

Applixware Words ELF Reference

COPYRIGHT NOTICE ON THE VERSION 6.0 SOFTWARE
©1990 - 2010 Vistasource, Inc. All Rights Reserved.

Vistasource, Inc. prepared the information contained in this document for use by Vistasource personnel, customers, and prospects. Vistasource reserves the right to change the information in this document without prior notice. The contents herein should not be construed as a representation or warranty by Vistasource. Vistasource assumes no responsibility for any errors that may appear in this document.

The Proximity Thesauri ®
©1985 Merriam-Webster Inc.
©1988 Williams Collins Sons & Co. Ltd.
©1989 Van Dale Lexicografie bv. ©1989 Nathan. ©1989 Kruger.
©1989 Zanichelli. ©1989 International Data Education a s.
©1989 C.A. Stromber A B. ©1989 Espasa-Calpe.
©1983-1996. Proximity Technology, Inc.
All Rights Reserved.

The Proximity Linguibase And Hyphenation Systems®
©1983 Merriam-Webster Inc.
©1984, 1985, 1986, 1988, 1990 Williams Collins Sons & Co. Ltd.
©1987, 1989 Van Dale Lexicografie bv. ©1988 Munksgaard International Publishers Ltd.
©1988, 1989 International Data Education a s.
©1983-1996 Proximity Technology, Inc.
All Rights Reserved

The Applixware Graphics Filter Pack contains elements of the Generator Metafile Development Libraries (MDL/G)
©1988-1996 Henderson Software, Inc.
All Rights Reserved

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraphs (c) (1) (ii) of SFARS 252.277-7013, or in FAR 52.227-19, as applicable.

Hardware and software products mentioned herein are used for identification purposes only and may be trademarks of their respective companies.

Applixware is a registered trademark of Vistasource, Inc. Applixware, Applixware Real Time, Applixware Data, and Applixware Builder are trademarks of Vistasource, Inc.

This manual was produced using Applixware.

Printed: June 2010

Applixware Words ELF Reference

WP_ACCESS_SPECIAL_DICT@

Returns the dictionary file name for the current Skip or Document dictionary

Format filename = WP_ACCESS_SPECIAL_DICT@(type)

Method filename = this.access_special_dict@(type)

Arguments type One of the following values:

- 0 Dictionary of skipped words. There is either 0 or 1 of these dictionaries.
- 1 Dictionary for the current document. There is either 0 or 1 of these dictionaries. (Not yet implemented.)

Description Returns the dictionary file name for the current Skip or Document dictionary. If the dictionary does not exist, it is created and used in subsequent spelling operations.

See also [WP_ACCESS_USER_DICT@](#)
[WP_ADD_DICT@](#)

WP_ACCESS_USER_DICT@

Returns the file name for the "default" user dictionary for the language

Format filename = WP_ACCESS_USER_DICT@(language[, addToWpFlag[, createDictFlag]])

Method filename = this.access_user_dict@(language[, addToWpFlag[, createDictFlag]])

Arguments language The passed language, one of the sixteen supported languages, as follows:

- 1 English

- 2 German
- 3 French
- 4 Spanish
- 5 Italian
- 6 British English
- 7 Swedish
- 8 Danish
- 9 Norwegian
- 10Dutch
- 11Portuguese
- 12Brazilian-Portuguese
- 13Canadian-French
- 14Swiss-German
- 15Nynorsk (Norwegian dialect)
- 16Finnish

addToWpFlag

Boolean value where TRUE means that the dictionary should be used when spell checking is performed.

createDictFlag

Boolean value where TRUE means that a dictionary will be created even if necessary.

Description Returns the file name for the "default" user dictionary for a language. If the dictionary does not exist, it is created and used in subsequent spelling operations.

See also [WP_ACCESS_SPECIAL_DICT@](#)
[WP_ADD_DICT@](#)

WP_ADD_COLOR@

Adds a color (by name)

Format WP_ADD_COLOR@(name, cyan, magenta, yellow, black)

Method [this.add_color@](#)(name, cyan, magenta, yellow, black)

Arguments

name	The name of the color.
cyan	The color's cyan component.
magenta	The color's magenta component.
yellow	The color's yellow component.

WP_SAVE_OBJECT@

WP_ADD_TAG@

Adds tag to the tag list

Format WP_ADD_TAG@(name, bead, offset)

Method [this.add_tag@](#)(name, bead, offset)

Arguments

name	The user-supplied name of the tag.
bead	The bead component of a tag's location.
offset	The offset component of a tag's location

Description Adds a tag to Applixware Word's tag list. If a tag with this name already exists, the existing tag is repositioned.

Use this function to save a location in the document for future use. For example, you could add a tag at the current cursor location, perform a series of operations that adjusts the cursor position, then move back to the place where you started.

You can obtain the bead and offset position of the cursor using the [WP_LOCATION_OF_CURSOR@](#) macro.

See also [WP_GET_ALL_TAGS@](#)
[WP_GET_TAG@](#)

WP_ADJUST_CURSOR_FOR_INSET@

Places the cursor in an accessible location

Format WP_ADJUST_CURSOR_FOR_INSET@(appType, inFrameFlag)

Method [this.adjust_cursor_for_inset@](#)(appType, inFrameFlag)

Arguments

appType	An application's type, as follows: 0 - AX_APP_TYPE_NONE 1 - AX_APP_TYPE_WORDS 2 - AX_APP_TYPE_GRAPHICS 3 - AX_APP_TYPE_SPREADSHEET 4 - AX_APP_TYPE_MACRO 5 - AX_APP_TYPE_AUDIO 6 - AX_APP_TYPE_EQUATION 7 - AX_APP_TYPE_QUERY 8 - AX_APP_TYPE_BITMAP
---------	---

inFrameFlag A Boolean value where TRUE means the object has a frame. (Objects from Graphics are assumed to have frames.) No adjustment will occur if this value is FALSE.

Description Places the cursor in an accessible location. In most cases, this means inserting a paragraph break before and after the current cursor position. If the location is in a table, a new paragraph is inserted after the inset and the cursor placed before it.

WP_ALIGN_CENTER@

Centers the current paragraphs

Format WP_ALIGN_CENTER@()

Method [this.align_center@](#)

Description Justifies the paragraph (or paragraphs) of text in which the cursor resides so that the text is centered with both the left and right column borders.

See also [WP_ALIGN_FULL@](#)
[WP_ALIGN_LEFT@](#)
[WP_ALIGN_RIGHT@](#)

WP_ALIGN_FULL@

Applies full justification to the current paragraphs

Format WP_ALIGN_FULL@()

Method [this.align_full@](#)

Description Justifies the paragraph (or paragraphs) of text in which the cursor resides so that the text is aligned with both the left and right column borders.

See also [WP_ALIGN_CENTER@](#)
[WP_ALIGN_LEFT@](#)
[WP_ALIGN_RIGHT@](#)

WP_ALIGN_LEFT@

Aligns the current paragraphs to the left margin

Format WP_ALIGN_LEFT@()

Method [this.align_left@](#)

Description Aligns text in the current paragraph (or paragraphs) with the left margin.

See also [WP_ALIGN_CENTER@](#)
[WP_ALIGN_FULL@](#)
[WP_ALIGN_RIGHT@](#)

WP_ALIGN_RIGHT@

Aligns the current paragraphs with the right margin

Format WP_ALIGN_RIGHT@()

Method [this.align_right@](#)

Description Aligns the current paragraph (or paragraphs) with the right column border.

See also [WP_ALIGN_CENTER@](#)
[WP_ALIGN_FULL@](#)
[WP_ALIGN_LEFT@](#)

WP_APPLICATION_DLG@

Creates a new Words window

Format taskID = WP_APPLICATION_DLG@
([menubarID][,windowlessFlag][,hooklessFlag])

Arguments menubar_id The number of a menu bar to be associated with this window. (This argument is optional.) This number should be a number between 200 and 299. If the value is set to NULL, you will receive the default menu bar.

windowlessFlag

Boolean value where TRUE indicates that no window will be displayed. FALSE is the default.

hooklessFlag

A Boolean value where TRUE indicates that no hook macro is run at startup time. If the value is FALSE, Words runs the hook macro configured in the Words preferences.

Description Creates a new Words window. This window can use the default menu bar or it can use the menu bar associated with menubar_id. The task id for the newly created task is returned.

The optional `menubar_id` parameters lets you load a menu bar according to the task that will be performed in the Words window. You could even display several versions at the same time by opening several windows, each with separate numbers. To display a custom menu bar, you have to first load it into memory using **SET SELECTIONS@**.

The `windowlessFlag` parameter lets you perform automated tasks "in the background," without displaying the Words application window. Using this option, two applications can work simultaneously without interrupting each other and without invoking two separate `axmain` processes. This windowless Words application becomes a "child" of the window from which it was invoked.

Performing non-interactive tasks without a window conserves computing resources as the window will never have to be displayed.

To perform a windowless task:

- You must suppress all messages that are displayed. Otherwise, the task will hang when it tries to display them. To suppress information displayed by `INFO_MESSAGE@`, use **SUPPRESS INFO MESSAGES@**. Suppress error messages with the **SUPPRESS ERROR MESSAGES@** macro.
- You cannot include any prompts using `PROMPT@`.
- Be sure to explicitly exit any windowless Words applications after they are complete. Otherwise, the tasks remain in memory until you log out.
- It is best to make a call to **SELECT WINDOW@** when exiting the windowless application. In this way, you guarantee that the exit command is invoked against the right window.
- Before testing a newly written macro that invokes a windowless application, it is best to first test the macro with all windows displaying in the foreground.

The `hooklessFlag` parameter determines whether a hook macro is run when the Words application starts. A hook macro is configured through the Words preferences dialog, and runs whenever you start Words or open a Words file.

If you configure a hook macro to run at startup time, and you set `hooklessFlag` to `NULL`, the hook macro runs. If you configure a hook macro to run at startup time, and you set `hooklessFlag` to `TRUE`, the hook macro does not run. This is an optional parameter.

`WP_APPLICATION_DLG@` is called by the *® Words menu option.

Example

WP_APPLY_FRAME@

Inserts a frame

Format `WP_APPLY_FRAME@(format wp_frame@ frame)`

Method `this.apply_frame@(format wp_frame@ frame)`

Arguments `frame` The attributes of the frame being inserted; this variable is of type `wp_frame@`.

Description Creates a new frame at the current cursor. The attributes of the frame are those specified in the passed `wp_frame@`, whose definition is as follows:

```
format wp_frame@
    type,
    xposType,
    xposRelative,
    xpos,
    yposType,
    yposRelative,
    ypos,
    widthType,
    width,
    heightType,
    height,
    leftMargin,
    rightMargin,
    topMargin,
    bottomMargin
```

WP_APPLY_SAVED_ATT@

Applies saved paragraph attributes to the current selection

Format `WP_APPLY_SAVED_ATT@()`

Method `this.apply_saved_atts@`

Description Applies the paragraph attributes that were saved using [WP_SAVE_ATT@](#). (`WP_SAVE_ATT@` saves all current paragraph attributes. These attributes are the ones that were defined as part of the paragraph's style and those which the user has changed.)

WP_APPLY_STYLE@

Changes style of current paragraph

Format `WP_APPLY_STYLE@(styleName, forceFlag)`

Method `this.apply_style@(styleName, forceFlag)`

Arguments styleName The style whose attributes will be applied to the current selection.
forceFlag A Boolean value, where FALSE means that local settings are not altered when the new style is applied.

Description Changes the style of the current paragraph or selection to styleName. The force-Flag argument, if TRUE, specifies that all the new style's attributes will be applied. That is, you will remove all local text and paragraph attributes. If it is FALSE, the style's attributes are only applied to attributes that the user has not changed.

For example, suppose the user changes a paragraph left margin. Setting force-Flag to TRUE will change the left margin to that specified for styleName. Setting the flag to FALSE says that this local margin will not be changed even though the new style has (or may have) a different margin.

WP_ASCII_TEMPLATE@

Imposes an ASCII template style (ascii.aw or ascii.am)

Format flag = WP_ASCII_TEMPLATE@([errMsgFlag])

Method flag = this.ascii_template@([errMsgFlag])

Arguments errMsgFlag A Boolean value where TRUE means that an error message is posted if the template file is not found.

Description Imposes the ASCII template style upon the current document. Only use this macro for an ASCII Macro Editor file or an ASCII Applixware Words import. This macro returns a Boolean value where TRUE means that the ASCII template file was available.

WP_BACKSPACE_KEY@

Deletes the character preceding the cursor

Format WP_BACKSPACE_KEY@()

Method this.backspace_key@

See also [WP_DELETE_CHAR_KEY@](#)

WP_BACKWARD_SEARCH@

Searches backward for a string

Format flag = WP_BACKWARD_SEARCH@(string, wholeWordsflag, matchCaseflag, format wp_para_attrs@ paraAtts, format wp_text_attr@ textAtts)

Method this.backward_search@(string, wholeWordsflag, matchCaseflag, format wp_para_attrs@ paraAtts, format wp_text_attr@ textAtts)

Arguments

string	The search string.
wholeWordsflag	A Boolean, where TRUE means that the search only succeeds if the letters of the word are not contained within another word.
matchCaseflag	A Boolean, where TRUE means that the search only succeeds if the case of the letters is the same as the case of the letters in string.
paraAtts	A variable of type wp_para_attrs@ that indicates paragraph attributes of the search string.
textAtts	A variable of type wp_text_attr@ that indicates character attributes of the search string. For example, you could request that the search string be in bold.

Description Searches from the current cursor position back to the beginning of the file looking for a string. A Boolean value is returned indicating if the search succeeded. You can modify this search as follows:

- You could allow the string to appear anywhere within text or you could require that your search succeed only if your string is not part of another string.
- You could allow the search to succeed if the case of the found string differs from the case of the search string.
- You could require that the search succeed only if the string were found in a paragraph matching the attributes you specify.
- You could require that the search succeed only if the found string has the specified character attributes.
- The search string can be a pattern. (Patterns must be preceded by a back slash ("\") escape character. To type a back slash character, you must type a double back slash ("\\").

The search string patterns are as follows:

- * any group of characters
- ? any one character
- soft hyphen

< word start
> word end
[paragraph start
] paragraph end
{ field start
} field end
|...| multi-byte pattern code delimiter
c clipboard
C column break
I inset
n newline
p paragraph marker
P pagebreak
S section break
t tab

Patterns within multi-byte sequences include:

F field method
G Glossary name
L Latin
N Style name
S Symbol font
Z Dingbat font

For example, the following pattern looks for a style named Head1Para:

`\|NHead1Para|`

See also [WP_FORWARD_SEARCH@](#)

WP_BOLD@

Makes selected text bold

Format WP_BOLD@()

[Method](#) this.bold@

Description Makes selected text bold in a Words document. If no text is selected, the bold attribute is applied to whatever you type after invoking the macro.

See also [WP_ITALICS@](#)
[WP_UNDERLINE@](#)

WP_BORDER@

Sets the selection's borders

Format WP_BORDER@(type, thickness, color, doubleLine)

Method [this.border@](#)(type, thickness, color, doubleLine)

Arguments type The border being set, which is as follows:

WP#BORDER#OUTLINE	0
(all exterior sides)	
WP#BORDER#LEFT	1
WP#BORDER#RIGHT	2
WP#BORDER#TOP	3
WP#BORDER#BOTTOM	4
WP#BORDER#HORIZONTAL	5
WP#BORDER#VERTICAL	6

thickness The width of the line in mils. (1000 mils = 1 inch)

color The string name of a color.

doubleLine Any value except NULL indicates that a double line will be drawn.

Description Adds or changes a paragraph's or cell's borders. The type argument lets you specify which border will be changed. The remaining three arguments specify the border's thickness, its color, and whether it is single or double line.

See also [WP_BORDER_BOTTOM_STYLE@](#)

[WP_BORDER_HORIZONTAL_STYLE@](#)

[WP_BORDER_LEFT_STYLE@](#)

[WP_BORDER_OUTLINE_STYLE@](#)

[WP_BORDER_RIGHT_STYLE@](#)

[WP_BORDER_TOP_STYLE@](#)

[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_BOTTOM_STYLE@

Sets the bottom border's style

Format WP_BORDER_BOTTOM_STYLE@(style, color)

Method [this.border_bottom_style@](#)(style, color)

Arguments style A line's visual appearance, which is as follows:

0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a paragraph's or cell's bottom border. The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn.

See also [WP_BORDER@](#)
[WP_BORDER_HORIZONTAL_STYLE@](#)
[WP_BORDER_LEFT_STYLE@](#)
[WP_BORDER_OUTLINE_STYLE@](#)
[WP_BORDER_RIGHT_STYLE@](#)
[WP_BORDER_TOP_STYLE@](#)
[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_HORIZONTAL_MARGIN@

Specifies a border's distance from text

Format WP_BORDER_HORIZONTAL_MARGIN@ (horizontalMargin)

Method [this.border_horizontal_margin@](#) (horizontalMargin)

Arguments horizontalMargin The distance in mils (1000 mils = 1 inch) that a border is separated from a paragraph or cell's text.

Description Specifies the distance above and below a paragraph's or cell's text at which a border is drawn.

See also [WP_BORDER@](#)
[WP_BORDER_MARGINS@](#)
[WP_BORDER_VERTICAL_MARGIN@](#)

WP_BORDER_HORIZONTAL_STYLE@

Sets the horizontal border's style

Format WP_BORDER_HORIZONTAL_STYLE@(style, color)

Method [this.border_horizontal_style@\(style, color\)](#)

Arguments

style	A line's visual appearance, which is as follows:	
0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a paragraph's or a cell's horizontal border. (These are the border's drawn within a table or between paragraph's.) The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn.

See also [WP_BORDER@](#)
[WP_BORDER_BOTTOM_STYLE@](#)
[WP_BORDER_LEFT_STYLE@](#)
[WP_BORDER_OUTLINE_STYLE@](#)
[WP_BORDER_RIGHT_STYLE@](#)
[WP_BORDER_TOP_STYLE@](#)
[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_LEFT_STYLE@

Sets the left border's style

Format WP_BORDER_LEFT_STYLE@(style, color)

Method [this.border_left_style@\(style, color\)](#)

Arguments style A line's visual appearance, which is as follows:

0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a paragraph's or cell's left border. The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn.

See also [WP_BORDER@](#)
[WP_BORDER_BOTTOM_STYLE@](#)
[WP_BORDER_HORIZONTAL_STYLE@](#)
[WP_BORDER_OUTLINE_STYLE@](#)
[WP_BORDER_RIGHT_STYLE@](#)
[WP_BORDER_TOP_STYLE@](#)
[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_MARGINS@

Sets the distance from text to a border

Format WP_BORDER_MARGINS@(horizontalMargin, verticalMargin)

Method [this.border_margins@](#)(horizontalMargin, verticalMargin)

Arguments horizontalMargin
The distance in mils (1000 mils = 1 inch) that a border is separated from a paragraph or cells text.

verticalMargin
The distance in mils that a border is separated from a cell's text. (Vertical margins do not occur within a paragraph selection.)

Description Specifies the horizontal and vertical distance separating a paragraph's or cell's text from a border.

See also [WP_BORDER@](#)
[WP_BORDER_HORIZONTAL_MARGIN@](#)
[WP_BORDER_VERTICAL_MARGIN@](#)

WP_BORDER_OUTLINE_STYLE@

Sets the style of the border surrounding a selection

Format WP_BORDER_OUTLINE_STYLE@(style, color)

Method [this.border_outline_style@](#)(style, color)

Arguments

style	A line's visual appearance, which is as follows:	
0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a paragraph's or cell's top, bottom, left and right borders. The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn. All of the borders are drawn using the same style and color.

See also [WP_BORDER@](#)
[WP_BORDER_BOTTOM_STYLE@](#)
[WP_BORDER_HORIZONTAL_STYLE@](#)
[WP_BORDER_LEFT_STYLE@](#)
[WP_BORDER_RIGHT_STYLE@](#)
[WP_BORDER_TOP_STYLE@](#)
[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_RIGHT_STYLE@

Sets the right border's style

Format WP_BORDER_RIGHT_STYLE@(style, color)

Method [this.border_right_style@](#)(style, color)

Arguments

style	A line's visual appearance, which is as follows:	
0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a paragraph's or cell's right border. The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn.

See also [WP_BORDER@](#)
[WP_BORDER_BOTTOM_STYLE@](#)
[WP_BORDER_HORIZONTAL_STYLE@](#)
[WP_BORDER_LEFT_STYLE@](#)
[WP_BORDER_OUTLINE_STYLE@](#)
[WP_BORDER_TOP_STYLE@](#)
[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_SHADOW@

Sets a selection's drop shadow attribute

Format WP_BORDER_SHADOW@(shadow)

Method [this.border_shadow@](#)(shadow)

Arguments shadow An integer that indicates the shadow's placement. The values you can specify are:

WP#SHADOW#NONE	0
WP#SHADOW#BOTTOM#RIGHT	1

Description Specifies if a selected paragraph (or paragraphs) is displayed with a drop shadow border.

See also [WP_BORDER@](#)

WP_BORDER_TOP_STYLE@

Sets the border's top style

Format WP_BORDER_TOP_STYLE@(style, color)

Method [this.border_top_style@\(style, color\)](#)

Arguments style A line's visual appearance, which is as follows:

0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a paragraph's or cell's top border. The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn.

See also [WP_BORDER@](#)

[WP_BORDER_BOTTOM_STYLE@](#)

[WP_BORDER_HORIZONTAL_STYLE@](#)

[WP_BORDER_LEFT_STYLE@](#)

[WP_BORDER_OUTLINE_STYLE@](#)

[WP_BORDER_RIGHT_STYLE@](#)

[WP_BORDER_VERTICAL_STYLE@](#)

WP_BORDER_VERTICAL_MARGIN@

Specifies a border's distance from text

Format WP_BORDER_VERTICAL_MARGIN@(verticalMargin)

Method [this.border_vertical_margin@](#)(verticalMargin)

Arguments verticalMargin

The distance in mils (1000 mils = 1 inch) that a border is separated from a paragraph or cell's text.

Description Specifies the distance between a cell's text and the vertical border lines drawn between cells. (Vertical borders cannot be drawn between paragraphs.)

See also [WP_BORDER@](#)

[WP_BORDER_HORIZONTAL_MARGIN@](#)

[WP_BORDER_MARGINS@](#)

WP_BORDER_VERTICAL_STYLE@

Sets the style of vertical borders

Format WP_BORDER_VERTICAL_STYLE@(style, color)

Method [this.border_vertical_style@](#)(style, color)

Arguments style A line's visual appearance, which is as follows:

0	no border	
1	hairline	4 mils (1000 mils = 1 inch)
2	single line	1 pixel
3	double line	1 pixel each
4	single line	2 pixels
5	double line	2 pixels each
6	single line	4 pixels
7	double line	4 pixels each
8	single line	6 pixels
9	single line	7 pixels
10	single line	9 pixels

color The name of color registered with Words.

Description Adds or changes a cell's vertical border. The style parameter specifies the kind of line to be drawn while the color parameter specifies the color in which the border will be drawn.

