Subject: EMACS wave-mode now available
Posted by lubosp on Thu, 28 Jan 1993 16:02:21 GMT
View Forum Message <> Reply to Message

Beta version of wave.el, EMACS wave mode for editing of applications
written in WAVE Command Language is now available on Visual Numerics, Inc.
anonymous ftp site.

This version was tested in-house with emacs versions 18.57, and 19.1
(lemacs), and WAVE CL. It is used extensively for internal development.
Because there have been numerous questions on this newsgroup in the past,
we thought it might be of public interest so we are making it available.

In accordance with Free Software Foundation policy, wave-mode is
distributed free of charge; standard FSF copyright policy applies.

Mail bugs, suggestions, improvements to lubosp@pvi.com.

Please read following release notes about wave-mode functionality, how to
get it, how to install it, and where to mail bug reports:

 Emacs WAVE Mode (wave-mode) for WAVE Command Language.

Features:

 o Case Conversion

 o reserved words are converted to upper case
 o system variables are capitalized
 o COMMON name is capitalized
 o PRO or FUNCTION name is capitalized

 o Spacing

o space inserted around:
 & - multiple statement
 ,
 = - including keywords
 <
 >

 o Defined Keys

 C-c ; - comment region

M-; - insert and indent parallel comment
M-C-d - mark library documentation region

C-M-a - beginning of subprogram
C-M-e - end of subprogram
M-C-h - mark subprogram
M-C-q - indent subprogram

M-RET - start continuation line
M-TAB - insert TAB
C-c C-p - previous statement
C-c C-n - next statement
TAB - indent line

 o User Option Variables

 Variables controlling indentation style and extra features:

  wave-block-indent
   Extra indentation within blocks. (default 3)

  wave-continuation-indent
   Extra indentation within continuation lines. (default 2)

 wave-doclib-start
   Start of Wave CL document library header. (default ^[;*]\+)

 wave-doclib-end
   End of Wave CL document library header.   (default ^[;*]-)

 wave-minimum-statement-indent
   Minimum indentation for wave statements. (default 0)

 wave-newline-and-indent
   Automatically indents current line, inserts newline and
   indents it.   (default is t)

 wave-comment-region
   String inserted by \\[wave-comment-region] at start of each line in
   region. (default \";; \")

 wave-surround-by-blank
   Automatically surrounds '=','<','>','&' with blanks,
   appends blank to comma. (default is t)

 wave-startup-message
   Set to nil to inhibit message first time wave-mode is used.

wave-block-match-back-max
  Maximum number of lines to search for matching block delimiter
  to blocks lines. (default is 40)
  Set to 0 to suppress block matching.

 o Indentation

By default current and next comment line is indented, when RET is
entered. This can be suppressed by setting `wave-newline-and-indent'
to nil. TAB is used for explicit indentation.

After the line is indented, all abbreviations on that line are
expanded, spacing ant case conversion is performed.

 Code Indentation:
   Variable `wave-block-indent' specifies relative indent for
   block statements(begin...end),
   variable `wave-continuation-indent' specifies relative indent for
   continuation lines.
   Continuation lines inside [], (), {} (structures) are indented by
   `wave-continuation-indent' chars after opening parenthesis.
   Continuation lines in PRO, FUNCTION declarations are indented
   just after the procedure/function name followed by comma.
Labels not followed by code are left justified to the beginning
of the line.
Include files introduced by '@' are left justified to the beginning
of the line.

 Comment Indentation:
   Full line comments starting at the beginning of the line are
are not indented.
Full lin comments starting with white space are indented as
a code.
   Code line comment is indented to the value of `comment-column'.

 o Block Statement, Quote Matching

When end of the block statement is entered (ENDCASE, ENDELSE,
ENDIF, ENDREP, ENDWHILE, or END), cursor flashes shortly to the
beginning of the block.

When closing double or single quote is entered, cursor flashes
shortly to the beginning of the block.

o Abbreviations

Abbreviations are available for all reserved words, often used functions, and WAVE Toolbox and Widget routines.

'.' character is used as a prefix for abbreviations.

To get entire list of abbreviations type .?, while in wave-mode.

