

CWind

manpage

NAME

cwind - keyboard and mouse simulation

SYNOPSIS

package require cwind ?1.5.1?

::cwind::command ?args ...?

DESCRIPTION

::cwind::wlist ?-exact? ?window-title?

::cwind::show ?-exact? window-title

::cwind::waitwind window-title ?seconds?

::cwind::gettext

::cwind::top ?-exact? window-title

::cwind::bottom ?-exact? window-title

::cwind::topmost ?-exact? window-title

::cwind::notopmost ?-exact? window-title

::cwind::getrect

::cwind::getwrect

::cwind::getpos

::cwind::setpos coordinates

::cwind::getwpos

::cwind::setwpos coordinates

::cwind::minimize

::cwind::maximize

::cwind::restore

::cwind::minimizeAll

::cwind::state ?-exact? window-title

::cwind::lclick

::cwind::rclick

::cwind::mclick

::cwind::dragl coordinates

::cwind::dragm coordinates

::cwind::dragr coordinates

::cwind::wdragl coordinates

::cwind::wdragm coordinates

::cwind::wdragr coordinates

::cwind::send text ...

::cwind::sendmessage command ?wparam? ?lparam?

::cwind::postmessage command ?wparam? ?lparam?

[::cwind::getcb](#)
[::cwind::putcb *string*](#)
[::cwind::sleep *?milliseconds?*](#)
[::cwind::version](#)

SPECIAL CHARACTERS

WINDOW MANAGER COMMANDS

REQUIREMENTS

INSTALLATION

LICENSE

NAME

cwind - keyboard and mouse simulation

SYNOPSIS

package require cwind ?1.5.1?

::cwind::*command* ?args ...?

DESCRIPTION

The CWind package allows one to simulate keyboard and mouse interactions on windows. Keystrokes, mouse clicks and window manager commands can be sent to arbitrary windows.

::cwind::wlist ?-exact? ?*window-title*?

If *window-title* (a glob pattern) is omitted, return a list of all existing window titles.

If *window-title* is specified, return a list of all the MDI child window names. It appears (2017) that there are very few applications using MDI and the technology is falling out of favor although [PhotoFiltre](#) and [Paint.NET](#) make use of it.

::cwind::show ?-exact? *window-title*

Search for a window with the given title (a glob pattern) and make it the active window.

The *window-title* must be a proper Tcl list of one or two elements. The first element is a parent window title and the optional second element is a MDI child window name. It appears (2017) that there are very few applications

using MDI and the technology is falling out of favor although [PhotoFiltre](#) and [Paint.NET](#) make use of it.

::cwind::waitwind *window-title ?seconds?*

Wait *seconds* (default 5) for a window with the given title (a glob pattern) to become the active window.

::cwind::gettext

Return the active window's title.

::cwind::top *?-exact? window-title*

Places the window with the given title (a glob pattern) at the top of the Z order. Refer to the microsoft documentation of the *SetWindowPos()* function for a full description.

::cwind::bottom *?-exact? window-title*

Places the window with the given title (a glob pattern) at the bottom of the Z order. Refer to the microsoft documentation of the *SetWindowPos()* function for a full description.

::cwind::topmost *?-exact? window-title*

Places the window with the given title (a glob pattern) above all non-topmost windows. The window maintains its topmost position even when it is deactivated. Refer to the microsoft documentation of the *SetWindowPos()* function for a full description.

::cwind::notopmost *?-exact? window-title*

Places the window with the given title (a glob pattern) above all non-topmost windows (that is, behind all topmost windows). Refer to the microsoft documentation of the *SetWindowPos()* function for a full description.

::cwind::getrect

Return a four-element list containing the coordinates of the client rectangle for the active window. This is effectively the size of the client window since the first two coordinates are always *0 0*.

::cwind::getwrect

Return a four-element list containing the screen coordinates of the active window

::cwind::getpos

Return a list containing the current mouse coordinates on the screen.

::cwind::setpos *coordinates*

Set the mouse position relative to the origin {0 0} at the upper left corner on the screen to *coordinates* (a list of two elements).

::cwind::getwpos

Return a list containing the current mouse coordinates on the screen relative to the active window.

::cwind::setwpos *coordinates*

Set the mouse position in the active window relative to the origin {0 0} at the upper left corner to *coordinates* (a list of two elements).

::cwind::minimize

Minimize the active window.

::cwind::maximize

Maximize the active window.

::cwind::restore

Restore the active window to its original size.

::cwind::minimizeAll

Minimize all the windows.