See also [WP_BORDER@](#)
[WP_BORDER_BOTTOM_STYLE@](#)
[WP_BORDER_HORIZONTAL_STYLE@](#)
[WP_BORDER_LEFT_STYLE@](#)
[WP_BORDER_OUTLINE_STYLE@](#)
[WP_BORDER_RIGHT_STYLE@](#)
[WP_BORDER_TOP_STYLE@](#)

WP_BULLETS@

Adds or replaces a bullet

Format WP_BULLETS@(format array of wp_bead@ newBullet, bulletPolicy, noBulletPolicy, latinBullets, spacePolicy, doAdjust, adjustment)

Method [this.bullets@](#)(format array of wp_bead@ newBullet, bulletPolicy, noBulletPolicy, latinBullets, spacePolicy, doAdjust, adjustment)

Arguments

newBullet	Beads containing the bullet character and whitespace or tab.						
bulletPolicy	For paragraphs that already have bullets, one of the following: <table border="0" style="margin-left: 20px;"> <tr> <td>WP_BULLET_NO_CHANGE</td> <td style="text-align: right;">0</td> </tr> <tr> <td>WP_BULLET_REPLACE</td> <td style="text-align: right;">2</td> </tr> <tr> <td>WP_BULLET_DELETE</td> <td style="text-align: right;">3</td> </tr> </table>	WP_BULLET_NO_CHANGE	0	WP_BULLET_REPLACE	2	WP_BULLET_DELETE	3
WP_BULLET_NO_CHANGE	0						
WP_BULLET_REPLACE	2						
WP_BULLET_DELETE	3						
noBulletPolicy	For paragraphs that do not have bullets, one of the following: <table border="0" style="margin-left: 20px;"> <tr> <td>WP_BULLET_ADD</td> <td style="text-align: right;">1</td> </tr> <tr> <td>WP_BULLET_NO_CHANGE</td> <td style="text-align: right;">0</td> </tr> </table>	WP_BULLET_ADD	1	WP_BULLET_NO_CHANGE	0		
WP_BULLET_ADD	1						
WP_BULLET_NO_CHANGE	0						
latinBullets	The ASCII character you are using as a bullet character.						
spacePolicy	For existing bullets, should trailing whitespace be added. The possible values are: <table border="0" style="margin-left: 20px;"> <tr> <td>WP_BULLET_NO_TRAILING</td> <td style="text-align: right;">0</td> </tr> <tr> <td>WP_BULLET_SPACES_OR_TAB</td> <td style="text-align: right;">1</td> </tr> <tr> <td>WP_BULLET_SPACES</td> <td style="text-align: right;">2</td> </tr> </table>	WP_BULLET_NO_TRAILING	0	WP_BULLET_SPACES_OR_TAB	1	WP_BULLET_SPACES	2
WP_BULLET_NO_TRAILING	0						
WP_BULLET_SPACES_OR_TAB	1						
WP_BULLET_SPACES	2						
doAdjust	A Boolean value where TRUE means that Words should adjust the paragraph's left margin so that the second line is indented more than the first margin.						
adjustment	Offset of the left margin from first margin. This value is only applied if doAdjust is TRUE.						

Description Adds or removes bullets that start paragraphs within the selected range. Empty paragraphs do not receive bullets.

For example, the bullet character and the tab character following it can be constructed in the following way:

```
newBullets[0] = WP_INIT_TEXT_BEAD@(FALSE, bullet)
newBullets[0].text_attrs.face = "Symbol"
newBullets[0].text_attrs_flags.face = TRUE
newBullets[1] = WP_INIT_TEXT_BEAD@(FALSE,
    NUM_TO_STRING@(9))
```

WP_CALCULATE@

Calculates numeric data in a Words document

Format WP_CALCULATE@(func, find, styles, noNum, parensFlag, goodCalcFlag, nParas, nNumbers)

Method [this.calculate@](#)(func, find, styles, noNum, parensFlag, goodCalcFlag, nParas, nNumbers)

Arguments

func	Indicates the operation performed on the selected paragraphs, as follows:
	WP_CALC_SUM 0
	WP_CALC_PRODUCT 1
	WP_CALC_AVERAGE 2
	WP_CALC_MEDIAN 3
	WP_CALC_MIN 4
	WP_CALC_MAX 5
find	Indicates which of the selected paragraphs are analyzed, as follows:
	WP_CALC_ALL 0
	WP_CALC_STYLES 1
styles	Indicates which styles contain information. This string is only used if find is set to 1.
noNum	Determines how to treat paragraphs without numbers, as follows:
	WP_CALC_NONUM_IGNORE 0
	WP_CALC_NONUM_ZERO 1
	WP_CALC_NONUM_WARN 2
parensFlag	A Boolean value where TRUE means numbers in parenthesis are treated as negative.
goodCalcFlag	A Boolean value, which when set by ELF to TRUE, indicates that the calculation succeeded. (This is an output argument.)

nParas The number of paragraphs that this macro looked at. (This is an output argument.)

nNumbers The number of values found in the evaluated paragraphs. This will be less than or equal to nParas. (This is an output argument.)

Description Performs simple numeric operations on data. If a paragraph does not contain numeric data and you specify WP_CALC_NONUM_WARN, goodCalcFlag is set to FALSE. If the caller does not specify this parameter, goodCalcFlag is set to TRUE (even if no good values were seen, and 0 is returned).

WP_CELL_IS_SELECTED@

Determines if the passed cell is selected

Format flag = WP_CELL_IS_SELECTED@(beadNum)

Method flag = this.cell_is_selected@(beadNum)

Arguments beadNum Any bead number.

Description Returns TRUE if the selection contains a fully selected cell that contains beadNum. Note that a row start bead is not in a cell.

See also [WP_SELECT_CELL@](#)
[WP_SELECT_NEXT_CELL@](#)
[WP_SELECT_PREV_CELL@](#)
[WP_SELECT_RANGE_CELLS@](#)

WP_CELL_LIST_BOX@

Manages a list of choices for a cell

Format WP_CELL_LIST_BOX@(beadNum, Cell)

Method this.cell_list_box@(beadNum, Cell)

Arguments beadNum The bead number of the cell.
ID The cell ID.

Description In a form, the macro searches for a document variable with the same name as ID. If a document variable with a one-dimensional array matches ID, a list box appears. A selected name from the list box is placed as a text string within the cell.

This macro is usually called by a double-click macro in a form cell.

WP_CELL_MANAGE_WIDGET@

Sets or alters the state of a radio or toggle widget in a cell

Format WP_CELL_MANAGE_WIDGET@(beadNum, operator, onGloss, offGloss)

Method [this.cell_manage_widget@](#)(beadNum, operator, onGloss, offGloss)

Arguments

beadNum	The bead number of the cell.
operator	One of the following operations: WP#CELL#WIDGET#TOGGLE Change the toggle widget WP#CELL#WIDGET#ON Set the radio widget to ON WP#CELL#WIDGET#OFF Set the radio widget to OFF
onGloss	The <i>on</i> glossary.
offGloss	The <i>off</i> glossary.

Description In a form, this macro sets or alters the state of a radio or toggle widget in a cell. This macro is usually called by a double-click macro in a form cell.

See also [WP_CELL_RADIO_VALUE@](#)
[WP_CELL_TOGGLE@](#)

WP_CELL_RADIO_VALUE@

Returns the toggled option in a radio button

Format WP_CELL_RADIO_VALUE@(ID, beadnum)

Method [this.cell_radio_value@](#)(ID, beadNum)

Arguments

ID	A string containing the cell ID.
beadNum	The bead number of a cell in the form. You can get a legal bead number by placing the cursor in the form, and call WP_GET_CURRENT_PLACE@.

Description Given a cell ID and a bead number, this function finds all cells in the table with this ID. If the widget in the first cell (with the proper ID) is ON, zero is returned. If the second cell is ON, 1 is returned, and so on. If none of the glossaries is ON (shouldn't happen), -1 is returned.

The table is searched in bead-number-order. Rows are searched top to bottom. Within each row, cells are examined left to right.

See also [WP_CELL_LIST_BOX@](#)
[WP_CELL_MANAGE_WIDGET@](#)
[WP_CELL_TOGGLE@](#)

WP_CELL_TOGGLE@

Manages a toggle widget in the cell

Format WP_CELL_TOGGLE@(beadNum, ID)

Method [this.cell_toggle@](#)(beadNum, ID)

Arguments beadNum The bead number of the cell.
ID The cell ID.

Description In a form, this macro changes the state of the toggle in a cell. The toggle state is a reference to either a [CellToggleOn@](#) or [CellToggleOff@](#) glossary. If a glossary does not exist, the glossary is created as [x] or [] and placed at the beginning of the cell.

This macro is usually called by a double-click macro in a form cell.

See also [WP_CELL_RADIO_VALUE@](#)
[WP_CELL_LIST_BOX@](#)
[WP_CELL_MANAGE_WIDGET@](#)

WP_CELL_WIDGET_VALUE@

Checks if cell contains an *on* glossary field

Format flag = WP_CELL_WIDGET_VALUE@(beadNum)

Method [flag = this.cell_widget_value@](#)(beadNum)

Arguments beadNum The bead number of any bead in a cell.

Description Returns TRUE if the cell indicated by beadNum contains an *on* glossary field. That is, TRUE is returned if the field contains a toggle or radio button and the field is turned on. FALSE means can mean that the toggle or radio button is turned off. It could also mean that the cell does not contain one of these controls.

Use [WP_CELL_RADIO_VALUE@](#) to determine if a radio button is on.

An error is thrown if beadNum is not within a table.

WP_CELLS_RADIO@

Turns radio buttons on and off

Format WP_CELLS_RADIO@(CellBead, ID)

Arguments CellBead The cell-end bead of a cell. This value is returned by the macro WP_FIND_BEAD_OF_CELL@.
ID A cell ID assigned to a set of radio bead cells.

Description This macro turns on the radio widget in the target cell in a form. Radio widgets in all other cells with the same ID are turned off.

WP_CENTER@

Centers text

Format WP_CENTER@()

Method [this.center@](#)

Description Centers the current or selected text between the left and right margins.

WP_CHANGE_BARS@

Turns change bar mode on and off

Format WP_CHANGE_BARS@(compareDoc, style)

Method [this.change_bars@\(compareDoc, style\)](#)

Arguments compareDoc The name of the document to compare to the current document.
style Indicates where to place the change bars in the document. style can be:
0 The change bars appear in the left margin.
1 The change bars appear in the right margin.
2 The change bars appear in the outer margins.

Description Turns change bar mode on and off. If change bar mode is off and WP_CHANGE_BARS@ is called, then change bar mode is enabled. You must include compareDoc and style when turning on change bars. When change bar mode is enabled, the current document is compared with the specified document and change bars appear in the specified margin of the current document.

The change bars indicate lines in the current document that differ from lines in the comparative document. To turn off change bar mode, specify `WP_CHANGE_BARS@` without any arguments.

See also [WP_CHANGE_BARS_DISPLAY@](#)
[WP_CHANGE_BARS_REMOVE@](#)
[WP_CHANGE_BARS_WRITE@](#)

WP_CHANGE_BARS_DISPLAY@

Show a difference file

Format `WP_CHANGE_BARS_DISPLAY@(diffFile, style)`

Method `this.change_bars_display@(diffFile, style)`

Arguments

<code>diffFile</code>	The file containing the differences between two versions of a file.
<code>style</code>	Indicates where to place the change bars in the document. style can be: <ul style="list-style-type: none">0 The change bars appear in the left margin1 The change bars appear in the right margin2 The change bars appear in the outer margins

Description Displays a version of a file that shows the differences between the file and another version of the same file. This displayed version contains change bars and other symbols indicating where differences occurred.

The procedure for creating a difference file is as follows:

1. Write the two versions of the file to disk using `WP_CHANGE_BARS_WRITE@`.
2. Perform a shell diff on the two files in the following order:
`diff old_version current_version`

See also [WP_CHANGE_BARS@](#)
[WP_CHANGE_BARS_REMOVE@](#)
[WP_CHANGE_BARS_WRITE@](#)

WP_CHANGE_BARS_REMOVE@

Removes change bars

Format `WP_CHANGE_BARS_REMOVE@()`

Method `this.change_bars_remove@`

Description Removes change bars and other related symbols from a document.

See also [WP_CHANGE_BARS@](#)
[WP_CHANGE_BARS_DISPLAY@](#)
[WP_CHANGE_BARS_WRITE@](#)

WP_CHANGE_BARS_WRITE@

Writes the contents to disk so a *diff* can be performed

Format WP_CHANGE_BARS_WRITE@(tempFile)

Method [this.change_bars_write@\(tempFile\)](#)

Arguments tempFile A temporary file into which ELF will write a version of the current file.

Description Writes the contents of the current file into a temporary file. This file is in a format that the shell's diff command can use to compare this file with another.

See also [WP_CHANGE_BARS@](#)
[WP_CHANGE_BARS_DISPLAY@](#)
[WP_CHANGE_BARS_REMOVE@](#)

WP_CHANGE_LINK@

Changes the name of a file to which an external link is made

Format WP_CHANGE_LINK@(arrayOffset, newName)

Method [this.change_link@\(arrayOffset, newName\)](#)

Arguments arrayOffset The array element within the array returned by [WP_GET_LINKS@](#). This array contains the entry whose value you wish to change. For example, if you want to change the name for the first link returned by WP_GET_LINKS@, you provide 0 as the arrayOffset argument.

newName The name of the Graphics or Spreadsheets file that you want to link to the current Words document. newName must include the appropriate file name extension.

Description Links the specified Graphics or Spreadsheets file with the current Words document. The new linked file is substituted for an existing linked file as specified by arrayOffset.

See also [WP_CHANGE_LINK_INFO@](#)

[WP_GET_LINKS@](#)

[WP_SET_LINKS_INFO@](#)

WP_CHANGE_LINK_INFO@

Changes information for an external link

Format WP_CHANGE_LINK_INFO@(arrayOffset, format wp_link_info@ info)

Method [this.change_link_info@\(arrayOffset, format wp_link_info@ info\)](#)

Arguments arrayOffset The array element within the array returned by [WP_GET_LINKS@](#). This array contains the entry whose information you wish to change. For example, if you want to change the name for the first link returned by WP_GET_LINKS@, you provide 0 as the arrayOffset argument.

info The information you are changing.

Description Changes a link's information. This information is in the wp_link_info@ format, which is as follows:

```
format wp_link_info@
  name,           'string:  file name
  docType,        'int:     doc type returned by RECOGNIZE\_FILE@
  appType,        'int:     application type returned by RECOGNIZE_FILE@
  filterMacro,    'string:  import filter macro
  launchMacro,    'string:  macro to launch application
  viewname        'string:  viewname (for spreadsheet)
```

See also [WP_CHANGE_LINK@](#)
[WP_GET_LINKS@](#)
[WP_SET_LINKS_INFO@](#)

WP_CHECK_DOC_LEGALITY@

Checks the legality of the (beads in the) document

Format array = WP_CHECK_DOC_LEGALITY@(nErrs, flags, lineLen[, dumpFile])

Method [array = this.check_doc_legality@\(nErrs, flags, lineLen\[, dumpFile\]\)](#)

Arguments nErrs Indicates the maximum number of errors to report, and at what point to stop the reporting.

flags A Boolean value which, when set to TRUE, indicates that transient flags are also checked.

lineLen Maximum line length for error messages. If this value is 0, this parameter is ignored.

dumpFile The file into which the error messages are written.

Description Checks the legality of the current document and its globals. It returns a two-element array as follows:

- 0 the number of real errors reported
- 1 the array of error messages broken into multiple lines.

This macro performs the following functions:

- Checks the legality of the Words file just read in (e.g., one generated by the user).
- Checks the legality of the current document after an operation (for example, when debugging new ELF functions).

The flags parameter, if set to FALSE, will avoid checking flags and values that are set by the software upon reading in a file (that is, those items not actually in the input file, but derived from the contents of the file). Presumably, any errors in these flags are based upon errors in the file itself. Users reading in a file would only be interested in errors pertaining to the file, not "side-effects" that would only obscure the root of the problem.

The syntax of error reporting is as follows.

- An error message begins with the bead number. (The first bead is 1.)
- This is followed immediately by a colon, a space, and the text of the message.

If the overall message exceeds lineLen characters, it is continued on the following line with leading whitespace. An error that does not pertain to a specific bead will start with a colon. For example:

8: Paragraph bead has parent style of zero

22: This Cell End bead follows a previous Cell End bead in the
 same row that claimed to be the last Cell End bead of the row

: Limit of 2 errors has been reached; halting legality check

NOTE: A severely malformed Words file may cause a fatal Applixware error to occur.

WP_CLEAR_FIELD_ATTR_EDITS@

Clears preserved attribute edits in a Spreadsheets, Query, or Object field

Format WP_CLEAR_FIELD_ATTR_EDITS@()

Method [this.clear_field_attr_edits@](#)

Description Clears local field value attributed within a Spreadsheets of Data link object field.

WP_CLEAR_SELECTION@

Clears the current selection

Format WP_CLEAR_SELECTION@()

Method [this.clear_selection@](#)

Description This function changes the current selection into a cursor at the start of the former selection. It has no effect if the selection is already a cursor.

WP_CLEAR_STATUS_LINE@

Clears message in the Words status line

Format WP_CLEAR_STATUS_LINE@()

Method [this.clear_status_line@](#)

Description Clears the message displayed in the status line. After the status line is cleared, the status line will begin displaying normal status messages again.

WP_CMYK_TO_RGB@

Converts a color from one system to another

Format WP_CMYK_TO_RGB@(cyan, magenta, yellow, black, red, green, blue)

Method [this.cmyk_to_rgb@\(cyan, magenta, yellow, black, red, green, blue\)](#)

Arguments

cyan	The color's cyan component.
magenta	The color's magenta component.
yellow	The color's yellow component.
black	The color's black component.
red	The color's red component.
green	The color's green component.
blue	The color's blue component.

Description Changes a color from the CMYK system to the RGB system. The three RGB colors are set by this macro. That is, you only specify the four CMYK values.

See also [WP_HSB_TO_RGB@](#)
[WP_RGB_CMYK@](#)
[WP_RGB_HSB@](#)

WP_COLON_TO_TABLE@

Converts colon text to table text

Format WP_COLON_TO_TABLE@(exitIfHasTable)

Method [this.colon_to_table@\(exitIfHasTable\)](#)

Arguments exitIfHasTable
If set to TRUE, no conversion takes place.

Description Scans selected paragraphs, or the entire document if there are no selections. If a colon-delimited "Vertical Format" or "Horizontal Format" Aster*x 2.0 data file is detected, it is converted into a table. The last paragraph is not converted into a row.

If the entire document is being processed, the caller can optionally request that no conversion takes place if the document already contains a table. The Merge Print logic should use this option so that it won't get confused by 3.0 documents that happen to have lines with colons in them, but also has a properly-defined table.

WP_COLON_TO_TABLE@ is used primarily by the Aster*x 2.0 merge data file format, but is also useful for specifying tables in an imported ASCII document.

See also [WP_TEXT_TO_TABLE@](#)

WP_COLUMN_BREAK@

Inserts a column break

Format WP_COLUMN_BREAK@()

Method [this.column_break@](#)

Description Inserts a column break at the current cursor position. WP_COLUMN_BREAK@ is called by Insert ® Break ® Columns.

WP_COL_NUM@

Returns a bead's column number

Format colNum = WP_COL_NUM@(beadNum)

Method colNum = this.col_num@(beadNum)

Arguments beadNum The bead being passed.

Description Returns a column number of a cell in which passed bead resides. Column numbers are zero-based. That is, the number 0 is returned for the first column; 1 is returned for the second, and so on. An error is thrown if the passed bead is not in a cell.

See also [WP_ROW_NUM@](#)

WP_COMPARE_INDEX_ENTRIES@

Compares two arrays

Format result = WP_COMPARE_INDEX_ENTRIES@(array1, array2)

Method result = this.compare_index_entries@(array1, array2)

Arguments array1 The first array being compared.
array2 The second array being compared.

Description Compares two arrays and returns the highest level number at which array2 differs from array1. If array2 has more items than array1, array2 will differ at the level that array2 has and array1 does not, if not before.

Level numbers are 1-based; 0 is returned if array1 and array2 are identical. This macro assumes that array2 is not a subset of the array1 (i.e., it will either run out of items in array1, or find a difference between the items, but not run out of items in array2).

WP_CONVERT_OBJECT@

Converts a named object to Applixware format

Format WP_CONVERT_OBJECT@(name)

Method this.convert_object@(name)

Arguments name The name of the object being converted.

Description Using the converter associated with the object, converts the object into Ap-
plixware format.

WP_COPY@

Copies selected material to the clipboard

Format WP_COPY@()

Method [this.copy@](#)

Description Copies selected material from the current Words document and places this infor-
mation in the clipboard. If no material is selected, an error is thrown.
WP_COPY@ is called by Edit ® Copy.

See also [WP_COPY_STYLES@](#)
[WP_COPY_STYLES_FROM_FILE@](#)
[WP_CUT@](#)
[WP_PASTE@](#)

WP_COPY_BEADS@

Copies a set of beads within the document

Format range = WP_COPY_BEADS@(format wp_range@ range, format wp_location@
loc)

Method [range = this.copy_beads@\(format wp_range@ range, format wp_location@ loc\)](#)

Arguments range The range of the data being copied.
loc The location to which the data is copied.

Description Copies the information within range to location loc. The format of the range vari-
able is:

format wp_range@
 format wp_location@ start,
 format wp_location@ last

The format of loc is:

format wp_location@
 bead,
 offset

The range of the copied information is returned.

NOTE: Copying the wrong beads to the wrong location will cause severe problems.

See also [WP_COPY_CELL@](#)
[WP_COPY_ROW@](#)
[WP_DELETE_BEAD@](#)
[WP_DELETE_BEADS@](#)
[WP_DELETE_BEADS_IN_RANGE@](#)

WP_COPY_CELL@

Duplicates a cell

Format range = WP_COPY_CELL@(beads, nCopies, withMaterialFlag, beforeCellFlag)

Method range = this.copy_cell@(beads, nCopies, withMaterialFlag, beforeCellFlag)

Arguments

bead	A bead in the cell being copied
nCopies	The number of copies being made
withMaterialFlag	A Boolean value where TRUE means you are also copying the content of the cell. (FALSE means that you are only copying the structure of the cell.)
beforeCellFlag	A Boolean value where TRUE means that the new cells will appear before the original cell. (FALSE means that they will appear after.)

Description Copies the cell indicated by bead. The copies are placed before or after bead's cell (depending on the value of beforeCellFlag). When the cell is duplicated, it can also include the contents of bead's cell if withMaterialFlag is set to TRUE. nCopies specifies how many copies of the cell are made.

The range of the new cells is returned. If you ask for less than 1 copy, NULL is returned.

See also [WP_COPY_BEADS@](#)
[WP_COPY_ROW@](#)
[WP_DELETE_CELL@](#)
[WP_DELETE_CELLS@](#)

WP_COPY_COLORS@

Copies colors to clipboard

Format WP_COPY_COLORS@(colorArray)

Method [this.copy_colors@](#)(colorArray)

Arguments colorArray An array of color name strings.

Description Copies colors specified in colorArray from the current Words document to the clipboard. You can use this macro to copy colors between Words documents.

