.

There is no way to replay any editing sequence except the most recent.