::cwind::state ?-exact? *window-title*

Return the status of the window with the given title (a glob pattern): Maximized, Minimized or Normal.

The *window-title* must be a proper Tcl list of one or two elements. The first element is a parent window title and the optional second element is a MDI (Multiple-Document Interface) child window name. It appears (2017) that there are very few applications using MDI and the technology is falling out of favor although [PhotoFiltre](#) and [Paint.NET](#) make use of it.

::cwind::lclick

Send a left mouse click to the active window.

::cwind::rclick

Send a right mouse click to the active window.

::cwind::mclick

Send a middle mouse click to the active window.

::cwind::dragl *coordinates*

Drag the mouse with the left-button to screen *coordinates* (a list of two elements).

::cwind::dragm *coordinates*

Drag the mouse with the middle-button to screen *coordinates* (a list of two elements).

::cwind::dragr *coordinates*

Drag the mouse with the right-button to screen *coordinates* (a list of two elements).

::cwind::wdragl *coordinates*

Drag the mouse with the left-button to *coordinates* (a list of two elements) within the active window.

::cwind::wdragm *coordinates*

Drag the mouse with the middle-button to *coordinates* (a list of two elements) within the active window.

::cwind::wdragr *coordinates*

Drag the mouse with the right-button to *coordinates* (a list of two elements) within the active window.

::cwind::send *text ...*

Send each text argument to the active window.

::cwind::sendmessage *command ?wparam? ?lparam?*

Send a windows manager *command* (either the integer value or name – see [WINDOW MANAGER COMMANDS](#)) to the active window via SendMessage. Wparam and lparam are additional integer values required by the *command*.

::cwind::postmessage *command ?wparam? ?lparam?*

Send a windows manager *command* (either the integer value or name – see [WINDOW MANAGER COMMANDS](#)) to the active window via PostMessage. Wparam and lparam are additional integer values required by the *command*.

::cwind::getcb

Return the text content of the clipboard.

::cwind::putcb *text*

Put text into the clipboard.

::cwind::sleeptime *?milliseconds?*

Return or set the *sleep time* which is the number of milliseconds to delay before sending each character. The default is zero.

::cwind::version

Return a string that identifies the Cwind version, the architecture for which it was built (32- or 64-bit), the date and time compiled and the GNU C compiler version.

SPECIAL CHARACTERS

Special characters may be included with sent text. For example ctrl+c, ctrl+o, PF4, ...

To distinguish a normal string from a special character, enclose it between pipes (the character '|').

For example, to send ctrl+c, use the following: |CTRL+| c |CTRL-|. This specifies that you want to hold the CTRL key, type 'c', and then release CTRL.

Since this may not be common knowledge, note that sending **[SNAP]** (Print Screen) will copy the entire screen into the clipboard while **[ALT+][SNAP][ALT-]** (Alt-Print Screen) restricts the copy to the active window.

SPECIAL CHARACTERS

ALT+	press ALT	PGUP	the Page Up key
ALT-	release ALT	PGDN	the Page Down key
ALTL+	press left ALT key	HOME	the Home key
ALTL-	release left ALT key	END	the End key
ALTR+	press right ALT key	SNAP	the Print Screen key
ALTR-	release right ALT key		
		F1 - F12	Function keys 1-12
CTRL+	press CTRL		
CTRL-	release CTRL	NUM0 - NUM9	Digits 0-9 on the numeric key pad
CTRLL+	press left CTRL key		
CTRLL-	release left CTRL key	NUM*	'*' on the numeric keypad
CTRLR+	press right CTRL key	NUM/	'/' on the numeric keypad
CTRLR-	release right CTRL key	NUM+	'+' on the numeric keypad
		NUM-	'-' on the numeric keypad
		NUM.	'.' on the numeric keypad
SHIFT+	press SHIFT	NUM#	synonym for NUM.
SHIFT-	release SHIFT		
SHIFTL+	press left SHIFT key	NUM,	apparently the keypad Enter key
SHIFTL-	release left SHIFT key		
SHIFTR+	press right SHIFT key	SEL	the menu select key (to the right of the space bar)
SHIFTR-	release right SHIFT key	LWIN	the <i>windows</i> key (to the left of the space bar)
		RWIN	the <i>windows</i> key (to the right of the space bar)
TAB	the Tab key		
RET	the Return key	KANJI	the kanji key (Japanese keyboard)
ESC	the Escape key	KANA	Japanese keyboard only
SPACE	the Space key	HANGUL	Japanese keyboard only
		JUNJA	Japanese keyboard only
BACK	the Backspace key	FINAL	Japanese keyboard only
DEL	the Delete key	HANJA	Japanese keyboard only
INS	the Insert key	CONV	the IME convert key
HELP	the help key	NCONV	the IME nonconvert key
		ACCEPT	the IME accept key
LEFT	arrow left	CLR	Unknown key
RIGHT	arrow right	PRINT	Unknown key
UP	arrow up	EXEC	Unknown key
DOWN	arrow down	APPS	Unknown key

WINDOW MANAGER COMMANDS

For a comprehensive list, see the **winuser.h** header file and the microsoft documentation.