WP_COPY_GLOSSARIES@

Copies glossaries to clipboard

Format WP_COPY_GLOSSARIES@(glossArray)

Method [this.copy_glossaries@](#)(glossArray)

Arguments glossArray An array of glossary name strings.

Description Copies glossaries specified in glossArray from the current Words document to the clipboard. You can use this macro to copy glossaries between Words documents.

WP_COPY_ROW@

Duplicates a row

Format newRange = WP_COPY_ROW@(beadNum, nCopies, withMaterialFlag, beforeRowFlag)

Method [this.copy_row@](#)(beadNum, nCopies, withMaterialFlag, beforeRowFlag)

Arguments beadNum The number of a bead in the row being copied.

nCopies The number of copies being made.

withMaterialFlag

A Boolean value where TRUE means you are also copying the contents of the row's cells. (FALSE means that you are only copying the structure of the row's cells.)

beforeRowFlag

A Boolean value where TRUE means that the new rows will appear before the original row. (FALSE means that they will appear after.)

Description Copies the row indicated by beadNum. The copies are placed before or after beadNum's row (depending on the value of beforeRowFlag). When the row is duplicated, it can also include the contents of beadNum's row if withMaterialFlag is set to TRUE.

nCopies specifies how many copies of the row are made.

The range of the new rows is returned. If you ask for less than 1 copy, NULL is returned.

See also [WP_COPY_BEADS@](#)
[WP_COPY_CELL@](#)
[WP_DELETE_ROW@](#)

WP_COPY_STYLES@

Copies named styles to clipboard

Format WP_COPY_STYLES@(names)

Method [this.copy_styles@\(names\)](#)

Arguments names An array of style names.

Description Copies style information contained in the current document for each of the named styles to the clipboard.

See also [WP_COPY@](#)
[WP_COPY_STYLES_FROM_FILE@](#)
[WP_CUT@](#)
[WP_PASTE@](#)

WP_COPY_STYLES_FROM_FILE@

Copies named styles to clipboard

Format WP_COPY_STYLES_FROM_FILE@(file, names)

Method [this.copy_styles_from_file@\(file, names\)](#)

Arguments file The name of the file from which style information will be extracted.
names An array of style names.

widthPad If buttonWidth is 0, the button width is textWidth plus the value of widthPad (in mils).

buttonHeight The requested height of the button (in mils).

heightPad If buttonHeight is 0, the button height is textHeight plus the value of heightPad (in mils).

bevel Thickness of 3-D button shadowing (in mils).

Description Creates a button that can invoke an ELF macro. You can insert the file in a Words document using [WP_INSERT_LINK_FILE_FIELD@](#)

WP_CREATE_GLOSSARY@

Adds a glossary item

Format WP_CREATE_GLOSSARY@(name)

Method [this.create_glossary@\(name\)](#)

Arguments name The name by which the glossary information will be identified.

Description Copies the selected information into the glossary. This information will be referred to by name.

If a glossary with the same name already exists, the existing glossary item is overwritten.

Note that you cannot place individual cells in a glossary. If you try to place a cell into a glossary, it is inserted as a row.

If the information being placed into the glossary contains footnote fields, the footnote fields are deleted.

See also [WP_CREATE_GLOSSARY_FROM_BEADS@](#)
[WP_CREATE_GLOSSARY_FROM_STR@](#)

WP_CREATE_GLOSSARY_FROM_BEADS@

Creates or changes a glossary

Format WP_CREATE_GLOSSARY_FROM_BEADS@(name, format arrayof wp_bead@ beads)

Method [this.create_glossary_from_beads@\(name, format arrayof wp_bead@ beads\)](#)

Arguments name The name of the glossary being created or changed.
beads An array of beads that will become the glossary's contents.

Description Creates glossary name or overwrites it using an array of bead information. As this macro adds a Glossary Start and Glossary End bead, your bead information should not include these beads.

It is the programmer's responsibility to insure that the beads being inserted into the glossary constitute a *legal* collection of beads. For example, you cannot include a Cell End bead without including a Row Start bead; a Field Start bead cannot be included without a matching Field Split or Field End bead.

Note that you cannot place individual cells in a glossary. If you try to place a cell into a glossary, it is inserted as a row.

If the information being placed into the glossary contains footnote fields, the footnote fields are deleted.

See also [WP_CREATE_GLOSSARY@](#)
[WP_CREATE_GLOSSARY_FROM_STR@](#)

WP_CREATE_GLOSSARY_FROM_STR@

Creates or changes a glossary whose contents will be 'string'

Format WP_CREATE_GLOSSARY_FROM_STR@(name, string)

Method [this.create_glossary_from_str@\(name, string\)](#)

Arguments

name	The name of the glossary being created or changed.
string	The text of the glossary.

Description Creates glossary name or overwrites it using the contents of string.

Note that you cannot place individual cells in a glossary. If you try to place a cell into a glossary, it is inserted as a row.

If the information being placed into the glossary contains footnote fields, the footnote fields are deleted.

See also [WP_CREATE_GLOSSARY@](#)
[WP_CREATE_GLOSSARY_FROM_BEADS@](#)

WP_CREATE_INDEX@

Generates an index

Format beadRange = WP_CREATE_INDEX@(resetViewFlag, makeIndStart)

Method [beadRange = this.create_index@\(resetViewFlag, makeIndStart\)](#)

Arguments resetViewFlag
A Boolean value, which if set to TRUE, shuts off field display.
makeIndStart
The bead location at which the created index is inserted.

Description Creates a new index and removes the old index if it exists. Because *field view mode* can change page numbers, you can set the resetViewFlag to TRUE. This indicates that ELF should temporarily reformat the document so that fields are not displayed. After the index is created, the document is restored to its original state.

The beadRange of the newly created index is returned.

See also [WP_CREATE_TOC@](#)

WP_CREATE_TOC@

Generates a table of contents

Format WP_CREATE_TOC@(resetViewFlag, makeTocStart)

Method [this.create_toc@](#)(resetViewFlag, makeTocStart)

Arguments resetViewFlag
A Boolean value, which if set to TRUE, shuts off field display.
makeTocStart
The bead location at which the created table of contents is inserted.

Description Creates a new table of contents and removes the old table of contents if it exists. Because *field view mode* can change page numbers, you can set the resetViewFlag to TRUE. This indicates that ELF should temporarily reformat the document so that fields are not displayed. After the index is created, the document is restored to its original state.

Styles in support of the table of contents may be added if necessary. The bead string representing the table of contents is returned.

See also [WP_CREATE_INDEX@](#)

WP_CURRENT_PAGE@

Returns the page number of the current cursor's position

Format pageNum = WP_CURRENT_PAGE@()

Method [pageNum = this.current_page@](#)

Description Determines the position of the current cursor, then returns the page number of its location as an integer. This macro assumes that the first page of the document is page 1 and that you have not applied a starting page number to any section within the document. If you have applied starting page numbers, they are ignored.

See also [WP_GET_PAGE_OF_LOCATION@](#)
[WP_PAGE_NUMBER_OF_LOC@](#)
[WP_PAGE_NUMBER_STRS_OF_LOC@](#)
[WP_PAGE_OF_LOCATION@](#)

WP_CUT@

Cuts selected text placing it on the clipboard

Format WP_CUT@()

Method [this.cut@](#)

Description Cuts selected text from the current Words document and places the material in the clipboard. If no material is selected, an error occurs. WP_CUT@ is called by Edit ® Cut.

See also [WP_COPY@](#)
[WP_PASTE@](#)

WP_DEFER_SCREEN_UPDATE@

Defers screen updating

Format WP_DEFER_SCREEN_UPDATE@(deferFlag)

Method [this.defer_screen_update@\(deferFlag\)](#)

Arguments deferFlag A Boolean value which if set to TRUE indicates that Applixware Words will not update the screen.

Description If deferFlag is TRUE, places Applixware Words into a state where the Words window will not be updated until this command is re-executed with an argument of FALSE.

WP_DEFINE_SERIES@

Creates an ordered sequence

Format WP_DEFINE_SERIES@(format wp_series@ series)

Method [this.define_series@](#)(format wp_series@ series)

Arguments series A data structure containing the information defining a series.

Description Adds a new series definition to Words. The series argument is of type wp_series@. This format has the following definition:

format wp_series@

name,	'The name of the series.
style,	'An array of one-character strings. Each string represents the character displayed at the level
multiLevel,	'If TRUE, show multiple levels.
showFrom,	'An array, but the 1st element not used. Indicates the level to begin display, up to the current level.
separator,	'An array, but the 1st element not used. Specifies the separator characters to be placed between levels.
leaderText,	'The text that is placed before all of the series numbering.
trailerText	'The text that is placed after all of the series numbering.

See Utilities ® [Edit Numbered Series](#) for more information.

See also [WP_SERIES_ASTERISKS@](#)

[WP_SERIES_DAYS@](#)

[WP_SERIES_MONTHS@](#)

WP_DELETE@

Deletes selected material

Format WP_DELETE@()

Method [this.delete@](#)

Description Deletes all selected material in the current Words document. Deleted material is not saved in the clipboard. WP_DELETE@ is called by Edit ® Delete.

WP_DELETE_BEAD@

Deletes a bead

Format WP_DELETE_BEAD@(beadNum[, dontLegalizeFlag])

Method [this.delete_bead@](#)(beadNum[, dontLegalizeFlag])

Arguments beadNum The number of the bead you want to delete.

dontLegalizeFlag

A Boolean value which if set to TRUE, indicates that the bead is deleted as requested. If set to FALSE, which is the default, the bead may be adjust in minor ways before it is deleted. For example, it may be moved to preserve the last paragraph marker in a table cell.

Description Deletes bead from the document. This can be a dangerous operation.

NOTE: Deleting the wrong bead will cause severe problems. While setting the dontLegalizeFlag to TRUE helps, it does not prevent all cases of unbalanced bead deletion.

See also [WP_COPY_BEADS@](#)
[WP_DELETE_BEADS@](#)
[WP_DELETE_BEADS_IN_RANGE@](#)
[WP_INSERT_BEAD@](#)

WP_DELETE_BEADS@

Deletes the indicated beads

Format WP_DELETE_BEADS@(beadNum, count[, dontLegalizeFlag])

Method [this.delete_beads@](#)(beadNum, count[, dontLegalizeFlag])

Arguments beadNum The number of the bead to delete.

count The total number of beads to delete. count must be at least one.

dontLegalizeFlag

A Boolean value which if set to TRUE, indicates that the beads are deleted as requested. If set to FALSE, which is the default, the beads may be adjusted in minor ways before they are deleted. For example, they may be moved to preserve the last paragraph marker in a table cell.

Description This function accepts a starting bead and a count, and deletes the indicated beads.

NOTE: Deleting the wrong beads will cause severe problems. This can be a very dangerous operation. While setting the dontLegalizeFlag to TRUE helps, it does not prevent all cases of unbalanced bead deletion.

See also [WP_COPY_BEADS@](#)
[WP_DELETE_BEAD@](#)
[WP_DELETE_BEADS_IN_RANGE@](#)
[WP_INSERT_BEADS@](#)

WP_DELETE_BEADS_IN_RANGE@

Deletes beads in a range

Format loc = WP_DELETE_BEADS_IN_RANGE@(format wp_range@ range[, dontLegalizeFlag])

Method loc = this.delete_beads_in_range@(format wp_range@ range[, dontLegalizeFlag])

Arguments range The range of beads being deleted.
dontLegalizeFlag A Boolean value which if set to TRUE, indicates that the beads are deleted as requested. If set to FALSE, which is the default, the beads may be adjusted in minor ways before they are deleted. For example, the range may be extended to preserve the last paragraph marker in a table cell.

Description Deletes the material in the range, and returns the location of the resulting cursor. The new location of the cursor is returned (as a format [wp_location@](#) variable).

NOTE: Deleting the wrong beads will cause severe problems. While setting the dontLegalizeFlag to TRUE helps, it does not prevent all cases of unbalanced bead deletion.

See also [WP_COPY_BEADS@](#)
[WP_DELETE_BEAD@](#)
[WP_DELETE_BEADS@](#)

WP_DELETE_CELL@

Deletes one cell

Format WP_DELETE_CELL@(bead)

Method this.delete_cell@(bead)

Arguments bead A bead in a cell.

Description Deletes one cell from a table. bead is any bead in the cell.

See also [WP_COPY_CELL@](#)
[WP_DELETE_CELLS@](#)
[WP_DELETE_SELECTED_CELLS@](#)

WP_DELETE_CELLS@

Deletes cells

Format WP_DELETE_CELLS@(bead, count)

Method [this.delete_cells@\(bead, count\)](#)

Arguments bead A bead in the first cell to be deleted.
count The total number of cells to delete. count must be greater than or equal to 0.

Description Deletes one or more cells. The deletion begins with the cell containing bead and continues until the number of cells indicated is deleted or the end of the row is reached.

See also [WP_COPY_CELL@](#)
[WP_DELETE_CELL@](#)
[WP_DELETE_SELECTED_CELLS@](#)
[WP_INSERT_CELLS@](#)

WP_DELETE_CHAR_KEY@

Deletes character to the right of cursor

Format WP_DELETE_CHAR_KEY@()

Method [this.delete_char_key@](#)

Description Deletes the character to the right of the cursor. (By default, Applixware binds the delete key to a function that deletes to the left of the cursor. This is because some keyboards do not also have a backspace key. If your keyboard has both keys, then you might bind this macro to the delete key.)

See also [WP_BACKSPACE_KEY@](#)

WP_DELETE_COLOR@

Deletes a color

Format WP_DELETE_COLOR@(colorName)

Method [this.delete_color@](#)(colorName)

Arguments colorName The name of the color being deleted.

Description Deletes a color from within Words.

WP_DELETE_DOC@

Deletes the current Words document

Format WP_DELETE_DOC@()

Method [this.delete_doc@](#)

Description Deletes the current Words document from disk and closes the Words window.

WP_DELETE_DOC_VAR@

Deletes the document variable

Format flag = WP_DELETE_DOC_VAR@(name)

Method [flag = this.delete_doc_var@](#)(name)

Arguments name The name of the document variable being deleted.

Description Deletes a document variable from the document. WP_DELETE_DOC_VAR@ returns TRUE if name actually exists. If name does not exist, FALSE is returned and no action is performed.

See also [WP_DELETE_SESSION_VAR@](#)

WP_DELETE_GLOSSARY@

Deletes a glossary

Format WP_DELETE_GLOSSARY@(name)

Method [this.delete_glossary@](#)(name)

Arguments name The name of the glossary being deleted.

Description Deletes a glossary from a Words document.

WP_DELETE_KEEP@

Removes a keep

Format WP_DELETE_KEEP@()

Method [this.delete_keep@](#)

Description Removes a *keep* that prevents two paragraphs from being split by a page break.

WP_DELETE_LINE@

Deletes the current line

Format WP_DELETE_LINE@()

Method [this.delete_line@](#)

Description Deletes the current line in the current Words document.

See also [WP_DELETE_TO_END_OF_LINE@](#)

WP_DELETE_MARKER_BEAD@

Removes a marker bead

Format WP_DELETE_MARKER_BEAD@(name)

Method [this.delete_marker_bead@](#)(name)

Arguments name The name of the marker bead.

Description Removes a marker bead from the current document.

See also [WP_DELETE_BEAD@](#)

WP_DELETE_OBJECT@

Deletes an object

Format WP_DELETE_OBJECT@(name)

Method [this.delete_object@\(name\)](#)

Arguments name The name of the object being deleted.

Description Deletes object name from the Words document. All references to name are also deleted.

See also [WP_EDIT_OBJECT@](#)

WP_DELETE_ROW@

Deletes a row

Format WP_DELETE_ROW@(beadNum)

Method [this.delete_row@\(beadNum\)](#)

Arguments beadNum The bead number of any bead in the row.

Description Deletes the entire row. The beadNum parameter can refer to any bead in the row.

See also [WP_COPY_ROW@](#)

WP_DELETE_SELECTED_CELLS@

Deletes cells

Format WP_DELETE_SELECTED_CELLS@()

Method [this.delete_selected_cells@](#)

Description Removes selected cells. Note that if more than one cell is selected, the [WP_DELETE@](#) macro empties these cells but does not delete them.

See also [WP_COPY_CELL@](#)
[WP_DELETE_CELL@](#)
[WP_DELETE_CELLS@](#)

WP_DELETE_SELECTION_MARKS@

Deletes selection marks

Format WP_DELETE_SELECTION_MARKS@(name)

Method [this.delete_selection_marks@\(name\)](#)

Arguments name The name used in WP_SET_SELECTION_MARKS@.

Description Deletes the two markers named name placed into a table with the [WP SET SELECTION MARKS@](#) macro.

WP_DELETE_SERIES@

Deletes a numbered series

Format WP_DELETE_SERIES@(name)

Method [this.delete_series@\(name\)](#)

Arguments name The name of the series being deleted.

Description Deletes a named numbered series from your document. If a field reference to the series exists, Words will still treat it as a numbered series reference. However, this "unregistered" numbered series will be represented as a simple numeric series; i.e., 1, 2, 3, and so on.

WP_DELETE_SESSION_VAR@

Deletes a session variable

Format flag = WP_DELETE_SESSION_VAR@(name)

Method [flag = this.delete_session_var@\(name\)](#)

Arguments name The name of a session variable.

Description Deletes a session variable. If the session variable exists, WP_DELETE_SESSION_VAR@ returns TRUE.

See also [WP_DELETE_DOC_VAR@](#)

WP_DELETE_STYLE@

Deletes a style

Format WP_DELETE_STYLE@(name, inheritFlag, noReformatFlag)

Method [this.delete_style@](#)(name, inheritFlag, noReformatFlag)

Arguments

name	The name of the style being deleted.
inheritFlag	A Boolean value, which if TRUE indicates that paragraphs of this style become paragraphs of this style's parent. If FALSE, paragraphs of this style are changed to the top-level style, which is usually Normal.
noReformatFlag	A Boolean value, which if TRUE specifies that the document is not reformatted (usually because there are more styles being added or deleted). If FALSE, the document is reformatted to incorporate the deleted style(s).

Description Deletes a style from a document's style list. If a paragraph has this style, the style can be changed to that of its parent (inheritFlag is TRUE) or to Normal (inheritFlag is FALSE).

If you are performing more than one style change or are executing windowless, you will not want to reformat the document based on the style change. In such cases, set the noReformatFlag to FALSE until you make the last change.

See also [WP_DELETE_UNUSED_STYLES@](#)

WP_DELETE_TABSTOP@

Deletes a tab stop

Format WP_DELETE_TABSTOP@(milpos)

Method [this.delete_tabstop@](#)(milpos)

Arguments milpos The position of the tab stop in mils. (1000 mils = 1 inch)

Description Deletes the tab at this position for the current or all selected paragraphs. If the tab is a local tab, it is deleted. If is inherited, all other inherited tabs become local tabs.

WP_DELETE_TAG@

Deletes a tag

Format WP_DELETE_TAG@(name)

Method [this.delete_tag@\(name\)](#)

Arguments name The name of the tag being deleted.

Description Removes a tag's name and the information associated with it from Word's tag list.

WP_DELETE_TO_END_OF_LINE@

Deletes from the cursor position to the end of the line

Format WP_DELETE_TO_END_OF_LINE@()

Method [this.delete_to_end_of_line@](#)

Description Deletes from the current cursor position to the end of the line. WP_DELETE_TO_END_OF_LINE@ is called by Keys ® Delete EOL and is bound to CTRL-K.

See also [WP_DELETE_LINE@](#)

WP_DELETE_UNUSED_STYLES@

Deletes unused styles

Format WP_DELETE_UNUSED_STYLES@()

Method [this.delete_unused_styles@](#)

Description Delete styles that are not used by any paragraphs, or are parents of styles with paragraphs. WP_DELETE_UNUSED_STYLES@ does not delete the top level style if it is unused. (This top level style should be Normal.)

See also [WP_DELETE_STYLE@](#)

WP_DELETE_WORD@

Deletes the word at the current cursor position

Format WP_DELETE_WORD@()

Method [this.delete_word@](#)

Description If the cursor is in a space, the first word following the space is deleted. If there is a selection, all words containing selected characters are deleted. In either case, the trailing space following the last word is also deleted.

WP_DELETE_WORD@ is called by Keys ® Delete Word.

WP_DEMOTE@

Demotes a section or numbered list item one level

Format WP_DEMOTE@()

Method [this.demote@](#)

Description Causes the paragraph in which the cursor resides to be demoted to the next level. For example, if you demote a level one paragraph, it becomes a level two paragraph. WP_DEMOTE@ is called by Edit ® Demote.

This macro affects the display of numbered series within the paragraph and possibly the paragraph's left indent.

See also [WP_PROMOTE@](#)

WP_DISPLAY_LOCATION@

Scrolls the screen to a location

Format WP_DISPLAY_LOCATION@(format wp_location@ loc)

Method [this.display_location@\(format wp_location@ loc\)](#)

Arguments loc A location in the document.

Description Scrolls the document so that location loc is displayed.

See also [WP_GET_PAGE_OF_LOCATION@](#)

WP_DISPLAY_LOC_OF_CURSOR@

Scrolls the document to display the cursor position

Format WP_DISPLAY_LOC_OF_CURSOR@()

Method [this.display_loc_of_cursor@](#)

Description Scrolls the screen display so that the cursor's position is displayed.

See also [WP_DISPLAY_PAGE_OF_CURSOR@](#)
[WP_LOCATION_OF_CURSOR@](#)

WP_DISPLAY_PAGE_OF_CURSOR@

Scrolls to the page containing the cursor

Format WP_DISPLAY_PAGE_OF_CURSOR@()

Method [this.display_page_of_cursor@](#)

Description Scrolls the display so that the top of the page containing the cursor is displayed at the top of the screen.

See also [WP_DISPLAY_LOC_OF_CURSOR@](#)

WP_DOC_IS_FORM@

Specifies whether document is a form

Format WP_DOC_IS_FORM@(flag)

Method [this.doc_is_form@\(flag\)](#)

Arguments flag A Boolean value, where TRUE means the document automatically opens in forms mode.

Description Specifies whether the Words document is a form.

See also [WP_GET_FORM_MODE@](#)
[WP_SET_FORM_MODE@](#)

WP_DOWN_ARROW_KEY@

Moves the cursor down one line

Format WP_DOWN_ARROW_KEY@()

Method [this.down_arrow_key@](#)

Description Moves the current cursor position down one line. This macro is usually bound to the down arrow key on alternate keyboards.

See also [WP_UP_ARROW_KEY@](#)

WP_EDIT_OBJECT@

Launches an application to edit an object

Format id = WP_EDIT_OBJECT@(objectname)

Method [id = this.edit_object@\(objectname\)](#)

Arguments objectname The name of the object that is being edited.

Description Opens the application specified for the object and loads the object into the application so that it can be edited.

The task id for object editor's process is returned.

See also [WP_DELETE_OBJECT@](#)

WP_ENLARGE_VIEW_SCALE@

Zooms view display larger

Format WP_ENLARGE_VIEW_SCALE@()

Method [this.enlarge_view_scale@](#)

Description Increases the viewing size of the displayed document. For example, if you are viewing the document at 120%, this command would increase the viewing size to 150% of normal.