List of abbreviations:

```
.b BEGIN    .c CASE OF
    .cb BYTE( )    .cc COMPLEX( )
    .cd DOUBLE( )  .cf FLOAT( )
    .cl LONG( )    .co COMMON
    .cs STRING( )  .cx FIX( )
    .d DO    .e ELSE
    .ec ENDCASE    .ee ENDELSE
    .ef ENDFOR    .ei ENDIF ELSE IF
    .el ENDIF ELSE   .en ENDIF
    .er ENDREP    .ew ENDWHILE
    .f FOR DO    .fu FUNCTION
    .g GOTO,    .h HELP,
    .i IF    .k KEYWORD_SET( )
    .n N_ELEMENTS( )   .oi ON_IOERROR,
    .on ON_ERROR,   .or OPENR,
    .ou OPENU,    .ow OPENW,
    .p PRO    .pr PRINT,
    .pt PLOT,    .r REPEAT UNTIL
    .re READ,    .rf READF,
.rt RETURN    .ru READU,
    .sc STRCOMPRESS( )  .sl STRLOWCASE( )
    .sm STRMID( )   .sn STRLEN( )
    .sp STRPOS( )   .sr STRTRIM( )
    .st STRPUT( )   .su STRUPCASE( )
    .t THEN    .tac WtAddCallback( )
    .tah WtAddHandler( )   .tc WtCreate( )
    .tca WtCallback( )   .tcl WtClose( )
    .tg WtGet( )   .ti WtInit( )
    .tl WtList( )   .tlo WtLoop( )
    .tml WtMainLoop( )   .tp WtPointer( )
    .ts WtSet( )   .tt WtTimer( )
    .u UNTIL    .w WHILE DO
    .wb WwButtonBox( )   .wcm WwCommand( )
    .wco WwControlBox( )   .wdi WwDialog( )
    .wdr WwDrawing( )   .wf WwFileSelection( )
    .wg WwGetValue( )   .wi WwInit( )
    .wla WwLayout( )   .wli WwList( )
```

.wlo WwLoop   .wm WwMessage( )
.wmb WwMenuBar( )   .wmw WwMainWindow( )
.wo WwOptionMenu( )   .wp WwPopupMenu( )
.wr WwRadioBox( )   .ws WwSetValue( )
.wt WwText( )   .wto WwToolBox( )
.wu WRITEU,

 o Modifying in Batch Mode

You can modify your WAVE CL *.pro file in batch mode using
following function and command:

Example:

1. Create following Lisp function in wave-indent.el file:

(wave-mode)
(let ((start 1) (end (save-excursion (end-of-buffer) (point))))
  (indent-region start end nil))
(save-buffer)

2. Issue following command:

emacs -batch <filename> -l wave-indent.el

 o Installation

 To install:

 1. Download wave.el file (and other files, if you wish), and place
   them to appropriate 'load' directory.

 2. Put in your ~/.emacs file:
 (you may want to specify a file path for wave.el file)

 (setq auto-mode-alist
    (append
     '(("\\.pro$" . wave-mode))
     auto-mode-alist))
 (autoload 'wave-mode "wave"
  "Major mode for edition WAVE CL .pro files" t)

 o Customization

   Turning on Wave mode calls the value of the variable wave-mode-hook

with no args, if that value is non-nil.

    Example:
;;
;; To customize wave-mode put following code to your .emacs file
;;
(setq wave-mode-hook
      (function (lambda ()
  (setq wave-block-indent 6
        wave-continuation-indent 3
        wave-minimum-statement-indent 2
            wave-newline-and-indent nil
            wave-surround-by-blank nil))))

 o Distribution

WAVE CL wave-mode Emacs mode is available on Visual Numerics, Inc.
anonymous ftp:

 address: ftp.pvi.com (128.138.213.22)

 files: PVI/emacs/README
   PVI/emacs/wave.el
   PVI/emacs/wave.elc
   PVI/emacs/wave-indent.el
   PVI/emacs/wave-mode-hook.el
   PVI/emacs/wave-mode.txt

For better performance byte compile wave.el file after download.

In accordance with Free Software Foundation policy, wave-mode is
distributed free of charge; standard FSF copyright policy applies.

Mail bugs, suggestions, improvements to lubosp@pvi.com.
--
 ++
+++++++++++++++++
 Lubos Pochman, Visual Numerics, Inc. phone: (303)530-9000,ext.355
     email: lubosp@pvi.com