WM_NULL	WM_NCPAINT	WM_XBUTTONDOWNBLCLK
WM_CREATE	WM_NCACTIVATE	WM_MOUSELAST
WM_DESTROY	WM_GETDLGCODE	WM_PARENTNOTIFY
WM_MOVE	WM_SYNCPAINT	WM_ENTERMENULOOP
WM_SIZE	WM_NCMOUSEMOVE	WM_EXITMENULOOP
WM_ACTIVATE	WM_NCLBUTTONDOWN	WM_NEXTMENU
WM_SETFOCUS	WM_NCLBUTTONUP	WM_SIZING
WM_KILLFOCUS	WM_NCLBUTTONDOWNBLCLK	WM_CAPTURECHANGED
WM_ENABLE	WM_NCRBUTTONDOWN	WM_MOVING
WM_SETREDRAW	WM_NCRBUTTONUP	WM_POWERBROADCAST
WM_SETTEXT	WM_NCRBUTTONDOWNBLCLK	WM_DEVICECHANGE
WM_GETTEXT	WM_NCMBUTTONDOWN	WM_MDICREATE
WM_GETTEXTLENGTH	WM_NCMBUTTONUP	WM_MDIDESTROY
WM_PAINT	WM_NCMBUTTONDOWNBLCLK	WM_MDIACTIVATE
WM_CLOSE	WM_NCXBUTTONDOWN	WM_MDIRESTORE
WM_QUERYENDSESSION	WM_NCXBUTTONUP	WM_MDINEXT
WM_QUERYOPEN	WM_NCXBUTTONDOWNBLCLK	WM_MDIMAXIMIZE
WM_ENDSESSION	WM_INPUT_DEVICE_CHANGE	WM_MDITILE
WM_QUIT	WM_INPUT	WM_MDICASCADE
WM_ERASEBKGD	WM_KEYFIRST	WM_MDIICONARRANGE
WM_SYSCOLORCHANGE	WM_KEYDOWN	WM_MDIGETACTIVE
WM_SHOWWINDOW	WM_KEYUP	WM_MDISETMENU
WM_WININCHANGE	WM_CHAR	WM_ENTERSIZEMOVE
WM_SETTINGCHANGE	WM_DEADCHAR	WM_EXITSIZEMOVE
WM_DEVMODECHANGE	WM_SYSKEYDOWN	WM_DROPFILES
WM_ACTIVATEAPP	WM_SYSKEYUP	WM_MDIREFRESHMENU
WM_FONTCHANGE	WM_SYSCHAR	WM_IME_SETCONTEXT
WM_TIMECHANGE	WM_SYSDEADCHAR	WM_IME_NOTIFY
WM_CANCELMODE	WM_UNICHAR	WM_IME_CONTROL
WM_SETCURSOR	WM_KEYLAST	WM_IME_COMPOSITIONFULL
WM_MOUSEACTIVATE	WM_IME_STARTCOMPOSITION	WM_IME_SELECT
WM_CHILDACTIVATE	WM_IME_ENDCOMPOSITION	WM_IME_CHAR
WM_QUEUESYNC	WM_IME_COMPOSITION	WM_IME_REQUEST
WM_GETMINMAXINFO	WM_IME_KEYLAST	WM_IME_KEYDOWN