See also [WP_VIEW_SCALE@](#)

WP_ENTER_CHAR@

Types a string of ASCII characters in any font at the cursor position

Format WP_ENTER_CHAR@(string[, font])

Method [this.enter_char@](#)(string[, font])

Arguments string A string of ASCII characters to be typed in one font. For each character that you cannot type directly from your keyboard, string must consist of either the corresponding ISO-Latin character, or a nested call to NUM_TO_STRING@. See the "Applixware Character Codes" table that follows for a list of all ASCII characters and their corresponding hexadecimal, ISO-Latin, and Symbol/Zapf Dingbats values.

font If string consists of ISO-Latin characters and font is set to either NULL or a number other than 0-10, string defaults to the default type style at the cursor position. The following table shows the font codes:

Code	Font	ELF Definition in WP_.am
0	Times Roman	WP#FONT#TIMES_
1	ITC Bookman	WP#FONT#BOOKMAN_
2	New Century Schlbk	WP#FONT#SCHOOL_
3	Zapf Chancery	WP#FONT#CHANCERY_
4	Palatino	WP#FONT#PALATINO_
5	Courier	WP#FONT#COURIER_
6	Helvetica	WP#FONT#HELVETICA_
7	Helvetica Narrow	WP#FONT#HELVNARROW_
8	Helvetica	WP#FONT#AVANT_
9	Symbol	WP#FONT#SYMBOL_
10	Zapf Dingbats	WP#FONT#DINGBATS_

Description Types a string of ASCII characters in one typeface. WP_ENTER_CHAR@ allows you to type characters that are not directly supported by the keys on your keyboard.

The table that follows is divided into three columns of ASCII characters where each character in a given row can be converted to another character on the same row by applying the appropriate "ISO-Latin," "Symbol" or "Zapf Dingbats" attribute. The first column displays the hexadecimal value that corresponds to the set of three characters.

Any ASCII character can be converted from its hexadecimal form by nesting within string a call to NUM_TO_STRING@. You may prefer to use the easier method whenever possible: if the character you want is in either the Symbol or Zapf Dingbats column of the table that follows, and that same row's ISO-Latin

entry is a character that you can type directly from your keyboard, pass that ISO-Latin character directly in the string argument.

NUM_TO_STRING@ requires that you precede the hexadecimal character code with ``0x" or ``0X" (a zero followed by an ``x"). For example, to type the character ``ÿ" (the last character in the Zapf Dingbats column), enter:

WP_ENTER_CHAR@(NUM_TO_STRING@(0xff), 10)

WP_ENTER_CHAR@ cannot be used in dialog boxes. WP_ENTER_CHAR@ is called by Insert ® Special Characters. All Symbol characters are Adobe-font-specific; all Zapf Dingbat characters are Applixware font-specific.

See also **NUM TO STRING@**.

Character Codes. If you select this option, be prepared to wait. Displaying this topic takes considerable time. Also, if the table columns overlap, you can resize your help window to make it larger.

CHARACTER CODES

Dec. Code	Hex Code	ISO Latin	Sym- bol	Z. Ding- bat
00-08	00-08	☐UNDEFINED		
09	09	☐TAB		
10	0a	☐LINE BREAK		
11-31	0b-1f	☐UNDEFINED		
32	20	☐SPACE		
33	21	!	!	!
34	22	"	"	"
35	23	#	#	#
36	24	\$	\$	\$
37	25	%	%	%
38	26	&	&	&
39	27	'	'	'
40	28	(((
41	29)))
42	2a	*	*	*
43	2b	+	+	+
44	2c	,	,	,
45	2d	-	-	-
46	2e	.	.	.
47	2f	/	/	/
48	30	0	0	0

49	31	1	1	1
50	32	2	2	2
51	33	3	3	3
52	34	4	4	4
53	35	5	5	5
54	36	6	6	6
55	37	7	7	7
56	38	8	8	8
57	39	9	9	9
58	3a	:	:	:
59	3b	;	;	;
60	3c	<	<	<
61	3d	=	=	=
62	3e	>	>	>
63	3f	?	?	?
64	40	@	@	@
65	41	A	A	A
66	42	B	B	B
67	43	C	C	C
68	44	D	D	D
69	45	E	E	E
70	46	F	F	F
71	47	G	G	G
72	48	H	H	H

73	49	I	I	I
74	4a	J	J	J
75	4b	K	K	K
76	4c	L	L	L
77	4d	M	M	M
78	4e	N	N	N
79	4f	O	O	O
80	50	P	P	P
81	51	Q	Q	Q
82	52	R	R	R
83	53	S	S	S
84	54	T	T	T
85	55	U	U	U
86	56	V	V	V
87	57	W	W	W
88	58	X	X	X
89	59	Y	Y	Y
90	5a	Z	Z	Z
91	5b	[[[
92	5c	\	\	\
93	5d]]]
94	5e	^	^	^
95	5f	_	_	_
96	60	`	`	`
97	61	a	a	a
98	62	b	b	b
99	63	c	c	c
100	64	d	d	d
101	65	e	e	e
102	66	f	f	f
103	67	g	g	g
104	68	h	h	h
105	69	i	i	i
106	6a	j	j	j
107	6b	k	k	k
108	6c	l	l	l
109	6d	m	m	m
110	6e	n	n	n
111	6f	o	o	o
112	70	p	p	p
113	71	q	q	q
114	72	r	r	r
115	73	s	s	s
116	74	t	t	t
117	75	u	u	u

118	76	v	v	v
119	77	w	w	w
120	78	x	x	x
121	79	y	y	y
122	7a	z	z	z
123	7b	{	{	{
124	7c			
125	7d	}	}	}
126	7e	~	~	~
127	7f	☐UNDEFINED		
128-159	80-9f	☐UNDEFINED		
160	a0	HARD SPACE		
161	a1	ı	ı	ı
162	a2	ç	ç	ç
163	a3	£	£	£
164	a4	¤	¤	¤
165	a5	¥	¥	¥
166	a6	ı	ı	ı
167	a7	§	§	§
168	a8	¨	¨	¨
169	a9	©	©	©
170	aa	ª	ª	ª
171	ab	«	«	«
172	ac	¬	¬	¬
173	ad	-	-	-
174	ae	®	®	®
175	af	¯	¯	¯
176	b0	°	°	°
177	b1	±	±	±
178	b2	²	²	²
179	b3	³	³	³
180	b4	´	´	´
181	b5	µ	µ	µ
182	b6	¶	¶	¶
183	b7	·	·	·
184	b8	¸	¸	¸
185	b9	¹	¹	¹
186	ba	º	º	º
187	bb	»	»	»
188	bc	¼	¼	¼
189	bd	½	½	½
190	be	¾	¾	¾
191	bf	¿	¿	¿
192	c0	À	À	À

193	c1	Á	Á	Á
194	c2	Â	Â	Â
195	c3	Ã	Ã	Ã
196	c4	Ä	Ä	Ä
197	c5	Å	Å	Å
198	c6	Æ	Æ	Æ
199	c7	Ç	Ç	Ç
200	c8	È	È	È
201	c9	É	É	É
202	ca	Ê	Ê	Ê
203	cb	Ë	Ë	Ë
204	cc	Ì	Ì	Ì
205	cd	Í	Í	Í
206	ce	Î	Î	Î
207	cf	Ï	Ï	Ï
208	d0	Ð	Ð	Ð
209	d1	Ñ	Ñ	Ñ
210	d2	Ò	Ò	Ò
211	d3	Ó	Ó	Ó
212	d4	Ô	Ô	Ô
213	d5	Õ	Õ	Õ
214	d6	Ö	Ö	Ö
215	d7	×	×	×
216	d8	Ø	Ø	Ø
217	d9	Ù	Ù	Ù
218	da	Ú	Ú	Ú
219	db	Û	Û	Û
220	dc	Ü	Ü	Ü
221	dd	Ý	Ý	Ý
222	de	Þ	Þ	Þ
223	df	ß	ß	ß
224	e0	à	à	à
225	e1	á	á	á
226	e2	â	â	â

227	e3	ã	ã	ã
228	e4	ä	ä	ä
229	e5	å	å	å
230	e6	æ	æ	æ
231	e7	ç	ç	ç
232	e8	è	è	è
233	e9	é	é	é
234	ea	ê	ê	ê
235	eb	ë	ë	ë
236	ec	ì	ì	ì
237	ed	í	í	í
238	ee	î	î	î
239	ef	ï	ï	ï
240	f0	ð	ð	ð
241	f1	ñ	ñ	ñ
242	f2	ò	ò	ò
243	f3	ó	ó	ó
244	f4	ô	ô	ô
245	f5	õ	õ	õ
246	f6	ö	ö	ö
247	f7	÷	÷	÷
248	f8	ø	ø	ø
249	f9	ù	ù	ù
250	fa	ú	ú	ú
251	fb	û	û	û
252	fc	ü	ü	ü
253	fd	ý	ý	ý
254	fe	þ	þ	þ
255	ff	ÿ	ÿ	ÿ

WP_ENTER_CHARS_FACE@

Inserts a character

Format WP_ENTER_CHARS_FACE@(code[, fontName])

Method [this.enter_chars_face@](#)(code[, fontName])

Arguments code The character code, which is an integer.
fontName The font code, which is an integer, as follows:

- 0 WP#FONT#TIMES_
Times Roman
- 1 WP#FONT#BOOKMAN_
ITC Bookman
- 2 WP#FONT#SCHOOL_
New Century Schl.
- 3 WP#FONT#CHANCERY_
Zapf Chancery
- 4 WP#FONT#PALATINO_
Palatino
- 5 WP#FONT#COURIER_
Courier
- 6 WP#FONT#HELVETICA_
Helvetica
- 7 WP#FONT#HELVNARROW_
Helvetica Narrow
- 8 WP#FONT#AVANT_
Avante Garde
- 9 WP#FONT#SYMBOL_
Symbol
- 10WP#FONT#DINGBATS_
Zapf Dingbats

Description Enters a character at the cursor position. This command differs from WP_ENTER_CHAR@ in that you are entering a character's integer value rather than the character as an argument. See [WP_ENTER_CHAR@](#) for a listing of these integer values.

WP_ENTER_HELP_MODE@

Puts Words in *Help On Context* mode

Format WP_ENTER_HELP_MODE@()

Method [this.enter_help_mode@](#)

Description Places Word into a mode where selecting items in the menu bar or in the *ExpressLine* allows the user to obtain help on these items (if the information exists). After choosing On Context, your cursor changes to a question mark. You can then click anywhere on a menu item or *ExpressLine* button to obtain help.

WP_ENTER_NEWLINE@

Inserts a line break at the current cursor position

Format WP_ENTER_NEWLINE@()

Method [this.enter_newline@](#)

Description Breaks the current line at the current cursor position, without splitting the line into separate paragraphs. The break is represented by a line-break marker (¶).

WP_ENTER_NEWLINE@ is called by Insert ¶ Break. The same kind of line break can be achieved by inserting a new-line character (\n) within a WP_ENTER_TEXT@ string. Line-break markers are visible only when WP_TOGGLE_CONTROLS_DISPLAY@ (View ¶ Breaks) is on.

WP_ENTER_TEXT@

Writes text into the document

Format WP_ENTER_TEXT@(text)

Method [this.enter_text@\(text\)](#)

Arguments text The string of text to type.

Description Types a string of text, and leaves the cursor at the end of the text just entered. This function can handle strings as long as a whole paragraph. It also handles new-line (\n) characters. In any Words document, WP_ENTER_TEXT@ is preferred to the TYPE@ function: it is faster and can accommodate longer strings of text.

WP_ENTER_TEXT@ is never recorded in keystroke recordings; TYPE@ is.

WP_EQN_GET_CURRENT_ALIGN@

Returns the equation alignment type

Format align = WP_EQN_GET_CURRENT_ALIGN@()

Method align = [this.eqn_get_current_align@](#)

Description Returns the equation pile alignment, which is one of the following values:

- 0 Left
- 1 Center
- 2 Right
- 3 At =
- 4 At Marker
- 5 At .

See also [WP_EQN_SET_ALIGN@](#)

WP_EQN_GET_DFLT_SIZE_INFO@

Returns the default size information

Format arrayof5 sizeInfo = WP_EQN_GET_DFLT_SIZE_INFO@()

Method arrayof5 sizeInfo = this.eqn_get_dflt_size_info@

Description Returns a five-element array describing the default point sizes used when displaying equation elements as follows:

sizeInfo[0] Full (normal) text size
sizeInfo[1] Subscript/superscript size
sizeInfo[2] Sub-Subscript/superscript size
sizeInfo[3] Symbol Size
sizeInfo[4] Sub-symbol size

See also [WP_EQN_GET_SIZE_INFO@](#)
[WP_EQN_SET_SIZE_INFO@](#)

WP_EQN_GET_DFLT_SPACING_INFO@

Returns the default spacing information

Format arrayof6 spaceInfo = WP_EQN_GET_DFLT_SPACING_INFO@()

Method arrayof6 spaceInfo = this.eqn_get_dflt_spacing_info@

Description Returns a six-element array describing the default spacing used when displaying equation elements as follows:

spaceInfo[0] Line spacing
spaceInfo[1] Matrix row spacing
spaceInfo[2] Matrix column spacing
spaceInfo[3] Superscript height
spaceInfo[4] Subscript height
spaceInfo[5] Limit height

See also [WP_EQN_GET_SPACING_INFO@](#)
[WP_EQN_GET_SPACING_INFO@](#)

WP_EQN_GET_MATRIX_INFO@

Returns Equation Editor matrix information

Format arrayof6 matrixInfo = WP_EQN_GET_MATRIX_INFO@()

Method arrayof6 matrixInfo = this.eqn_get_matrix_info@

Description Returns a six-element array describing the layout of matrix information, as follows:

matrixInfo[0] Rows
matrixInfo[1] Columns
matrixInfo[2] Row alignment
 0 Top
 1 Center
 2 Bottom
 3 Baseline
matrixInfo[3] Column alignment
 0 Left
 1 Center
 2 Right
 3 At =
 4 At marker
 5 At .
matrixInfo[4] Equal row height flag
matrixInfo[5] Equal column height flag

See also [WP_EQN_SET_MATRIX_INFO@](#)

WP_EQN_GET_SIZE_INFO@

Returns equation size information

Format arrayof5 sizeInfo = WP_EQN_GET_SIZE_INFO@()

Method arrayof5 sizeInfo = this.eqn_get_size_info@

Description Returns a five-element array describing the point sizes used when displaying equation elements as follows:

sizeInfo[0] Full (normal) text size
sizeInfo[1] Subscript/superscript size
sizeInfo[2] Sub-Subscript/superscript size
sizeInfo[3] Symbol Size
sizeInfo[4] Sub-symbol size

See also [WP_EQN_GET_DFLT_SIZE_INFO@](#)
[WP_EQN_SET_SIZE_INFO@](#)

WP_EQN_GET_SPACING_INFO@

Returns current spacing information

Format arrayof6 spacingInfo = WP_EQN_GET_SPACING_INFO@()

Method arrayof6 spacingInfo = this.eqn_get_spacing_info@

Description Returns a six-element array describing the spacing used when displaying equation elements as follows:

spacingInfo[0] Line spacing
spacingInfo[1] Matrix row spacing
spacingInfo[2] Matrix column spacing
spacingInfo[3] Superscript height
spacingInfo[4] Subscript height
spacingInfo[5] Limit height

See also [WP_EQN_GET_DFLT_SPACING_INFO@](#)
[WP_EQN_SET_SPACING_INFO@](#)

WP_EQN_INSERT_ALIGNMARKER@

Inserts an alignment marker

Format WP_EQN_INSERT_ALIGNMARKER@()

Method this.eqn_insert_alignmarker@

Description Inserts an align marker that allows you to align equation piles. This function is bound to Insert ® Symbol ® Insert Align Marker in the Applixware Equations.

WP_EQN_INSERT_SYMBOL_CODE@

Inserts a symbol in Applixware Equations

Format WP_EQN_INSERT_SYMBOL_CODE@(fontName, code)

Method this.eqn_insert_symbol_code@(fontName, code)

Arguments fontName The font code, which is an integer, as follows:

- 0 WP#FONT#TIMES_
Times Roman
- 1 WP#FONT#BOOKMAN_
ITC Bookman
- 2 WP#FONT#SCHOOL_
New Century Schl.
- 3 WP#FONT#CHANCERY_
Zapf Chancery
- 4 WP#FONT#PALATINO_
Palatino
- 5 WP#FONT#COURIER_
Courier
- 6 WP#FONT#HELVETICA_
Helvetica
- 7 WP#FONT#HELVNARROW_
Helvetica Narrow
- 8 WP#FONT#AVANT_
Avant Garde
- 9 WP#FONT#SYMBOL_
Symbol
- 10WP#FONT#DINGBATS_
Zapf Dingbats

code The character code, which is an integer. The number may be a decimal or hexadecimal number. If code is hexadecimal, the hexadecimal number must be preceded by "0x" or "0X" (a zero followed by an "x").

Description The macro inserts a symbol in Applixware Equations at the current cursor position.

See also **CHARACTER CODES**. Note that if you select this option, be prepared to wait. Displaying this topic takes considerable time. Also, you may need to resize your window to view all of the character codes.

WP_ENTER_CHARS_FACE@

WP_EQN_INSERT_TEMPLATE@

Inserts an equation template

Format format wp_location@ newLoc = WP_EQN_INSERT_TEMPLATE@(type, subtype, p1, p2, p3, p4, p5, p6)

Method format wp_location@ newLoc = this.eqn_insert_template@(type, subtype, p1, p2, p3, p4, p5, p6)

Arguments type & subtype One of the following:

- 0 Empty template
 - 0 Empty deletable
 - 1 Empty nondeletable
- 1 Symbol template
 - 0 simple character from variable font
 - 1 equals character (=)
 - 2 relational operator
 - 3 embellishments like ', ", `
 - 4 Space adjustments between templates
 - 5 Operators like +, -, *
 - 6 Character is digit
 - 7 Different types of arrows
 - 8 Logical symbols
 - 9 Set theory symbols
 - 10 Lowercase Greek symbols
 - 11 Uppercase Greek symbols
 - 12 Miscellaneous symbols
 - 13 Function names
- 2 Fence template
 - 0 Parentheses
 - 1 Left parentheses
 - 2 Right parentheses
 - 3 Braces
 - 4 Left brace
 - 5 Right brace
 - 6 Brackets
 - 7 Left bracket
 - 8 Right bracket
 - 9 Vertical Bars
 - 10 Left vertical bar
 - 11 Right vertical bar
 - 12 Double vertical bars
 - 13 Left double vertical bars
 - 14 Right double vertical bars
 - 15 Single overbar
 - 16 Double overbar
 - 17 Single underbar
 - 18 Double underbar
 - 19 Floor ($\lfloor \dots \rfloor$)
 - 20 Left floor
 - 21 Right floor
 - 22 Ceiling
 - 23 Left ceiling
 - 24 Right ceiling
 - 25 Bracket 2 Left
 - 26 Bracket 2 Right
 - 27 Bracket opposite
 - 28 Parenthesis/bracket

- 29 Bracket/parenthesis
- 4 Fraction or radicals template
 - 0 up OVER down
 - 1 sub up OVER sub down
 - 2 up SLASH down
 - 3 sup up SLASH sub down
 - 4 80 degrees bar up 80 SLASH down
- 5 Square root template
 - 0 Square root of
 - 1 nth root of
 - 2 Curvy square root symbol
 - 3 Curvy nth root
 - 4 Body over square root
- 6 Subscript and superscript template
 - 0 Subscript only
 - 1 Superscript only
 - 2 Superscript and subscript
 - 3 Superscript on left side
 - 4 Subscript on left side
 - 5 sub and superscripts on left
 - 6 Superscript on top
 - 7 Subscript on top
 - 8 Both on top
- 7 Summation template
 - 0 Summation with no limits
 - 1 Up limit
 - 2 Down limit
 - 3 Up and down limits
 - 4 Down limit as subscript
 - 5 Up limit as superscript
 - 6 Up and down limits as superscript and subscript
- Integrals
 - 7 Integration with no limits
 - 8 Up limit
 - 9 down limit
 - 10 Up and down limits
 - 11 Up limit as superscript
 - 12 Down limit as subscript
 - 13 Up and down limits as superscript and subscript
- 2 Integrals
 - 14 Integration with no limits
 - 15 Up limit
 - 16 down limit
 - 17 Up and down limits
 - 18 Up limit as superscript
 - 19 Down limit as subscript

- 20 Up and down limits as superscript and subscript
- 3 Integrals
 - 21 Integration with no limits
 - 22 Up limit
 - 23 down limit
 - 24 Up and down limits
 - 25 Up limit as superscript
 - 26 Down limit as subscript
 - 27 Up and down limits as superscript and subscript
- 0 Integral
 - 28 Integration with no limits
 - 29 Up limit
 - 30 down limit
 - 31 Up and down limits
 - 32 Up limit as superscript
 - 33 Down limit as subscript
 - 34 Up and down limits as superscript and subscript
- Product
 - 35 Product with no limits
 - 36 Up limit
 - 37 down limit
 - 38 Up and down limits
 - 39 Up limit as superscript
 - 40 Down limit as subscript
 - 41 Up and down limits as superscript and subscript
- Cup
 - 42 Cup with no limits
 - 43 Up limit
 - 44 down limit
 - 45 Up and down limits
 - 46 Up limit as superscript
 - 47 Down limit as subscript
 - 48 Up and down limits as superscript and subscript
- Cap
 - 49 Integration with no limits
 - 50 Up limit
 - 51 down limit
 - 52 Up and down limits
 - 53 Up limit as superscript
 - 54 Down limit as subscript
 - 55 Up and down limits as superscript and subscript
- Script

- 56 Script with no limits
- 57 Up limit
- 58 down limit
- 59 Up and down limits
- 60 Up limit as superscript
- 61 Down limit as subscript
- 62 Up and down limits as superscript and subscript
- 8 Labelled arrow template
 - 0 Left arrow with no limits
 - 1 Left arrow with up limit
 - 2 Left arrow with down limit
 - 3 Right arrow with no limits
 - 4 Right arrow with up limit
 - 5 Right arrow with down limit
 - 6 Left-right arrow with no limits
 - 7 Left-right arrow with up limit
 - 8 Left-right arrow with down limit
- 9 Matrix template
 - 0 Size: 1 row/2 columns
 - 1 Size: 2 rows/1 column
 - 2 Size: 2 rows/2 columns
 - 3 Size: 1 row/3 columns
 - 4 Size: 3 rows/1 column
 - 5 Size: 3 rows/3 columns
 - 6 Size: 1 row/n columns
 - 7 Size: m rows/1 column
 - 8 Size: m rows/n columns
- 10 Template attribute
 - 0 X nudge
 - 1 Y nudge
- p1 Symbols: default spacing
Matrices: Row width
- p2 Matrices: Column width
- p3 Matrices: row alignment type
- p4 Matrices: column alignment type
- p5 Matrices: equal row heights flag
- p6 Matrices: equal column widths flag

Description Inserts an equation template at the current cursor location. The actual location at which the insertion occurs is returned.

WP_EQN_RESET_NUDGE@

Restores equation to original position

Format WP_EQN_RESET_NUDGE@()

Method [this.eqn_reset_nudge@](#)

Description *Nudging* an equation allows you to make fine adjustments in the equation character's position. The WP_EQN_RESET_NUDGE@ restores the equation's characters back to their original position if you are unhappy with your adjustments.

WP_EQN_SET_ALIGN@

Sets the equation alignment type

Format WP_EQN_SET_ALIGN@(align)

Method [this.eqn_set_align@\(align\)](#)

Arguments align One of the following values

- 0 Left
- 1 Center
- 2 Right
- 3 At =
- 4 At Marker
- 5 At .

See also [WP_EQN_GET_CURRENT_ALIGN@](#)

WP_EQN_SET_DEFAULT_STYLE@

Sets equation characters to default style

Format WP_EQN_SET_DEFAULT_STYLE@()

Method [this.eqn_set_default_style@](#)

Description Sets equation characters to their default text settings. This is bound to Attributes ® Defaults in the Applixware Equations.

WP_EQN_SET_MATRIX_INFO@

Sets matrix information: rows & columns; row & column alignment, and so on

Format WP_EQN_SET_MATRIX_INFO@(format eqn_matrix_info@ matrixInfo)

Method [this.eqn_set_matrix_info@\(format eqn_matrix_info@ matrixInfo\)](#)

Arguments matrixInfo The alignment information, which is defined as a eqn_matrix_info@ format, as follows:

format eqn_matrix_info@
rows, 'Number of rows
cols, 'Number of columns
rowAlign, 'Row alignment type
colAlign, 'Column align type
equalRowHts, 'Boolean: TRUE if heights are equal
equalColWds 'Boolean: TRUE if widths are equal

See also [WP_EQN_GET_MATRIX_INFO@](#)

WP_EQN_SET_NUDGE@

Sets template relative nudge

Format WP_EQN_SET_NUDGE@(xNudge, yNudge)

Method [this.eqn_set_nudge@\(xNudge, yNudge\)](#)

Arguments xNudge The horizontal nudge factor in mils. (1000 mils = 1 inch)
yNudge The vertical nudge factor in mils.

Description Allows you set the horizontal and vertical *nudge* factor. These values allow you to move individual sections of the equations around.

WP_EQN_SET_SIZE_INFO@

Sets all five equation font sizes

Format WP_EQN_SET_SIZE_INFO@(sizeArray)

Method [this.eqn_set_size_info@\(sizeArray\)](#)

Arguments sizeArray An array of size information, as follows:
sizeArray[0] Full size

sizeArray[1] Subscript/superscript size
sizeArray[2] Sub-subscript/superscript size
sizeArray[3] Symbol size
sizeArray[4] Sub-Symbol size

See also [WP_EQN_GET_DFLT_SIZE_INFO@](#)
[WP_EQN_GET_SIZE_INFO@](#)

WP_EQN_SET_SPACING_INFO@

Sets all six spacing parameters

Format WP_EQN_SET_SPACING_INFO@(spacingInfo)

Method [this.eqn_set_spacing_info@\(spacingInfo\)](#)

Arguments spacingInfo The spacing attributes for an equation, as follows:

spacingInfo[0] Line spacing
spacingInfo[1] Matrix row spacing
spacingInfo[2] Matrix column spacing
spacingInfo[3] Superscript height
spacingInfo[4] Subscript height
spacingInfo[5] Limit height

See also [WP_EQN_GET_DFLT_SPACING_INFO@](#)
[WP_EQN_GET_SPACING_INFO@](#)

WP_EQN_SET_STYLE@

Set style information

Format WP_EQN_SET_STYLE@(styleName, format wp_text_attrs@ attrs)

Method [this.eqn_set_style@\(styleName, format wp_text_attrs@ attrs\)](#)

Arguments styleName The style whose attributes are being set.
attrs The attributes being set

Description Sets new style attributes for the equation style. If a selection exists when this macro is invoked, these attributes are applied to the selection.

WP_EQN_VIEW_*@

Determines the scale of the displayed text in Applixware Equations

Format WP_EQN_VIEW_*@ ()

Method [this.eqn_view_*@](#)

Description This macro determines the scale of the displayed text in an Applixware Equations window. This macro calls [WP_VIEW_SCALE@](#). The following reduced and enlarged view modes are available:

WP_EQN_VIEW_40@
WP_EQN_VIEW_60@
WP_EQN_VIEW_80@
WP_EQN_VIEW_100@
WP_EQN_VIEW_120@
WP_EQN_VIEW_150@
WP_EQN_VIEW_200@
WP_EQN_VIEW_300@
WP_EQN_VIEW_400@

WP_EQN_VIEW_CHAR_ACCURATE@

Sets character accurate display state and updates profile

Format WP_EQN_VIEW_CHAR_ACCURATE@()

Method [this.eqn_view_char_accurate@](#)

Description Places Applixware Words into its WYSIWYG state. See [WP_SET_VIEW_CHAR_ACCURATE@](#). It also sets a profile so that this state is remembered.

Use [WP_EQN_VIEW_LINE_ACCURATE@](#) to place Applixware Words into its line-accurate state.

WP_EQN_VIEW_LINE_ACCURATE@

Information is displayed line accurate

Format WP_EQN_VIEW_LINE_ACCURATE@()

Method [this.eqn_view_line_accurate@](#)

Description Places Applixware into to its line-accurate display state; character spacing may be slightly inaccurate. See [WP SET VIEW LINE ACCURATE@](#). It also sets a profile so that this state is remembered.

Use [WP EQN VIEW CHAR ACCURATE@](#) to place Applixware Words into its WYSIWYG state.

WP_ESCAPE_KEY@

Aborts an operation

Format WP_ESCAPE_KEY@()

[Method](#) this.escape_key@

Description Terminates an operation. If you are in a dialog box, this macro produces the same effect as pressing the Cancel button. Within a document, it places the cursor at the start of the current selection.

If you are in a header or footer, this macro places the cursor back in the main flow. If information is selected, the information will no longer be selected.

WP_ESCAPE_TEXT@

Places escape characters in text

Format outText = WP_ESCAPE_TEXT@(inText)

[Method](#) outText = this.escape_text@(inText)

Arguments inText The passed text string.

Description Places escape characters in text. Parameters in field methods or comments may require quotes. The macro places an escape (backslash) character before double-quote and backslash characters, so these characters will appear in the string.

For example, if you wanted the string

"Absolute" value \ number

in a field method, with the quotes surrounding Absolute, you would pass the string to the macro. The macro would return the following string in outText:

\"Absolute\" value \\ number

See also [WP UNESCAPE TEXT@](#)

WP_EVAL_ALL_FIELDS@

Evaluates all fields

Format WP_EVAL_ALL_FIELDS@()

Method [this.eval_all_fields@](#)

Description Evaluates all fields in the document. *Evaluating* a field can change the information that is displayed if the field's method has changed. For example, evaluating a date field will change the displayed date.

WP_EVAL_FIELDS@

Evaluates selected fields

Format WP_EVAL_FIELDS@()

Method [this.eval_fields@](#)

Description Evaluates all fields in the current selection. *Evaluating* a field can change the information that is displayed if the field's method has changed. For example, evaluating a date field will change the displayed date.

WP_EXIT@

Exits the current Words document and closes the window

Format WP_EXIT@()

Method [this.exit@](#)

Description Does not save the contents of the current Words document or display a dialog box asking if you want to save the document. Any edits made to a document since the last time it was saved to a file are lost.

WP_EXPORT_*@

Writes document in a foreign format

Format WP_EXPORT_ASCII_LAYOUT@(newName[, dontLocalizeFlag])
WP_EXPORT_ASCII_LINES@(newName[, dontLocalizeFlag])
WP_EXPORT_ASCII_PARAS@(newName[, dontLocalizeFlag])
WP_EXPORT_HTML@(newname)

WP_EXPORT_RTF@(newName[, dontLocalizeFlag])
WP_EXPORT_WP311@(newName)

Method this.export_*(newName[, dontLocalizeFlag])

Arguments newName The name to which the converted file is written.
dontLocalizeFlag
A Boolean value which if set to TRUE indicates that no links are localized. See **LOCALIZE LINKS@**.
In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an Applixware Words document into a file format that can be used by a non-Applixware product. For other word export file formats refer to the FILTER_WP_*@ macros.

WP_EXTRACT_INDEX_FIELD_ITEMS@

Breaks an index string into substrings

Format flag = WP_EXTRACT_INDEX_FIELD_ITEMS@(entry, lowChar, array, nLevels)

Method flag = this.extract_index_field_items@(entry, lowChar, array, nLevels)

Arguments entry A single string for this entry.
lowChar The separation character used in the strings.
array Equivalent array for this entry, split by levels. (This is an output field.)
nLevels The number of levels in this entry. (This is an output field.)

Description WP_EXTRACT_INDEX_FIELD_ITEMS@ extracts individual levels from sorted entry items, and outputs the result as an array. This macro also outputs the number of levels in the item.

It returns TRUE if this is a "See Also" item (indicated by a pair of separation characters before the last level). It assumes that the passed entry neither starts nor ends with the separation character, and that the paired separation character occurs only once, before the lowest level.

For example, if "!" represents the separation character, the entry:

Animals!mammals!cats!Pets

will be output in the following array, with nLevels = 4 and TRUE being returned:

0: Animals
1: mammals
2: cats

Equation Insert Macros

Format WP_EQN_INSERT_*@()

Method this.eqn_insert_*@

Description Inserts a symbol or character into the Words equation editor.

WP_EXECUTE_MACRO_FIELD@

Executes the first
hyperlink in the selected area

Format WP_EXECUTE_MACRO_FIELD@()

Description Locates and executes the first selected macro field in a words document. Only the first macro field within the selected area is executed.

WP_FIND_BEAD_CLASS@

Finds a class of beads

Format outbeadNum = WP_FIND_BEAD_CLASS@(beadNum, class, forwardFlag,
endBead, endClass)

Method outbeadNum = this.find_bead_class@(beadNum, class, forwardFlag, endBead,
endClass)

Arguments

beadNum	The sequence number of the bead to start searching at.
class	The bead class you want to find.
forwardFlag	A Boolean value that sets the search direction. TRUE searches forward and FALSE searches backwards.
endBead	The bead number that the search stops at.
endClass	The class that the search stops at.

Description This function accepts a bead sequence number, a search direction, and a desired bead class. It also accepts a terminating bead number and class. It returns the sequence number of the first bead before/after the passed bead with the desired class, or it returns 0 if such a bead is not found before the terminating conditions are met.

The search includes the passed bead. If you do not want to stop looking upon encountering any particular class, set endClass to WP_NO_CLASS.

NOTE: This function does not stop looking when it comes to the end of a flow.

The following table lists bead classes:

WP_NO_CLASS	0
WP_TEXT_CLASS	1
WP_PARAGRAPH_CLASS	2

WP_STYLE_CLASS	3
WP_BITMAP_CLASS	4
WP_OBJECT_CLASS	5
WP_COLUMN_BREAK_CLASS	6
WP_PAGE_BREAK_CLASS	7
WP_MARKER_CLASS	8
WP_ERROR_CLASS	9
WP_CELL_END_CLASS	10
WP_ROW_START_CLASS	11
WP_SECTION_CLASS	12
WP_SERIES_CLASS	13
WP_FIELD_SPLIT_CLASS	14
WP_EQUATION_CLASS	15
WP_VARIABLE_CLASS	16
WP_COLOR_CLASS	17
WP_GRAPHICS_CLASS	18
WP_TAG_CLASS	19
WP_FIELD_START_CLASS	20
WP_FIELD_END_CLASS	21
WP_STYLE_START_CLASS	22
WP_STYLE_END_CLASS	23
WP_GROUP_START_CLASS	24
WP_GROUP_END_CLASS	25
WP_FOOTNOTE_START_CLASS	26
WP_FOOTNOTE_END_CLASS	27
WP_HDRFTR_START_CLASS	28
WP_HDRFTR_END_CLASS	29
WP_GLOSSARY_START_CLASS	30
WP_GLOSSARY_END_CLASS	31
WP_FLOW_START_CLASS	32
WP_FLOW_END_CLASS	33
WP_VARIABLES_START_CLASS	34
WP_VARIABLES_END_CLASS	35
WP_DOCUMENT_START_CLASS	36
WP_DOCUMENT_END_CLASS	37

See also [WP FIND BEAD OF CELL@](#)
[WP FIND BEAD OF PARA@](#)
[WP FIND FIELD OF BEAD@](#)
[WP FIND FIRST IN PARA@](#)
[WP FIND FLOW BEAD@](#)
[WP FIND STYLE BEAD@](#)
[WP FIND VISIBLE BEAD@](#)

WP_FIND_BEAD_OF_CELL@

Returns a cell's Cell End bead

Format beadBum = WP_FIND_BEAD_OF_CELL@(bead)

Method beadBum = this.find_bead_of_cell@(bead)

Arguments bead The passed bead.

Description Returns the Cell End bead number of the cell in which bead resides. It returns 0 if the passed bead is not in a cell.

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_PARA@](#)
[WP_FIND_FIELD_OF_BEAD@](#)
[WP_FIND_FIRST_IN_PARA@](#)
[WP_FIND_FLOW_BEAD@](#)
[WP_FIND_STYLE_BEAD@](#)
[WP_FIND_VISIBLE_BEAD@](#)

WP_FIND_BEAD_OF_PARA@

Finds a paragraph's Paragraph bead

Format outbeadNum = WP_FIND_BEAD_OF_PARA@(beadNum)

Method outbeadNum = this.find_bead_of_para@(beadNum)

Arguments beadNum The sequence number of the bead at which searching will begin.

Description Returns the Paragraph bead of the paragraph which contains the passed bead. Note that the returned bead corresponds to the paragraph marker symbol that represents the *end* of a paragraph. If a bead is not contained within a paragraph (e.g., Cell End or Row Start beads), 0 is returned.

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_CELL@](#)
[WP_FIND_FIELD_OF_BEAD@](#)
[WP_FIND_FIRST_IN_PARA@](#)
[WP_FIND_FLOW_BEAD@](#)
[WP_FIND_STYLE_BEAD@](#)

WP_FIND_VISIBLE_BEAD@

WP_FIND_CELL_ID@

Finds cell with passed cell ID

Format beadNum = WP_FIND_CELL_ID@(ID, forwardFlag, startBead, stopBead)

Method beadNum = this.find_cell_id@(ID, forwardFlag, startBead, stopBead)

Arguments

ID	The string representing the cell ID.
forwardFlag	A Boolean value, where TRUE means search forward, FALSE means search back.
startBead	The starting bead number for the search.
stopBead	The ending bead number for the search.

Description In a form, the macro searches for the first cell ID matching the passed string. If a matching cell ID is found, the macro returns the bead number of the cell, otherwise it returns 0. If startBead is a Cell End bead, the cell ID of startBead is included in the search.

By default, WP_FIND_CELL_ID@ searches from the startBead to the end of the document. If your document is very long, you should specify a stopBead to limit the length of your search.

If forwardFlag is TRUE, the macro searches from startBead to the end of the flow. If startBead is 0, the search starts at the beginning of the flow.

If forwardFlag is FALSE, the macro searches from startBead to the beginning of the flow. If startBead is 0, the search starts at the end of the flow.

See also [WP_FIND_CELL_ID_IN_ROW@](#)
[WP_GET_SELECTED_FORM_ATTRS@](#)

WP_FIND_CELL_ID_IN_ROW@

Finds cell with passed ID in specific row

Format beadNum = WP_FIND_CELL_ID_IN_ROW@(ID, rowBead)

Method beadNum = this.find_cell_id_in_row@(ID, rowBead)

Arguments

ID	The string representing the cell ID.
rowBead	A bead number in the search row.

Description In a form, the macro searches for the first cell ID matching the passed string in the row. If a matching cell ID is found, the macro returns the bead number of the cell, otherwise it returns 0.

See also [WP_FIND_CELL_ID@](#)
[WP_GET_SELECTED_FORM_ATTRS@](#)

WP_FIND_FIELD_OF_BEAD@

Finds the Field Start bead of a field

Format outbeadNum = WP_FIND_FIELD_OF_BEAD@(beadNum)

Method outbeadNum = this.find_field_of_bead@(beadNum)

Arguments beadNum The sequence number of the bead at which searching begins.

Description Returns the Field Start bead of the field in which the bead falls, or 0 if the bead does not fall within a field. A Field Start or Field End bead is *not* within the field that they define. If beadNum corresponds to a Field Start or a Field End bead and the field is nested within a larger field, the Field Start bead of that larger bead is returned; otherwise, 0 is returned.

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_CELL@](#)
[WP_FIND_BEAD_OF_PARA@](#)
[WP_FIND_FIRST_IN_PARA@](#)
[WP_FIND_FLOW_BEAD@](#)
[WP_FIND_STYLE_BEAD@](#)
[WP_FIND_VISIBLE_BEAD@](#)

WP_FIND_FIRST_IN_PARA@

Finds the first bead in a paragraph

Format outbeadNum = WP_FIND_FIRST_IN_PARA@(beadNum)

Method outbeadNum = this.find_first_in_para@(beadNum)

Arguments beadNum The sequence number of a paragraph bead within a paragraph.

Description Returns the first bead of the paragraph which contains the passed bead. Note that the paragraph marker symbol represents the *end* of a paragraph. An error occurs if a non-paragraph bead is passed.

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_CELL@](#)
[WP_FIND_BEAD_OF_PARA@](#)
[WP_FIND_FIELD_OF_BEAD@](#)
[WP_FIND_FLOW_BEAD@](#)
[WP_FIND_STYLE_BEAD@](#)
[WP_FIND_VISIBLE_BEAD@](#)

WP_FIND_FLOW_BEAD@

Finds a bead with a specific class

Format outbeadNum = WP_FIND_FLOW_BEAD@(beadNum, class, forwardFlag, endBead, endClass)

Method outbeadNum = this.find_flow_bead@(beadNum, class, forwardFlag, endBead, endClass)

Arguments

beadNum	The sequence number of the bead to start searching at.
class	The bead class you want to find.
forwardFlag	A Boolean value that sets the search direction. TRUE searches forward and FALSE searches backwards.
endBead	The bead number that the search stops at.
endClass	The class that the search stops at.

Description This function accepts a bead sequence number, a search direction, and a desired bead class. It also accepts a terminating bead number and class. It returns the sequence number of the first bead within the current flow before/after the passed bead with the desired class, or 0 if such a bead is not found before the terminating conditions are met.

The search includes the passed bead. If you do not want to stop searching upon encountering any particular class, set endClass to WP_NO_CLASS.

The following table lists bead classes:

WP_NO_CLASS	0
WP_TEXT_CLASS	1
WP_PARAGRAPH_CLASS	2
WP_STYLE_CLASS	3
WP_BITMAP_CLASS	4
WP_OBJECT_CLASS	5
WP_COLUMN_BREAK_CLASS	6
WP_PAGE_BREAK_CLASS	7
WP_MARKER_CLASS	8

WP_ERROR_CLASS	9
WP_CELL_END_CLASS	10
WP_ROW_START_CLASS	11
WP_SECTION_CLASS	12
WP_SERIES_CLASS	13
WP_FIELD_SPLIT_CLASS	14
WP_EQUATION_CLASS	15
WP_VARIABLE_CLASS	16
WP_COLOR_CLASS	17
WP_GRAPHICS_CLASS	18
WP_TAG_CLASS	19
WP_FIELD_START_CLASS	20
WP_FIELD_END_CLASS	21
WP_STYLE_START_CLASS	22
WP_STYLE_END_CLASS	23
WP_GROUP_START_CLASS	24
WP_GROUP_END_CLASS	25
WP_FOOTNOTE_START_CLASS	26
WP_FOOTNOTE_END_CLASS	27
WP_HDRFTR_START_CLASS	28
WP_HDRFTR_END_CLASS	29
WP_GLOSSARY_START_CLASS	30
WP_GLOSSARY_END_CLASS	31
WP_FLOW_START_CLASS	32
WP_FLOW_END_CLASS	33
WP_VARIABLES_START_CLASS	34
WP_VARIABLES_END_CLASS	35
WP_DOCUMENT_START_CLASS	36
WP_DOCUMENT_END_CLASS	37

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_CELL@](#)
[WP_FIND_BEAD_OF_PARA@](#)
[WP_FIND_FIELD_OF_BEAD@](#)
[WP_FIND_FIRST_IN_PARA@](#)
[WP_FIND_STYLE_BEAD@](#)
[WP_FIND_VISIBLE_BEAD@](#)

WP_FIND_FORM_ID@

Finds an entry area with the passed ID

Format bead = WP_FIND_FORM_ID@(ID, forwardFlag, startBead)

Arguments ID The string representing the cell ID.

forwardFlag A Boolean value, where TRUE means search forward, FALSE means search back.
startBead The starting bead number for the search.

Description In a form, the macro searches for the first form location matching the passed ID. If a matching ID is found, the macro returns the bead number of that location, otherwise it returns 0.

This macro works on both cells and entry fields. If the entry area is a cell, the Cell End bead number is returned. If the entry area is an Entry field, the Field Start bead number is returned.

If forwardFlag is TRUE, the macro searches from startBead to the end of the flow. If startBead is 0, the search starts at the beginning of the flow.

If forwardFlag is FALSE, the macro searches from startBead to the beginning of the flow. If startBead is 0, the search starts at the end of the flow.

WP_FIND_FORM_ID_BETWEEN_BEADS@

Finds an entry area
with the passed ID between two beads

Format bead = WP_FIND_FORM_ID_BETWEEN_BEADS@(firstbead, lastbead, ID)

Arguments firstbead A Boolean value, where TRUE means search forward, FALSE means search back.
lastbead The starting bead number for the search.
ID The string representing the cell ID.

Description In a form, the macro searches between the firstbead and the last bead for the first form location matching the passed ID. If a matching ID is found, the macro returns the bead number of that location. Otherwise it returns 0.

This macro works on both cells and entry fields. If the entry area is a cell, the Cell End bead number is returned. If the entry area is an Entry field, the Field Start bead number is returned.

Zero is returned if no entry area with the proper ID is found. Searching will occur backwards if first is larger than last.

WP_FIND_HYPER_TARGET@

Locates a hypertarget

Format flag = WP_FIND_HYPER_TARGET@(string)

Method flag = this.find_hyper_target@(string)

Arguments string The hypertarget method; that is, this is the name of the hypertarget.

Description Sets the cursor at the first hypertarget in the current document whose method is string. This search always begins at the beginning of the document. Headers, footers, and footnotes are also searched. However, targets within glossaries are ignored.

This macro returns TRUE if the hypertarget string is found.

The Applixware Help system uses this macro indirectly as this macro is not capable of managing targets in multiple files.

See also [WP_FIND_NEXT_HYPER_TARGET@](#)
[WP_FIND_PREV_HYPER_TARGET@](#)

WP_FIND_LAST_PARA@

Returns the last paragraph's Paragraph bead

Format beadNum = WP_FIND_LAST_PARA@()

Method beadNum = this.find_last_para@

Description Returns the Paragraph bead number of the last paragraph in the main document flow. This paragraph always exists in a Words document. While you may place information after this bead from within ELF, the results are unpredictable (and often catastrophic).