WM_PAINTICON	WM_INITDIALOG	WM_IME_KEYUP
WM_ICONERASEBKGD	WM_COMMAND	WM_MOUSEHOVER
WM_NEXTDLGCTL	WM_SYSCOMMAND	WM_MOUSELEAVE
WM_SPOOLERSTATUS	WM_TIMER	WM_NCMOUSEHOVER
WM_DRAWITEM	WM_HSCROLL	WM_NCMOUSELEAVE
WM_MEASUREITEM	WM_VSCROLL	WM_WTSSESSION_CHANGE
WM_DELETEITEM	WM_INITMENU	WM_TABLET_FIRST
WM_VKEYTOITEM	WM_INITMENUPOPUP	WM_TABLET_LAST
WM_CHARTOITEM	WM_MENUSELECT	WM_CUT
WM_SETFONT	WM_MENUCHAR	WM_COPY
WM_GETFONT	WM_ENTERIDLE	WM_PASTE
WM_SETHOTKEY	WM_MENURBUTTONUP	WM_CLEAR
WM_GETHOTKEY	WM_MENUDRAG	WM_UNDO
WM_QUERYDRAGICON	WM_MENUGETOBJECT	WM_RENDERFORMAT
WM_COMPAREITEM	WM_UNINITMENUPOPUP	WM_RENDERALLFORMATS
WM_GETOBJECT	WM_MENUCOMMAND	WM_DESTROYCLIPBOARD
WM_COMPACTING	WM_CHANGEUISTATE	WM_DRAWCLIPBOARD
WM_COMMNOTIFY	WM_UPDATEUISTATE	WM_PAINTCLIPBOARD
WM_WINDOWPOSCHANGING	WM_QUERYUISTATE	WM_VSCROLLCLIPBOARD
WM_WINDOWPOSCHANGED	WM_CTLCOLOORMSGBOX	WM_SIZECLIPBOARD
WM_POWER	WM_CTLCOLOREDIT	WM_ASKCBFORMATNAME
WM_COPYDATA	WM_CTLCOLORLISTBOX	WM_CHANGECHAIN
WM_CANCELJOURNAL	WM_CTLCOLORBTN	WM_HSCROLLCLIPBOARD
WM_NOTIFY	WM_CTLCOLORDLG	WM_QUERYNEWPALETTE
WM_INPUTLANGCHANGEREQUEST	WM_CTLCOLORSCROLLBAR	WM_PALETTEISCHANGING
WM_INPUTLANGCHANGE	WM_CTLCOLORSTATIC	WM_PALETTECHANGED
WM_TCARD	WM_MOUSEFIRST	WM_HOTKEY
WM_HELP	WM_MOUSEMOVE	WM_PRINT
WM_USERCHANGED	WM_LBUTTONDOWN	WM_PRINTCLIENT
WM_NOTIFYFORMAT	WM_LBUTTONUP	WM_APPCOMMAND
WM_CONTEXTMENU	WM_LBUTTONDBLCLK	WM_THEMECHANGED
WM_STYLECHANGING	WM_RBUTTONDOWN	WM_CLIPBOARDUPDATE
WM_STYLECHANGED	WM_RBUTTONUP	WM_HANDHELDFIRST
WM_DISPLAYCHANGE	WM_RBUTTONDBLCLK	WM_HANDHELDLAST
WM_GETICON	WM_MBUTTONDOWN	WM_AFXFIRST
WM_SETICON	WM_MBUTTONUP	WM_AFXLAST
WM_NCCREATE	WM_MBUTTONDBLCLK	WM_PENWINFIRST
WM_NCDESTROY	WM_MOUSEWHEEL	WM_PENWINLAST
WM_NCCALCSIZE	WM_XBUTTONDOWN	WM_APP
WM_NCHITTEST	WM_XBUTTONUP	WM_USER

REQUIREMENTS

CWind requires Tcl 8.1 or higher.

The CWind package is available for Windows 95/98/Me/NT4.0/2000/+

For windows versions prior to vista, you can download the 32-bit version 1.4 from the project home page or the archived ZIP from this site. You might also be able to compile the original source with the appropriate microsoft sdk.

For windows versions vista and beyond, either download the appropriate pre-built binary from this repository or compile it from source. The included **src/GNUMakefile** was tested with [MSYS2](#)'s 32- and 64-bit gcc.

Note that I am only able to test the compiled version on a virtual windows XP system and a windows 7 64-bit system. Furthermore, the test is reduced to running the script **sample/english/sample.tcl**.

INSTALLATION

Edit **src/GNUMakefile** and update the locations for your Tcl installation and the location (if possible) to the latest **winuser.h** header file.

Run `make install` to create the directory **../CWind1.5.1**

Copy that directory to your TCL library.

LICENSE

CWind is free software copyrighted by Satoshi Imai. You can redistribute it and/or modify it under the terms of the [GNU Lesser General Public License](#) as published by the Free Software Foundation.

Use this package at your own risk and expense.

s-imai@japan.interq.or.jp

[Author's Project home page](#)

Tcl [wiki](#) page which discusses the cwind extension