See also [WP_FIND_NEXT_PARA@](#)
[WP_FIND_PREV_PARA@](#)

WP_FIND_MATCHING_FIELD_END@

Returns the Field End bead matching a Field Start bead

Format beadNum = WP_FIND_MATCHING_FIELD_END@ (fieldStart)

Method beadNum = this.find_matching_field_end@ (fieldStart)

Arguments fieldStart The sequence number of the Field Start bead you are trying to match.

Description Returns the Field End bead that closes the field begun with a particular Field Start bead. An error occurs if fieldStart is not a Field Start bead.

See also [WP_FIND_MATCHING_FIELD_SPLIT@](#)
[WP_FIND_MATCHING_FIELD_START@](#)

WP_FIND_MATCHING_FIELD_SPLIT@

Returns the Field Split bead associated with a Field Start bead

Format beadNum = WP_FIND_MATCHING_FIELD_SPLIT@ (fieldStart)

Method beadNum = this.find_matching_field_split@ (fieldStart)

Arguments fieldStart The sequence number of the Field Start bead you are trying to match.

Description Returns the Field Split bead that separates a field's method from its value. An error occurs if fieldStart is not a Field Start bead.

See also [WP_FIND_MATCHING_FIELD_END@](#)
[WP_FIND_MATCHING_FIELD_START@](#)

WP_FIND_MATCHING_FIELD_START@

Returns the Field Start bead matching a Field End bead

Format beadNum = WP_FIND_MATCHING_FIELD_START@ (fieldEnd)

Method beadNum = this.find_matching_field_start@ (fieldEnd)

Arguments fieldEnd The sequence number of the Field End bead you are trying to match.

Description Returns the Field Start bead that starts the field ended with a particular Field Ends bead. An error occurs if fieldEnd is not a Field End bead.

See also [WP_FIND_MATCHING_FIELD_SPLIT@](#)
[WP_FIND_MATCHING_FIELD_SPLIT@](#)

WP_FIND_NEXT@

Searches forward for next occurrence

Format WP_FIND_NEXT@()

Method this.find_next@

Description Repeats the last search you made, searching forward from the current cursor position. This search is based on three stored system variables:

- wp_last_srch_str_\$
- wp_last_srch_whole_\$
- wp_last_srch_match_\$

This macro selects the located material it finds.

See also [WP_FIND_PREV@](#)

WP_FIND_NEXT_HYPER_TARGET@

Searches for the next occurrence of a hypertarget

Format outbeadNum = WP_FIND_NEXT_HYPER_TARGET@(beadNum, string)

Method [outbeadNum = this.find_next_hyper_target@\(beadNum, string\)](#)

Arguments beadNum The sequence number of the bead at which searching begins.
 string The text of the hypertarget.

Description Searches for a hypertarget beginning at position beadNum within the current flow. If the target is found, this macro returns the hypertarget's bead position. Otherwise, it returns 0.

See also [WP_FIND_HYPER_TARGET@](#)
[WP_FIND_PREV_HYPER_TARGET@](#)

WP_FIND_NEXT_PARA@

Finds the next paragraph's Paragraph bead

Format outbeadNum = WP_FIND_NEXT_PARA@(beadNum, limit[, paraInLimit])

Method [outbeadNum = this.find_next_para@\(beadNum, limit\[, paraInLimit\]\)](#)

Arguments beadNum The sequence number of the bead at which searching begins.
 limit Indicates the place where searching stops. 0 indicates that the search stop at the end of the flow.
 paraInLimit If TRUE, the paragraph bead must be within the limit. If this value is FALSE, the paragraph must contain a bead within the limit. If this value is zero, it is ignored.

Description Returns the Paragraph bead number of the paragraph that contains or follows the passed bead. It will not look beyond the bead limit. If the paragraph bead is not found, this macro returns the value 0.

In most cases, you will want to use [WP_GOTO_NEXT_PARA@](#) instead of this macro.

See also [WP_FIND_LAST_PARA@](#)
[WP_FIND_PREV_PARA@](#)

WP_FIND_NTH_CELL@

Returns the range of the *n*th cell in the row

Format WP_FIND_NTH_CELL@(bead, colNum, format [wp_range@](#) range)

Method [this.find_nth_cell@](#)(bead, colNum, format wp_range@ range)

Arguments

bead	A bead in a table's row.
colNum	The column number.
range	The range of the found cell; if fewer than <i>N</i> cells in the current row, the range of the last cell of the row is returned.

Description Writes the range of the *n*th cell in the row containing the passed bead into range. That is, use this macro to find a particular column within a row. Note column numbers are zero-based. An error occurs if the passed bead is not in a row.

WP_FIND_PARA_OF_STYLE@

Finds the first paragraph of a given style

Format beadNum = WP_FIND_PARA_OF_STYLE@(first, last, style)

Method beadNum = [this.find_para_of_style@](#)(first, last, style)

Arguments

first	The bead number at which searching begins.
last	The last bead number of the search range.
style	The name of the style you are searching for. (This is an output argument.)

Description Finds the first paragraph (bead) of a style following first. The paragraph's bead number is returned. If a paragraph of the named style is not found, this macro returns 0.

WP_FIND_PATTERN@

Finds the next occurrence of a text pattern

Format flag = WP_FIND_PATTERN@(pattern, forwardFlag, format [wp_para_attrs@](#) paraAtts, format [wp_text_attrs@](#) textAtts)

Method flag = this.find_pattern@(pattern, forwardFlag, format wp_para_attrs@ paraAtts, format wp_text_attrs@ textAtts)

Arguments

pattern	A pattern to be found. See WP_FORWARD_SEARCH@ for a discussion of these patterns.
forwardFlag	A Boolean value that sets the search direction. TRUE is forward.
paraAtts	A format containing the set of paragraph properties that need to be in the found string.
textAtts	A format containing the set of text properties that need to be applied to the found string.

Description Finds and selects material in the document. This text can be a set of characters or it can contain pattern strings that alter the way in which the search works. (For an explanation of these patterns, see [Search Patterns](#).)
A Boolean value is returned indicating if the search was successful.

WP_FIND_PREV@

Searches backwards for previous occurrence

Format WP_FIND_PREV@()

Method this.find_prev@

Description Repeats the last search you made, searching from the current cursor position back to the beginning of the document. This search is based on three stored system variables:

- wp_last_srch_str_\$
- wp_last_srch_whole_\$
- wp_last_srch_match_\$

This macro selects the located material it finds.

See also [WP_FIND_NEXT@](#)

WP_FIND_PREV_HYPER_TARGET@

Searches for the previous occurrence of a hypertarget

Format outbeadNum = WP_FIND_PREV_HYPER_TARGET@ (beadNum, string)

Method outbeadNum = this.find_prev_hyper_target@ (beadNum, string)

Arguments beadNum The sequence number of the bead at which searching begins.
string The text of the hypertarget.

Description Searches for a hypertarget beginning at position beadNum within the current flow. If the target is found, this macro returns the hypertarget's bead number. Otherwise, it returns 0.

See also [WP_FIND_HYPER_TARGET@](#)
[WP_FIND_NEXT_HYPER_TARGET@](#)

WP_FIND_PREV_PARA@

Find the paragraph bead preceding a bead

Format foundBeadNum = WP_FIND_PREV_PARA@(beadNum, limit)

Method foundBeadNum = this.find_prev_para@(beadNum, limit)

Arguments beadNum The bead number from which the search will begin.
limit Where to stop looking. If you enter 0 (zero), you will be looking from your current point back to the beginning of the flow.

Description Locates the paragraph bead number of the paragraph that contains beadNum. If beadNum is not within a paragraph, then this macro returns the bead number of the preceding paragraph.

In most cases, you will want to use [WP_GOTO_PREV_PARA@](#) instead of this macro.

A zero is returned if the search is not successful.

See also [WP_FIND_LAST_PARA@](#)
[WP_FIND_NEXT_PARA@](#)

WP_FIND_STYLE_BEAD@

Returns the bead number of the style

Format beadNum = WP_FIND_STYLE_BEAD@(name)

Method beadNum = this.find_style_bead@(name)

Arguments name The name of a defined style.

Description Returns the bead number of the style's definition. If name does not exist, 0 (zero) is returned.

In most cases, the returned bead number is less than the bead number at which the main flow begins. That is, this bead is defined and used by Applixware Words; however, it is not visible.

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_CELL@](#)
[WP_FIND_BEAD_OF_PARA@](#)
[WP_FIND_FIELD_OF_BEAD@](#)
[WP_FIND_FIRST_IN_PARA@](#)
[WP_FIND_FLOW_BEAD@](#)
[WP_FIND_VISIBLE_BEAD@](#)

WP_FIND_VISIBLE_BEAD@

Returns the sequence number of the first visible bead of a specified type

Format outbeadNum = WP_FIND_VISIBLE_BEAD@(beadNum, class, forwardFlag, endBead, endClass)

Method outbeadNum = this.find_visible_bead@(beadNum, class, forwardFlag, endBead, endClass)

Arguments beadNum The sequence number of the bead at which searching begins.
class The bead class you want to find.
forwardFlag A Boolean value that sets the search direction. TRUE searches forward and FALSE searches backwards.
endBead The bead number that the search stops at.
endClass The class that the search stops at.

Description Returns the sequence number of the first visible bead within the current flow before/after the passed bead with the desired class, or 0 if such a bead is not found before the terminating conditions are met.

Note that the search includes the passed bead. This function accepts a bead sequence number, a search direction, and a desired bead class. WP_FIND_VISIBLE_BEAD@ also accepts a terminating bead number and class.

If fields are being displayed, the returned bead number may not be "visible" in all cases.

See also [WP_FIND_BEAD_CLASS@](#)
[WP_FIND_BEAD_OF_CELL@](#)
[WP_FIND_BEAD_OF_PARA@](#)
[WP_FIND_FIELD_OF_BEAD@](#)
[WP_FIND_FIRST_IN_PARA@](#)
[WP_FIND_FLOW_BEAD@](#)
[WP_FIND_STYLE_BEAD@](#)

WP_FORCE_CURSOR@

Insures that a cursor exists

Format format wp_location@ loc = WP_FORCE_CURSOR@(atEndFlag)

Method format wp_location@ loc = this.force_cursor@(atEndFlag)

Arguments atEndFlag A Boolean value which, if set to TRUE, places the cursor to appear at the end of the current selection. FALSE places the cursor at the beginning.

Description Either returns the location of the current cursor position or, if a range is selected, moves the cursor to either the beginning or ending location of the range and returns the resulting cursor.

The definition of wp_location@ is as follows:

```
format wp_location@  
    bead,  
    offset
```

WP_FORMAT_COLUMNS@

Formats selection into columns

Format WP_FORMAT_COLUMNS@(numCols, gutterWidth, applyType)

Method `this.format_columns@(numCols, gutterWidth, applyType)`

Arguments

<code>numCols</code>	The number of columns you want. A NULL means that this argument is ignored.
<code>gutterWidth</code>	The width between each column. A NULL means that this argument is ignored.
<code>applyType</code>	The specified area you want in columns. The values you can use are as follows: WP#APPLY#WHOLE#DOCUMENT WP#APPLY#THIS#POINT#FORWARD WP#APPLY#CURRENT#SECTION WP#APPLY#SELECTED#SECTIONS WP#APPLY#SELECTED#TEXT

Description Reformats selected material into the number of columns indicated by `numCols`. If necessary, it creates a new section for the material. This macro can reformat one of the following:

- The entire document
- From the current position to the end of the document
- The current section only
- The selected sections
- The selected material

The material being formatted can already be in a multi-column section.

WP_FORMAT_FRAME@

Reformats a frame

Format `WP_FORMAT_FRAME@(format wp_frame@ frame)`

Method `this.format_frame@(format wp_frame@ frame)`

Arguments

<code>frame</code>	A <code>wp_frame@</code> structure containing the attributes to apply to a frame. If an attribute is NULL, the attribute is removed from the frame.
--------------------	---

Description Sets (or resets) a frame's attributes to those described in the `wp_frame@` structure. Before executing this macro, the frame must be selected. If more than one frame is selected, this macro can change attributes for all selected frames.

WP_FORMAT_GRAPHIC@

Reformats selected graphic

Format WP_FORMAT_GRAPHIC@(format wp_graphic_attrs@ attrs)

Method [this.format_graphic@](#)(format wp_graphic_attrs@ attrs)

Arguments attrs The new graphic attributes.

Description Sets new graphic attributes to the selected graphic. The definition of wp_graphic_attrs@ is as follows:

format wp_graphic_attrs@
clipMode,
proportionalScaling,
xScale,
yScale,
useSrcXOffset,
useSrcYOffset,
useSrcWidth,
useSrcHeight,
xOffset,
yOffset,
srcWidth,
srcHeight

WP_FORMAT_NUMBER@

Generates a member of a series

Format fmtValue = WP_FORMAT_NUMBER@(series, value)

Method [fmtValue = this.format_number@](#)(series, value)

Arguments series An integer value, as follows:

-1	WP_NUMBERING_NONE Unrecognized series
0	WP_NUMBERING_1 1, 2, 3, 4, ...
1	WP_NUMBERING_SMALL_A a, b, c, d, ...
2	WP_NUMBERING_CAP_A A, B, C, D, ...
3	WP_NUMBERING_SMALL_I i, ii, iii, iv, ...

4 WP_NUMBERING_CAP_I
I, II, III, IV, ...

value The value you want to format.

Description Returns the string representing the passed value in the appropriate format.

NOTE: The first value in a list is assumed to be 1, not 0.

For example, WP_FORMAT_NUMBER@
(WP_NUMBERING_CAP_A, 3) returns C.

WP_FORMAT_SECTION_TYPE@

Sets the section break type

Format WP_FORMAT_SECTION_TYPE@(type)

Method [this.format_section_type@](#)(type)

Arguments type One of the following types of sections:

- 0 Continuous
- 1 Next Column
- 2 Next Page
- 3 Odd Page
- 4 Even Page

Description Sets the relationship of the previous section with the current section. That is, this sets the kind of section break between two sections.

WP_FORWARD_SEARCH@

Searches forward for a pattern

Format flag = WP_FORWARD_SEARCH@(string, wholewordsFlag, matchCaseflag,
format wp_para_attrs@ paraAtts, format wp_text_attrs@ textAtts)

Method [flag = this.forward_search@](#)(string, wholewordsFlag, matchCaseflag, format
wp_para_attrs@ paraAtts, format wp_text_attrs@ textAtts)

Arguments string The search string.

wholewordsflag

A Boolean, where TRUE means that the search only succeeds if the letters of the word are not contained within another word.

matchCaseflag

A Boolean, where TRUE means that the search only succeeds if the case of the letters is the same as the case of the letters in string.

paraAtts	A variable of type <code>wp_para_attrs@</code> that indicates paragraph attributes of the search string.
textAtts	A variable of type <code>wp_text_attrs@</code> that indicates character attributes of the search string. For example, you could request that the search string be in bold.

Description Searches from the current cursor position forward to the end of the file looking for a string. The found material is selected.

This macro returns a flag indicating if the search was successful.

For a discussion of pattern matching, see [Search Patterns](#).

See also [WP BACKWARD SEARCH@](#)

Search Patterns

You can modify this search as follows:

- You could allow the string to appear anywhere within text or you could require that your search succeed only if your string is not part of another string.
- You could allow the search to succeed if the case of the found string differs from the case of the search string.
- You could require that the search succeed only if the string were found in a paragraph matching the attributes you specify.
- You could require that the search succeed only if the found string has the specified character attributes.
- The search string can be a pattern.

When entering these characters in a search string, you must signal that they are special by preceding them with the backslash ("`\`") escape character. For example, "`\>`" indicates that you are looking for a word end boundary and not the "greater than" symbol.

The search string patterns are as follows:

<code>*</code>	any group of characters
<code>?</code>	any one character
<code>-</code>	soft hyphen
<code><</code>	word start
<code>></code>	word end
<code>[</code>	paragraph start
<code>]</code>	paragraph end
<code>{</code>	field start

} field end
 [...] multi-byte pattern code delimiter
 c clipboard
 C column break
 I inset
 n newline
 p paragraph marker
 P pagebreak
 S section break
 t tab

Patterns within multi-byte sequences include:

F field method
 G Glossary name
 L Latin
 N Style name
 S Symbol font
 Z Dingbat font

For example, the following pattern looks for a style named Head1Para:

`\|NHead1Para|`

WP_GET_WP_GENERATE_NEW_OBJECT_NAME@

Returns the name to be used for the next new object

Format newName = WP_GENERATE_NEW_OBJECT_NAME@()

Method newName = this.generate_new_object_name@

Description Returns the name that will be assigned to the next object inserted into the document. This name takes the form object + number. The value of number is one greater than the number of existing objects within the document.

WP_GET_ACTUAL_FRAME_SIZE@

Returns frame's height and width

Format WP_GET_ACTUAL_FRAME_SIZE@(width, height)

Method this.get_actual_frame_size@(width, height)

Arguments

width	The returned width of the frame in mils. (1000 mils = 1 inch)
height	The returned height of the frame in mils.

Description Returns the height and width of the currently selected frame.

See also [WP_GET_FRAME_ATTRS@](#)

WP_GET_ALL_DOC_VAR_NAMES@

Returns a list of a document's document variables

Format array = WP_GET_ALL_DOC_VAR_NAMES@()

Method array = this.get_all_doc_var_names@

Description Returns an array of document variable names. WP_GET_ALL_DOC_VAR_NAMES@ also returns names that end in one of the *hidden* characters that are reserved for customer ELF programmers.

See also [WP_GET_DOC_VAR@](#)
[WP_GET_DOC_VAR_NAMES@](#)

WP_GET_ALL_PARAS@

Returns all the text in the document as an array of strings

Format array = WP_GET_ALL_PARAS@()

Method array = this.get_all_paras@

Description Returns all the text in the document as an array of strings. Each string is returned as a simple paragraph text.

WP_GET_ALL_SESSION_VAR_NAMES@

Returns a list of session variables

Format array = WP_GET_ALL_SESSION_VAR_NAMES@()

Method array = this.get_all_session_var_names@

Description Returns an array of session variables. These variable names include those that end in one of the *hidden* characters that are reserved for use by customer ELF programmers.

See also [WP_GET_SESSION_VAR@](#)
[WP_GET_SESSION_VAR_NAMES@](#)

WP_GET_ALL_TAGS@

Returns tag names and values

Format array = WP_GET_ALL_TAGS@()

Method array = this.get_all_tags@

Description Returns an array containing all tags, their positions and their offsets. For example, if two tags are defined, a 2 ´ 3 array is returned as follows:

array[0,0]	tag1: name
array[0,1]	tag1: position
array[0,2]	tag1: offset
array[1,0]	tag2: name
array[1,1]	tag2: position
array[1,2]	tag2: offset

See also [WP_GET_TAG@](#)
[WP_GET_TAG_BEAD@](#)
[WP_GET_TAG_LOCATION@](#)
[WP_GET_TAG_NUM@](#)

WP_GET_ASCII_DOC@

Returns TRUE if current document will be saved as an ASCII file

Format flag = WP_GET_ASCII_DOC@()

Method flag = this.get_ascii_doc@

Description Returns TRUE if the current document will be saved as an ASCII (unformatted) file. It returns FALSE if the document will be saved in the normal Appixware format.

See also [WP_SET_ASCII_DOC@](#)

WP_GET_AVAILABLE_WIDTH@

Returns the available width for a passed location

Format width = WP_GET_AVAILABLE_WIDTH@(format [wp_location@](#) loc, raw)

Method width = this.get_available_width@(format wp_location@ loc, raw)

Arguments loc The location where you want to get the width.
raw A Boolean value, which, if set to TRUE, returns the width of a section column. FALSE means return the additional width of a blocking frame.

Description Returns the available width location loc. This is the width of the section's column. Using the raw flag, you can decide whether to take the width of blocking frames into account.

WP_GET_BEAD@

Returns a bead number

Format format wp_bead@ bead = WP_GET_BEAD@(beadNum)

Method format wp_bead@ bead = this.get_bead@(beadNum)

Arguments beadNum The sequence number of the desired bead.

Description Returns the bead corresponding to beadNum. The format of this returned bead is wp_bead@.

This function calls **WP_GET_BEADS@** with an argument of 1.

WP_GET_BEADS@

Returns a series of beads

Format format arrayof wp_bead@ beads = WP_GET_BEADS@(beadNum, nBeads)

Method format arrayof wp_bead@ beads = this.get_beads@(beadNum, nBeads)

Arguments beadNum The sequence number of the first of the desired bead.
nBeads The number of beads to be returned.

Description Returns a series of beads. This macro does not guarantee the integrity of the returned data. For example, you could be requesting part of a field.

WP_GET_BEADS_IN_RANGE@

Returns beads in a range

Format format arrayof wp_beads@ beads = WP_GET_BEADS_IN_RANGE@(format wp_range@ range)

Method format arrayof wp_beads@ beads = this.get_beads_in_range@(format wp_range@ range)

Arguments range The range whose beads will be returned.

Description Returns the beads within a range. If range.last.offset is zero, then the last bead must be removed. If the last bead is a text bead, the characters including and following the offset location are deleted. However, the bead is kept.

In a text bead, delete the characters before the offset location but keep the bead unless the operation empties the bead of all the text. For other beads, if a non-zero offset occurs for the first bead, the first bead is discarded.

WP_GET_BEAD_CLASS@

Returns a bead's bead class

Format class = WP_GET_BEAD_CLASS@(beadNum)

Method class = this.get_bead_class@(beadNum)

Arguments beadNum The sequence number of a desired bead.

Description Returns the bead class of a bead, which is an integer value. See [WP_GET_BEAD_CLASSES@](#) for a list of these values.

WP_GET_BEAD_CLASSES@

Returns classes of a series of beads

Format classArray = WP_GET_BEAD_CLASSES@ (beadNum, nBeads)

Method classArray = this.get_bead_classes@(beadNum, nBeads)

Arguments beadNum The sequence number of the first bead.

nBeads The number of beads we want classes for.

Description Returns an array of bead classes, one for each bead in nBeads.

The following table lists bead classes:

WP_NO_CLASS	0
WP_TEXT_CLASS	1
WP_PARAGRAPH_CLASS	2
WP_STYLE_CLASS	3
WP_BITMAP_CLASS	4
WP_OBJECT_CLASS	5
WP_COLUMN_BREAK_CLASS	6

WP_PAGE_BREAK_CLASS	7
WP_MARKER_CLASS	8
WP_ERROR_CLASS	9
WP_CELL_END_CLASS	10
WP_ROW_START_CLASS	11
WP_SECTION_CLASS	12
WP_SERIES_CLASS	13
WP_FIELD_SPLIT_CLASS	14
WP_EQUATION_CLASS	15
WP_VARIABLE_CLASS	16
WP_COLOR_CLASS	17
WP_GRAPHICS_CLASS	18
WP_TAG_CLASS	19
WP_FIELD_START_CLASS	20
WP_FIELD_END_CLASS	21
WP_STYLE_START_CLASS	22
WP_STYLE_END_CLASS	23
WP_GROUP_START_CLASS	24
WP_GROUP_END_CLASS	25
WP_FOOTNOTE_START_CLASS	26
WP_FOOTNOTE_END_CLASS	27
WP_HDRFTR_START_CLASS	28
WP_HDRFTR_END_CLASS	29
WP_GLOSSARY_START_CLASS	30
WP_GLOSSARY_END_CLASS	31
WP_FLOW_START_CLASS	32
WP_FLOW_END_CLASS	33
WP_VARIABLES_START_CLASS	34
WP_VARIABLES_END_CLASS	35
WP_DOCUMENT_START_CLASS	36
WP_DOCUMENT_END_CLASS	37

WP_GET_BEAD_END@

Returns the offset *beyond* the passed bead

Format offset = WP_GET_BEAD_END@(beadNum)

Method offset = this.get_bead_end@(beadNum)

Arguments beadNum The sequence number of a bead.

Description Returns the offset *beyond* the passed bead. Use this macro to obtain the offset for the end of a range that includes the bead.

WP_GET_BEAD_FLOW_BOUNDS@

Finds limits of current flow

Format WP_GET_BEAD_FLOW_BOUNDS@(bead, flowStart, flowEnd)

Method [this.get_bead_flow_bounds@](#)(bead, flowStart, flowEnd)

Arguments

bead	A bead in the flow.
flowStart	The bead at the beginning of the flow. (This is a returned value.)
flowEnd	The bead at the end of the flow. (This is a returned value.)

Description Finds and outputs the sequence numbers of the Flow Start and Flow End beads for the flow containing the passed bead. This will be the main flow, a header or footer flow, or a footnote flow. If the passed bead is not in a flow (e.g., it is a style bead), the returned flowStart and flowEnd bead numbers are 0.

See also [WP_FIND_FLOW_BEAD@](#)

WP_GET_CELL@

Retrieves a cell's beads

Format format wp_beads@ beads = WP_GET_CELL@(beadNum, materialFlag)

Method [format wp_beads@ beads = this.get_cell@](#)(beadNum, materialFlag)

Arguments

bead	The bead in the desired cell.
materialFlag	A Boolean value, which, if set to TRUE, indicates that the entire contents of the cell are retrieved. If set to FALSE, only the terminating paragraph and Cell End beads are returned.

Description Retrieves the beads of the cell containing beadNum. An error occurs if the bead is not in a cell. Optionally, the material within the cell can also be included. If the material within the cell is not included, the array will only contain the terminating paragraph and Cell End bead.

See also [WP_SET_CELL_ATTRS@](#)

WP_GET_CHAR_AT_LOC@

Retrieves a character

Format char = WP_GET_CHAR_AT_LOC@(format [wp location@](#) loc)

Method char = this.get_char_at_loc@(format wp_location@ loc)

Arguments loc The place to look for the character.

Description Returns the 8-bit code for the character at location loc. 0 is returned if location loc is not in a text bead.

WP_GET_COLOR@

Returns a color's CMYK values

Format format wp_color@ colorFormat = WP_GET_COLOR@(colorName)

Method format wp_color@ colorFormat = this.get_color@(colorName)

Arguments colorName The name of a color within Words.

Description Returns wp_color@ information for a color named within Words. The values defined in this format are in the CMYK color system. For more information, see [WP_GET_COLOR_TABLE@](#).

Here is the definition of wp_color@:

```
format wp_color@
  name,
  cyan,
  magenta,
  yellow,
  black
```

See also [WP_SET_COLOR_TABLE@](#)

WP_GET_COLOR_TABLE@

Returns an array of wp_color@

Format format wp_color@ array = WP_GET_COLOR_TABLE@()

Method format wp_color@ array = this.get_color_table@

Description Returns an array of wp_color@ information where each array element contains information about one color. The color values listed in each format are in the CMYK color system.

Here is the definition of wp_color@:

```
format wp_color@
  name,
  cyan,
```

magenta,
yellow,
black

See also [WP_SET_COLOR_TABLE@](#)

WP_GET_COLUMN_INFO@

Returns number of columns and gutter width

Format WP_GET_COLUMN_INFO@(secBeadNum, numCols, gutterWidth)

Method [this.get_column_info@](#)(secBeadNum, numCols, gutterWidth)

Arguments secBeadNum The section's section bead number.
numCols The number of columns. (This is a returned value.)
gutterWidth The width between columns in millimeters. (This is a returned value.)

Description Specifies the number of columns and the gutter width of a section in your document.

See also [WP_SET_COLUMN_INFO@](#)
[WP_GET_CURRENT_SECTION_BEAD@](#)

CURRENT_BORDER_ATTRS@

Returns the current border attributes

Format format wp_borders_attrs@ info = WP_GET_CURRENT_BORDER_ATTRS@()

Method [format wp_borders_attrs@ info = this.get_current_border_attrs@](#)

Description Returns a wp_border_attrs@ format that describes the attributes of the current border. Some of these attributes may be explicitly set; others are set due to inheritance.

Here is the definition of wp_borders_attrs@:

```
format wp_border_attrs@
  format wp_line_attrs left,
  format wp_line_attrs right,
  format wp_line_attrs top,
  format wp_line_attrs bottom,
  format wp_line_attrs horizontal,
```

```
format wp_line_attrs vertical,  
format wp_shading_attrs@ shading,  
horizontalMargin, 'in mils (1000 mils = 1 inch)  
verticalMargin, 'in mils  
dropShadow
```

See also [WP SET CURRENT BORDER ATTRS@](#)

WP_GET_CURRENT_COLUMN_INFO@

Returns the number of columns and gutter width

Format WP_GET_CURRENT_COLUMN_INFO@(numCols, gutterWidth)

Method [this.get_current_column_info@](#)(numCols, gutterWidth)

Arguments numCols The number of columns in the current section. (This is a returned value.)
gutterWidth The width of the space between columns. (This is a returned value.)

Description Returns the number of columns and the width of the gutter between columns in the current section.

See also [WP GET COLUMN INFO@](#)

[WP SET COLUMN INFO@](#)

[WP SET CURRENT COLUMN INFO@](#)

WP_GET_CURRENT_HDRFTR_INFO@

Returns a wp_hdrftr@ structure

Format format wp_hdrftr@ info = WP_GET_CURRENT_HDRFTR_INFO@()

Method [format wp_hdrftr@ info = this.get_current_hdrftr_info@](#)

Description Returns a wp_hdrftr@ structure that describes the state of the document's headers and footers. This format's definition is as follows:

```
format wp_hdrftr@  
    oddHeaderType,  
    evenHeaderType,  
    firstHeaderType,  
    lastHeaderType,  
    oddFooterType,
```

evenFooterType,
 firstFooterType,
 lastFooterType,
 oddHeaderName,
 evenHeaderName,
 firstHeaderName,
 lastHeaderName,
 oddFooterName,
 evenFooterName,
 firstFooterName,
 lastFooterName

The values that the *Type* elements can have are as follows:

Odd Headers and Footers

WP#ODD#HDRFTR#INHERIT	0	<i>/* use same hdr/fttr as prev section */</i>
WP#ODD#HDRFTR#NORMAL	1	<i>/* use local definition */</i>
WP#ODD#HDRFTR#NONE	2	<i>/* no hdr/fttr for this section */</i>

Even Headers and Footers

WP#EVEN#HDRFTR#USE#ODD	0	<i>/* use odd hdr/fttr (for even,first) */</i>
WP#EVEN#HDRFTR#INHERIT	1	<i>/* use same hdr/fttr as prev section */</i>
WP#EVEN#HDRFTR#NORMAL	2	<i>/* use local definition */</i>
WP#EVEN#HDRFTR#NONE	3	<i>/* no hdr/fttr for this section */</i>

First Headers and Footers

WP#FIRST#HDRFTR#NOT#SPECIAL	0	<i>/* use appropriate odd/even hdr/fttr */</i>
WP#FIRST#HDRFTR#INHERIT	1	<i>/* use same hdr/fttr as prev section */</i>
WP#FIRST#HDRFTR#NORMAL	2	<i>/* use local definition */</i>

WP#FIRST#HDRFTR#NONE 3
/* no hdr/fttr for this section */

Last Headers and Footers

WP#LAST#HDRFTR#NOT#SPECIAL 0
/* use appropriate odd/even hdr/fttr */

WP#LAST#HDRFTR#INHERIT 1
/* use same hdr/fttr as prev section*/

WP#LAST#HDRFTR#NORMAL 2
/* use local definition */

WP#LAST#HDRFTR#NONE 3
/* no hdr/fttr for this section */

These types indicate how the header or footer is constructed. For example, the *normal* headers and footers are created locally in the section.

Each header or footer also has a system-defined name; for example, `_AX_HDRFTR_1_`. You can use this name as an argument to the **WP_GET_HDRFTR_FROM_NAME@** macro to obtain the first bead in that header or footer's flow.

Manipulating headers and footers is a multi-step procedure, as follows:

1. Use this macro to obtain information about a header.
2. Use the name returned by this macro to obtain the first bead in the header or footer.
3. Use **WP_GET_FLOW_BOUNDS@** to obtain the Flow Start and Flow End beads or **WP_SELECT_TO_FLOW_END@**.
4. Manipulate the header or footer text in the same way you would manipulate any other text.

NOTE: There are very few circumstances when you will need to change the names in this structure.

See also **WP_SET_CURRENT_HDRFTR_INFO@**

WP_GET_CURRENT_LANGUAGE@

Returns dictionary languages

Format WP_GET_CURRENT_LANGUAGE@(docLanguage, sgLanguage)

Method [this.get_current_language@](#)(docLanguage, sgLanguage)

Arguments docLanguage

The number corresponding to the language setting for this document.

- 0 - not set
- 1 - English
- 2 - German
- 3 - French
- 4 - Spanish
- 5 - Italian
- 6 - British English
- 7 - Swedish
- 8 - Danish
- 9 - Norwegian
- 10 - Dutch
- 11 - Portuguese
- 12 - Brazilian Portuguese
- 13 - Canadian-French
- 14 - Swiss-German
- 15 - Nynorsk (Norwegian dialect)
- 16 - Finnish

sgLanguage Retained for compatibility. It is the same as doc_language.

Description Determines which language is used by the dictionary and thesaurus in a given document as set in the user's ax_prof4 axLanguage preference. WP_GET_CURRENT_LANGUAGE@ is called by Utilities ® Spelling ® Change Language.

To set the default language for this user (in the user's ax_prof4 file), see [EDIT-PREFS@](#) or [CHANGE PREFS@](#). To set the language in a document, see [WP SET LANGUAGE@](#).

WP_GET_CURRENT_PAGE_SETUP@

Returns page size information

Format format wp_page_setup@ info = WP_GET_CURRENT_PAGE_SETUP@()

Method format wp_page_setup@ info = this.get_current_page_setup@

Description Returns a wp_page_setup@ structure that contains information about the page's size. The definition of this format is as follows:

```
format wp_page_setup@
    pageOrientation,    ' portrait, landscape
    pageWidth,         ' in mils (1000 mils = 1 inch)
    pageHeight,        ' in mils
    leftMargin,        ' in mils
    rightMargin,       ' in mils
    topMargin,         ' in mils
```

bottomMargin, ' in mils
bindingMargin ' in mils

See also [WP SET CURRENT PAGE SETUP@](#)

WP_GET_CURRENT_PARA_RANGE@

Returns current paragraph's range

Format format [wp_range@](#) range = WP_CURRENT_PARA_RANGE@()

Method format wp_range@ range = this.get_current_para_range@

Description Returns the range of the current paragraph.

WP_GET_CURRENT_PLACE@

Returns the location of the current cursor

Format format [wp_location@](#) loc = WP_GET_CURRENT_PLACE@()

Method format wp_location@ loc = this.get_current_place@

Description Returns the location of the current cursor.

See also [WP CURRENT PAGE@](#)

[WP DISPLAY LOC OF CURSOR@](#)

[WP SET CURSOR@](#)

WP_GET_CURRENT_RANGE@

Returns the current selection as a range

Format format [wp_range@](#) range = WP_GET_CURRENT_RANGE@()

Method format wp_range@ range = this.get_current_range@

Description Returns the range of the current selection.

If the current selection has more than one table cell, not every bead between the start and end of the range will be considered to be selected.

WP_GET_CURRENT_SECTION_BEAD@

Returns the current section's section bead number

Format beadNum = WP_GET_CURRENT_SECTION_BEAD@()

Method beadNum = this.get_current_section_bead@

Description Returns the bead sequence number of the current section.

See also [WP_GET_BEAD@](#)

WP_GET_CURRENT_STYLE_NAME@

Returns the style name of the current paragraph

Format styleName = WP_GET_CURRENT_STYLE_NAME@()

Method styleName = this.get_current_style_name@

Description Returns the name of the current paragraph's style. If more than one paragraph is selected and the selection contains more than one paragraph type, the empty string ("") is returned.

See also [WP_GET_STYLE@](#)

WP_GET_CURRENT_WORD@

Returns the current word

Format text = WP_GET_CURRENT_WORD@ (keepTrailingSpacesFlag, prevIfParaFlag, alphaNumFlag)

Method text = this.get_current_word@ (keepTrailingSpacesFlag, prevIfParaFlag, alphaNumFlag)

Arguments

- keepTrailingSpacesFlag
A Boolean value, which, if TRUE, returns any trailing space that may occur.
- prevIfParaFlag
A Boolean value, which, if TRUE, says to examine the preceding character if the cursor is sitting before a paragraph marker.
- alphaNumFlag
A Boolean value, which, if TRUE, indicates that only letters and

number make up words. For more information, see [WP_GET_RANGE_OF_WORD@](#).

Description Returns the raw text of the current word or the current selection. If the current selection is a cursor, `WP_GET_CURRENT_WORD@` selects and returns the word at the cursor. In either event, the caller may specify whether the returned text should include or exclude trailing spaces (the selection is adjusted accordingly). The empty string is returned if the passed location does not represent a word.

See also [WP_GET_SELECTED_WORD@](#)

WP_GET_CURSOR_XY@

Returns the X and Y coordinates of the cursor in a WP window

Format array = WP_GET_CURSOR_XY@()

Description Returns a two-element array that contains the X and Y coordinates of the current cursor position within an Applixware Words window. Array[1] contains the X coordinate. Array[2] contains the Y coordinate.

WP_GET_DICTS@

Returns an array of dictionary names

Format names = WP_GET_DICTS@()

Method names = this.get_dicts@

Description Returns an array containing the names of all loaded dictionaries.

See also [WP_SET_DICTS@](#)

WP_GET_DOC_ATTRS@

Returns the document's attributes

Format WP_GET_DOC_ATTRS@(levelIndent, printIndexFlag, printTOCFlag, allowWidowOrphanFlag, hyphMethod)

Method this.get_doc_attrs@(levelIndent, printIndexFlag, printTOCFlag, allowWidowOrphanFlag, hyphMethod)

Arguments

- levelIndent** The amount of space paragraphs are indented using Promote/Demote. This is a returned value.
- printIndexFlag** A Boolean value, which if TRUE indicates that the index should be printed with the document. This is a returned value.
- printTOCFlag** A Boolean value, which if TRUE indicates that the table of contents should be printed with the document. This is a returned value.
- allowWidowOrphanFlag** A Boolean value, which if TRUE indicates that widows and orphans are allowed. This is a returned value.
- hyphMethod** The method used for hyphenation. This can be one of the following values:
 - WP#HYPH#CALC#ONLY
Rule-based only
 - WP#HYPH#DICTI#CALC
Dictionary with rule-based backup
 - WP#HYPH#NONE
Don't hyphenate
 - WP#HYPH#DICT#ONLY
Dictionary only
 - WP#HYPH#PROFILE
Use Profile setting

Description Returns information into the five arguments of this function.

See also [WP SET DOC ATTRS@](#)

WP_GET_DOC_BOUNDS@

Determines the start and end of a document

Format WP_GET_DOC_BOUNDS@(docStart, docEnd)

Method [this.get_doc_bounds@\(docStart, docEnd\)](#)

Arguments

- docStart** The location of the first bead sequence number. This is a returned value.
- docEnd** The location of the last bead sequence number. This is a returned value.

Description Finds the start and end of an entire document. Finds and outputs the sequence numbers of the first and last beads. Everything, including document flow, styles, and glossaries contents are contained within these beads.

WP_GET_DOC_DICT@

Returns the dictionary name

Format name = WP_GET_DOC_DICT@()

Method name = this.get_doc_dict@

Description Returns the name of the current spelling dictionary.

See also [WP_SET_DICTS@](#)

WP_GET_DOC_INFO@

Returns an array of information on the current Words document

Format format doc_format_info = WP_GET_DOC_INFO@()

Method format doc_format_info = this.get_doc_info@

Description Returns an array giving the following information on the current Words document. The FORMAT template for the array information is named doc_format_. The header file containing the FORMAT template is fileinf_.am. Here is the format's definition:

```
format doc_format_,
    name,          ' The full path name of the current document.
    docid,         ' Not important: The document's unique ID number.
    on_disk,       ' Indicates if the document has been saved or read using the file
                   name specified by name. Returns TRUE if the file has been
                   saved or read. Returns FALSE if it hasn't.
    save_mode,     ' The mode in which the document was saved. save_mode can
                   be one of the following:
                   '0 Not supported: binary format
                   '1 Machine-independent (portable) format
    grp_access,    ' The permission setting for the file for members of the same
                   group. grp_access can be one of the following:
                   '0 No read or write permissions for the file
                   '1 Read permission for the file.
                   '2 Read and write permission for the file
    all_access,    ' One of the following file permissions for the users:
                   '0 No read or write permissions for the file
                   '1 Read permission for the file.
                   '2 Read and write permission for the file.
    writeable,     ' Indicates whether this user has write access. TRUE indicates
                   that this user can write this file; the value FALSE indicates that
                   the user cannot write this file.
```

See also [WP_SAVE@](#)
[WP_SET_DOC_INFO@](#)

WP_GET_DOC_LINKS_INFO@

Returns a list of linked file fields

Format format arrayof wp_link_info@ nameList =
WP_GET_DOC_LINKS_INFO@(name)

Method format arrayof wp_link_info@ nameList = this.get_doc_links_info@(name)

Arguments name The name of the document.

Description Returns the names and attributes of linked file fields contained in document name. This information is formatted as an arrayof wp_link_info@. The definition of this format is as follows:

format wp_link_info@
 name, 'string:filename
 docType, 'int: doc type returned by WP_RECOGNIZE_FILE@
 appType, 'int: app type returned by WP_RECOGNIZE_FILE@
 filterMacro, 'string: import filter macro
 launchMacro, 'string:macro to launch application
 viewname 'string: viewname (for spreadsheet)

See also [WP_SET_DOC_LINKS_INFO@](#)

WP_GET_DOC_NAME_FOR_LINKS@

Returns an object's name

Format name = WP_GET_DOC_NAME_FOR_LINKS@()

Method name = this.get_doc_name_for_links@

Description Returns the internal name of an object. Use this name in programs that need to resolve relative links.

WP_GET_DOC_VAR@

Returns a document variable's value

Format value = WP_GET_DOC_VAR@(name)

Method value = this.get_doc_var@(name)

Arguments name The name of an existing document variable.

Description Returns the value previously assigned to a document variable using [WP_SET_DOC_VAR@](#). Document variables (and their values) are stored with the document. These values are retained between Applixware sessions.

WP_GET_DOC_VAR_NAMES@

Returns a list of document variables

Format nameList = WP_GET_DOC_VAR_NAMES@()

Method nameList = this.get_doc_var_names@

Description Returns a list of the document variable names stored within a Words document. WP_GET_DOC_VAR_NAMES@ will *not* return names that end in one of the "hidden" characters that are reserved.

WP_GET_DOC_WINDOW_ATTR_TYPE@

Returns the document window attribute type

Format attributeType = WP_GET_DOC_WINDOW_ATTR_TYPE@()

Method attributeType = this.get_doc_window_attr_type@

Description Returns the document window creation option attribute type, which is one of the following values:

WP#DOC#WIN#ATTR#DEFAULT	0
WP#DOC#WIN#ATTR#LAST	1
WP#DOC#WIN#ATTR#FIXED	2

For more information, see [WP_SET_DOC_WINDOW_ATTR_TYPE@](#).

WP_GET_FACING_PAGES@

Returns TRUE if document uses facing pages

Format flag = WP_GET_FACING_PAGES@()

Method flag = this.get_facing_pages@

Description Returns a Boolean value where TRUE means that the document uses facing pages. *Facing pages* means that the inside and outside margins are dependent

on the page number. FALSE means that margins of odd and even pages are always the same.

See also [WP_SET_FACING_PAGES@](#)

WP_GET_FIELD_METHOD@

Returns a method

Format method = WP_GET_FIELD_METHOD@(beadNum)

Method method = this.get_field_method@(beadNum)

Arguments beadNum The sequence number of the desired Field Start bead.

Description Returns the method for the specified Field Start beadNum. An array is returned, representing the raw text of the field method, divided into quoted or space/tab-delimited parameters. The first item in the array is the field name (for example, date). The values that follow are the field's parameters. (While these parameters were explicitly double-quoted by the user in the method, they are not double-quoted in the array.)

WP_GET_FIELD_METHOD_TEXT@

Extracts the field method associated with a *fieldStart* bead number

Format text = WP_GET_FIELD_METHOD_TEXT@(fieldStart, nChars)

Method text = this.get_field_method_text@(fieldStart, nChars)

Arguments fieldStart The bead number of the fields fieldStart bead.
nChars The maximum number of characters to be returned. Note that this parameter is required.

Description Extracts the field method of fieldStart for display in a dialog box and returns it as a string.

If the returned text is longer than nChars, the returned text string will actually be three characters less than nChars because the string "..." is appended.

WP_GET_FIELD_NESTING_LEVEL@

Returns a bead's field nesting level

Format level = WP_GET_FIELD_NESTING_LEVEL@(beadNum)

Method level = this.get_field_nesting_level@(beadNum)

Arguments beadNum The sequence number of the desired bead.

Description Returns the field nesting level for a bead. A 0 is returned for a bead not in a field, 1 for a bead in a top-level field (including the field start, split, and end beads), and 2 for a field nested within a field, and so on.

WP_GET_FIRST_ROW@

Returns the first row of a table's Row Start bead

Format rowStartBead = WP_GET_FIRST_ROW@(bead)

Method rowStartBead = this.get_first_row@(bead)

Arguments bead A bead that is located within a table.

Description Returns the Row Start bead of the first row in the table that contains bead. bead can be a bead in the first row. An error is thrown if bead is not in a row.

See also [WP_GET_LAST_ROW@](#)

WP_GET_FLAGGED_PARA_ATTRS@

Returns paragraph attributes settings

Format format wp_flagged_para_attrs@ attr = WP_GET_FLAGGED_PARA_ATTRS@()

Method format wp_flagged_para_attrs@ attr = this.get_flagged_para_attrs@

Description Returns the following wp_flagged_para_attrs@ structure:

```
format wp_flagged_para_attrs@
  format wp_para_attrs_flags@ setFlags,
    'set if attribute value is used or inherited
  format wp_para_attrs_flags@ inheritFlags,
    'set if attribute value is inherited
  format wp_para_attrs@ attrs
    'attribute values where set and not inherited

format wp_para_attrs_flags@
  pageBreak,      ' int: TRUE->local, FALSE->inherited
  keepWith,       ' int: TRUE->local, FALSE->inherited
  block,          ' int: TRUE->local, FALSE->inherited
  justify,        ' int: TRUE->local, FALSE->inherited
  inTOC,          ' int: TRUE->local, FALSE->inherited
  indentToLevel, ' int: TRUE->local, FALSE->inherited
  noSpell,        ' int: TRUE->local, FALSE->inherited
  firstIndent,    ' int: TRUE->local, FALSE->inherited
```

```

leftIndent,      ' int: TRUE->local, FALSE->inherited
rightIndent,    ' int: TRUE->local, FALSE->inherited
preParaSpacing,' int: TRUE->local, FALSE->inherited
postParaSpacing,
                ' int: TRUE->local, FALSE->inherited
lineSpacing,    ' int: TRUE->local, FALSE->inherited
level,          ' int: TRUE->local, FALSE->inherited
hyphZone,       ' int: TRUE->local, FALSE->inherited
hyphMinFrag     ' int: TRUE->local, FALSE->inherited

format wp_para_attrs@
pageBreak,      ' int: TRUE (pagebreak before para) or FALSE
keepWith,       ' int: TRUE (keep w/next para) or FALSE
block,          ' int: TRUE (keep para on single page) or FALSE
justify,        ' int: horiz alignment
                ' (WP_ATTRS_JUSTIFY_XXXX)
inTOC,          ' int: TRUE (put para in Tbl of Cont) or FALSE
indentToLevel, ' int: TRUE (indent para to level) or FALSE
noSpell,        ' int: TRUE (don't spellcheck para) or FALSE
firstIndent,    ' int: left indent of 1st line in mils (1000 mils = 1 inch)
leftIndent,     ' int: left indent of other lines in mils
rightIndent,    ' int: right indent from right edge in mils
preParaSpacing,' mils before para
postParaSpacing,
                ' mils after para
lineSpacing,    ' mils after each line
level,          ' int: para level number
hyphZone,       ' int: percent whitespace
hyphMinFrag     ' int: min chars before and/or after hyphen

```

See also [WP SET FLAGGED PARA ATTRS@](#)

WP_GET_FLAGGED_TEXT_ATTRS@

Returns text attributes

Format format wp_flagged_text_attrs@ attr = WP_GET_FLAGGED_TEXT_ATTRS@()

Method format wp_flagged_text_attrs@ attr = this.get_flagged_text_attrs@

Description Returns the following wp_flagged_text_attrs@ format:

```

format wp_flagged_text_attrs@
format wp_text_attrs_flags@ setFlags,
                            'set if attribute value is used
format wp_text_attrs_flags@ inheritFlags,
                            'set if attribute value is inherited
format wp_text_attrs@ attrs
                            'attribute values where setFlags is set

```

```

format wp_text_attrs_flags@
    bold,          ' int: TRUE->local, FALSE->inherited
    italic,        ' int: TRUE->local, FALSE->inherited
    strikethru,    ' int: TRUE->local, FALSE->inherited
    hidden,        ' int: TRUE->local, FALSE->inherited
    capitals,      ' int: TRUE->local, FALSE->inherited
    underline,     ' int: TRUE->local, FALSE->inherited
    position,      ' int: TRUE->local, FALSE->inherited
    hyphenate,     ' int: TRUE->local, FALSE->inherited
    color,         ' int: TRUE->local, FALSE->inherited
    face,          ' int: TRUE->local, FALSE->inherited
    size,          ' int: TRUE->local, FALSE->inherited
    tag            ' int: TRUE->local, FALSE->inherited

```

```

format wp_text_attrs@
    bold,          ' int: TRUE or FALSE
    italic,        ' int: TRUE or FALSE
    strikethru,    ' int: TRUE or FALSE
    hidden,        ' int: TRUE or FALSE
    capitals,      ' int: WP_ATTRS_CAPS_XXXX
    underline,     ' int: WP_ATTRS_UNDER_XXXX
    position,      ' int: WP_ATTRS_POS_XXXX
    hyphenate,     ' int: TRUE or FALSE
    color,         ' string: color name
    face,          ' string: Font face name
    size,          ' int: in points
    tag            ' string: tag name

```

WP_GET_FLOW_BOUNDS@

Finds the start and end of the text

Format WP_GET_FLOW_BOUNDS@(flowStart, flowEnd)

Method [this.get_flow_bounds@](#)(flowStart, flowEnd)

Arguments flowStart The sequence number of the first bead in the flow.
 flowEnd The sequence number of the last bead in the flow.

Description Finds and outputs the sequence numbers of the flow start and flow end beads. The "document" is contained within these beads. Other material such as styles and glossaries are located outside these beads.

WP_GET_FONT_FAMILIES@

Returns an array of font names

Format fontArray = WP_GET_FONT_FAMILIES@()

Method fontArray = this.get_font_families@

Description Returns a list of all font names that are currently usable.

WP_GET_FORM_MODE@

Indicates if Words is in form mode

Format flag = WP_GET_FORM_MODE@()

Method flag = this.get_form_mode@

Description Returns a Boolean value indicating the form mode of a document. flag is TRUE if the document is a form, else returns FALSE.

See also [WP_DOC_IS_FORM@](#)
[WP_SET_FORM_MODE@](#)
[WP_TOGGLE_FORM_MODE@](#)

WP_GET_FRAME_ATTRS@

Returns wp_frame@ frame attributes

Format format wp_frame_attrs@ attr = WP_GET_FRAME_ATTRS@()

Method format wp_frame_attrs@ attr = this.get_frame_attrs@

Description Returns a wp_frame_attrs@ format that describes the selected frame. This format's definition is as follows:

```
format wp_frame@  
    type,  
    xposType,  
    xposRelative,  
    xpos,  
    yposType,  
    yposRelative,  
    ypos,  
    widthType,
```

width,
heightType,
height,
leftMargin,
rightMargin,
topMargin,
bottomMargin

This macro returns NULL if no frame is selected.

WP_GET_GLOSSARY_INFO@

Returns TRUE if a glossary is live

Format flag = WP_GET_GLOSSARY_INFO@(name)

Method flag = this.get_glossary_info@(name)

Arguments name The name of a glossary within the document.

Description Returns a Boolean value where TRUE indicates that a live link exists between the glossary and the glossary field.

See also [WP_SET_GLOSSARY_INFO@](#)

WP_GET_GLOSSARY_REFS_IN_RANGE@

Returns an array of names of glossary reference fields

Format names = WP_GET_GLOSSARY_REFS_IN_RANGE@ (format wp_range@ range)

Method names = this.get_glossary_refs_in_range@ (format wp_range@ range)

Arguments range The range you are passing.

Description Returns an array of names of all glossary fields in range. If there are no glossary fields, a NULL value is returned.

WP_GET_GOOD_LANGUAGE@

Returns the current language

Format value = WP_GET_GOOD_LANGUAGE@()

Method value = this.get_good_language@

Description Returns a number indicating the current language:

- 0 - not set
- 1 - English
- 2 - German
- 3 - French
- 4 - Spanish
- 5 - Italian
- 6 - British English
- 7 - Swedish
- 8 - Danish
- 9 - Norwegian
- 10 - Dutch
- 11 - Portuguese
- 12 - Brazilian Portuguese
- 13 - Canadian-French
- 14 - Swiss-German
- 15 - Nynorsk (Norwegian dialect)
- 16 - Finnish

See also [WP_GET_CURRENT_LANGUAGE@](#)
[WP_SET_LANGUAGE@](#)

WP_GET_GRAPHIC_ATTRS@

Returns graphic information

Format format wp_graphic_attrs@ attr = WP_GET_GRAPHIC_ATTRS@()

Method format wp_graphic_attrs@ attr = this.get_graphic_attrs@

Description Returns a wp_graphic_attrs@ format that contains information about the current graphic. The definition of this format is:

format wp_graphic_attrs@
clipMode,
proportionalScaling,
xScale,
yScale,
useSrcXOffset,
useSrcYOffset,
useSrcWidth,
useSrcHeight,
xOffset,
yOffset,
srcWidth,
srcHeight

WP_GET_HDRFTR_FROM_NAME@

Returns a header or footer bead

Format beadNum = WP_GET_HDRFTR_FROM_NAME@ (name)

Method beadNum = this.get_hdrftr_from_name@ (name)

Arguments name The name of a header or footer. In most cases, you will be using the name that Words assigns rather than a user-defined value.

Description Returns the bead position of the first bead in the header or footer named name. name will be one of the values returned by WP_GET_CURRENT_HDRFTR_INFO@ or [WP_GET_HDRFTR_INFO@](#).

Manipulating headers and footers is usually a multi-step process. See [WP_GET_CURRENT_HDRFTR_INFO@](#) for more information.

WP_GET_HDRFTR_INFO@

Returns information about a section's headers and footers

Format format wp_hdrftr@ info = WP_GET_HDRFTR_INFO@(section)

Method format wp_hdrftr@ info = this.get_hdrftr_info@(section)

Arguments section A section bead within the document.

Description Returns information about a section's headers and footers. See [WP_GET_CURRENT_HDRFTR_INFO@](#) for more information.

See also [WP_SET_HDRFTR_INFO@](#)

WP_GET_HDRFTR_MARGINS@

Returns the header and footer margin

Format WP_GET_HDRFTR_MARGINS@(headerMargin, footerMargin)

Method this.get_hdrftr_margins@(headerMargin, footerMargin)

Arguments headerMargin

The distance from the top of the page to the header. This is a returned value.

footerMargin The distance from the bottom of the page to the footer. This is a returned value.

Description Returns the distance from the page's top or bottom edge or the page's header or footer.

See also [WP_SET_HDRFTR_MARGINS@](#)

WP_GET_HOOK@

Returns the name of the hook macro for a class

Format macroName = WP_GET_HOOK@()

Method macroName = this.get_hook@

Description Returns the name of the hook macro defined for the current document. NULL is returned if there is no hook macro set for the document.

See [WP_SET_HOOK@](#) for more information.

WP_GET_HYPER_TARGET_NAMES@

Returns an array of all hypertext names in a document

Format nameArray = WP_GET_HYPER_TARGET_NAMES@()

Method nameArray = this.get_hyper_target_names@

Description Returns an array of all hypertext names found in hypertext fields in a document. If no hypertexts exist, NULL is returned. nameArray may contain duplicate entries if duplicate hypertexts exist.

WP_GET_LANGUAGE@

Returns current language

Format num = WP_GET_LANGUAGE@()

Method num = this.get_language@

Description Determines which language is used by the dictionary and thesaurus in a given document as set in the user's ax_prof4 axLanguage preference. See the [Language Used by Dictionary and Thesaurus](#) preference option for more information. The returned language number is one of the following:

- 0 not set
- 1 English
- 2 German

- 3 French
- 4 Spanish
- 5 Italian
- 6 British English
- 7 Swedish
- 8 Danish
- 9 Norwegian
- 10 Dutch
- 11 Portuguese
- 12 Brazilian Portuguese
- 13 Canadian-French
- 14 Swiss-German
- 15 Nynorsk (Norwegian dialect)
- 16 Finnish

To set the default language for this user (in the user's ax_prof4 file), see [EDIT-PREFS@](#) or [CHANGE_PREFS@](#). To set the language in a document, see [WP_SET_LANGUAGE@](#).

WP_GET_LANG_AND_DICTS@

Retrieves the current language and the set of dictionaries to use

Format WP_GET_LANG_AND_DICTS@(language, dictionaries)

Method [this.get_lang_and_dicts@](#)(language, dictionaries)

Arguments language The returned language of the dictionaries and thesaurus. For more information, see [WP_GET_LANGUAGE@](#).
dictionaries A returned array of dictionary names.

Description Returns the current language and the set of dictionaries to use. If there is a Default User Dictionary for the current language, it is placed after the other user dictionaries. If you are using a skip dictionary, it is placed at the end of the returned dictionaries list.

WP_GET_LAST_ROW@

Returns the Row Start bead of the last row

Format rowStartBeadNum = WP_GET_LAST_ROW@(beadNum)

Method [rowStartBeadNum = this.get_last_row@](#)(beadNum)

Arguments beadNum A bead number within a table.

Description Returns the Row Start bead of the last row in the same table. bead can be any bead in the table including a bead in the last row of the table. An error is thrown if the passed bead is not in a row.

See also [WP_GET_FIRST_ROW@](#)

WP_GET_LINKS@

Returns an array of external linked file names in a Words document

Format array = WP_GET_LINKS@()

Method array = this.get_links@

Description Returns a string array in which each string element is the name of an external linked inset file in the current document. If the document does not contain any external links, NULL is returned.

WP_GET_LINKS_INFO@

Returns information on all links

Format format arrayof wp_link_info@ info = WP_GET_LINKS_INFO@()

Method format arrayof wp_link_info@ info = this.get_links_info@

Description Returns an array of wp_link_info@ format information. This array describes all of a documents links. The definition of this format is:

format wp_link_info@

name,	' string:	file name
docType,	' int:	doc type returned by WP_RECOGNIZE_FILE@
appType,	' int:	app type returned by WP_RECOGNIZE_FILE@
filterMacro,	' string:	import filter macro
launchMacro,	' string:	macro to launch application
viewname	' string:	viewname (for spreadsheet)

See also [WP_SET_LINKS_INFO@](#)

WP_GET_MARKER_BEAD@

Returns sequence number of marker bead

Format sequenceNum = WP_GET_MARKER_BEAD@(name)

Method sequenceNum = this.get_marker_bead@(name)

Arguments name The name of an existing or proposed marker.

Description Returns the sequence number of the marker bead whose name is name. The macro returns 0 if the marker bead is not found.

WP_GET_NEXT_BEAD_CHARACTER@

Returns a character at a specified offset

Format char = WP_GET_NEXT_BEAD_CHARACTER@(format wp_bead@ bead, format wp_location@ loc, skipNonLatin, badNum)

Method char = this.get_next_bead_character@(format wp_bead@ bead, format wp_location@ loc, skipNonLatin, badNum)

Arguments bead A bead position in a flow. On input, this is the bead from which we are proposing to obtain the character. On output, this is the actual bead used.

loc The location of the character within the bead. On input, this is a proposed location. On output, this location is incremented by 1. This new location is not *legalized*.

skipNonLatin If set to TRUE, non-Latin beads are skipped over. If FALSE, 0 is returned if non-Latin beads are found.

badNum Bead number of the bead upon exit. This is a returned value.

Description Given a text bead and a location (assuming the location bead corresponds to the passed bead), WP_GET_NEXT_BEAD_CHARACTER@ extracts and returns the character at the specified offset. The offset is then incremented.

If the offset is beyond the last character, the next bead is retrieved. "" (double quotes) are returned (as the character value) if ELF is unable to retrieve a Latin character. It is acceptable to pass a non-legalized location; i.e. offset beyond end of bead. If this occurs, the next bead is returned. Note that the returned location is not legalized.

WP_GET_NEXT_ROW@

Returns the Row Start bead of the next row

Format WP_GET_NEXT_ROW@(beadNum)

Method this.get_next_row@(beadNum)

Arguments beadNum A bead number within a table.

Description Returns the Row Start bead of the next row in the same table, or 0 if beadNum is in the last row. An error occurs if the passed bead is not in a row.

See also [WP_GET_PREV_ROW@](#)

WP_GET_NEXT_SELECTION@

Gets the next selection

Format format wp_range@ selectionRange = WP_GET_NEXT_SELECTION@(rowFlag, selBead, selCol, format [wp_range@](#) range)

Method format wp_range@ selectionRange = this.get_next_selection@(rowFlag, selBead, selCol, format wp_range@ range)

Arguments

rowFlag	A Boolean value which if set to TRUE indicates that the entire set of cells that falls within the selection in the current row is returned. More information is contained within the description section.
selbead	The returned column number in which the selection begins. Initially, you should set this variable to 0. Afterwards, your code should not alter this value.
selCol	The returned Row Start bead number (or, if "rows" is FALSE, the Cell End bead number) of the range being returned. Initially, you should set this variable to 0. Afterwards, your code should not alter this value.
range	The current selection

Description Returns the next selection.

When the contents of the current selection must be examined, it should be done by calling this function in a loop. Unless the number of selected columns is zero, you cannot assume that everything in the range representing the selection is selected. The assumption fails when a table column is selected. Thus, code that directly operates on the current selection should operate on a series of ranges. This function successively returns those ranges.

To use this function, initialize the selBead and selCol arguments to zero and call this function until it returns FALSE:

```
selBead = 0
selCol = 0
while(WP_GET_NEXT_SELECTION@
      (rows, selBead, selCol, rng))
```

...operate upon rng...
wend

If rows is TRUE, the returned range will represent the entire set of cells that falls within the selection in the current row. If rows is FALSE, range will represent a single selected cell (including the Cell End Bead). Either way, the same material is returned; this simply controls whether you want to get it a cell or a (partial) row at a time. You must keep rows the same throughout the entire loop.

This function also works with a single (non-table) selection. In this case, it returns TRUE on the first cycle, and FALSE thereafter. The returned contents of selBead and selCol are not meaningful in this case.

If your code adds or deletes beads, it may not be appropriate to use this macro. (WP_SET_SELECTION_MARKS@ and WP_GET_SELECTION_MARKS@ are designed for this.) You can get away with it as long as you do not destroy the selection, and make sure selBead contains the current bead number of the Row Start bead of the row you just processed.

This macro is much faster than the WP_SET/GET_SELECTION_MARKS@ method, so use it whenever possible!

WP_GET_NTH_PARA_RANGE@

Returns the range of the nth paragraph

Format format wp_range@ range = WP_GET_NTH_PARA_RANGE@(paraNum)

Method format wp_range@ range = this.get_nth_para_range@(paraNum)

Arguments paraNum The paragraph number of the paragraph being retrieved.

Description Returns the range of the nth paragraph in the document. This function uses a zero-based system; the first paragraph is numbered starting with 0.

WP_GET_NUMBER_FORMAT@

Returns a numbering series and value

Format newNum = WP_GET_NUMBER_FORMAT@(fmtvalue, series, value)

Method newNum = this.get_number_format@(fmtvalue, series, value)

Arguments fmtvalue The text for which we need a format.

series fmtvalue's series. This is a returned value.

value fmtvalue's representation in series. This is a returned value.

Description Guess the numbering series of which that string could be a member, and thus the corresponding value. WP_GET_NUMBER_FORMAT@ returns TRUE if ELF can make a guess; otherwise, it returns FALSE (and a value of zero and series of WP_NUMBERING_NONE).

series can be one of the following:

- 1 WP_NUMBERING_NONE 'Unrecognized series
- 0 WP_NUMBERING_1 '1, 2, 3, 4, ...
- 1 WP_NUMBERING_SMALL_A 'a, b, c, d, ...
- 2 WP_NUMBERING_CAP_A 'A, B, C, D, ...
- 3 WP_NUMBERING_SMALL_I 'i, ii, iii, iv, ...
- 4 WP_NUMBERING_CAP_I 'I, II, III, IV, ...

NOTE: The first value in a list is assumed to be 1, not 0. Also, WP_GET_NUMBER_FORMAT@ does not recognize Roman and Letter series that are very large.

WP_GET_NUMBER_IN_RANGE@

Returns a numeric value in the text in the range

Format value = WP_GET_NUMBER_IN_RANGE@(format [wp_range@](#) range, parensFlag, gotNumFlag)

Method value = this.get_number_in_range@(format [wp_range@](#) range, parensFlag, gotNumFlag)

Arguments

- range The range where you are looking for a numeric value.
- parensFlag If set to TRUE, parentheses are interpreted; that is (x) represents -x. If set to FALSE, parentheses are ignored.
- gotNumFlag Returns TRUE if a numeric value is found; returns FALSE if no numeric value is found.

Description Returns the numeric value of the text in the range. 0 is returned if there is no legal number there.

See also [WP_GET_NUMBER_IN_TEXT@](#).

WP_GET_NUMBER_IN_TEXT@

Returns the numeric value of text

Format value = WP_GET_NUMBER_IN_TEXT@(text, parensFlag, gotNumFlag)

Method value = this.get_number_in_text@(text, parensFlag, gotNumFlag)

Arguments text The text containing a number.
 parensFlag If set to TRUE, parentheses are interpreted; that is (x) represents -x. If set to FALSE, parentheses are ignored.
 gotNumFlag Returns TRUE if a numeric value is found, returns FALSE if no numeric value is found.

Description Returns the numeric value contained within the text. 0 is returned if there is no legal number there. Optionally, the caller can request that parenthesized text be considered a negative number. Note that (-x) is returned as -x.

See also See also [WP_GET_NUMBER_IN_RANGE@](#).

WP_GET_NUM_SELECTIONS@

Returns the number of discontinuous selections

Format num = WP_GET_NUM_SELECTIONS@(rowFlag)

Method num = this.get_num_selections@(rowFlag)

Arguments rowFlag A Boolean value, which, if set to TRUE, indicates that selections are returned in chunks. If set to FALSE, selections are returned as individual cells.

Description Returns the number of discontinuous selections. If the current selection falls completely within a table, an implied rectangular block of cells is selected; this macro indicates the number of rows in that block. In any other case, the value 1 is returned.

WP_GET_OBJECTS_INFO@

Returns information on all local objects

Format format wp_object_info@ info = WP_GET_OBJECTS_INFO@()

Method format wp_object_info@ info = this.get_objects_info@

Description Returns an array of wp_object_info@ formats. Each format describes the characteristics of one local object. The definition of this format is:

format wp_object_info@

name,n	' string:	name
docType,	' int:	doc type returned by RECOGNIZE_FILE@
appType,	' int:	app type returned by RECOGNIZE_FILE@
filterMacro,	' string:	import filter macro

launchMacro, ' string: macro to launch application
allowUnreferenced ' int: TRUE->save even if not referenced

WP_GET_PAGE_NUMBERING_FORMAT@

Returns page numbering information

Format WP_GET_PAGE_NUMBERING_FORMAT@ (pageNumStyle, restartPageNum, restartPageNumVal, separator, sectNumStyle, sectNumPolicy, restartSectNumVal)

Method [this.get_page_numbering_format@](#) (pageNumStyle, restartPageNum, restartPageNumVal, separator, sectNumStyle, sectNumPolicy, restartSectNumVal)

Arguments pageNumStyle

One of the following predefined constants:

WP#NUMBERING#NONE

-1

WP#NUMBERING#ARABIC

0 /* 1, 2, 3... */

WP#NUMBERING#LETTER#UPPER

1 /* A, B, C... */

WP#NUMBERING#LETTER#LOWER

2 /* a, b, c... */

WP#NUMBERING#ROMAN#UPPER

3 /* I, II, III... */

WP#NUMBERING#ROMAN#LOWER

4 /* i, ii, iii... */

restartPageNum

A Boolean value, which if TRUE means that page numbers are restarted in this section; FALSE means that the page numbers are incremented.

restartPageNumVal

If set, the page numbers begin with this value.

separator The separator used between a section value and a page value.

sectNumStyle

The numbering style used for section numbers. This is one of the same constants as are used for pageNumStyle.

sectNumPolicy

A constant defining the relationship of section numbers in this section with section numbers in the previous section. This can be one of the following values:

WP#SECT#NUM#SAME 0
WP#SECT#NUM#INCREM 1
WP#SECT#NUM#RESTART 2

restartSectNumVal

If set, the section number begins with this value.

Description Returns information describing a section's page numbering.

See also [WP SET PAGE NUMBERING FORMAT@](#)

WP_GET_PAGE_OF_LOCATION@

Returns a location's page number

Format num = WP_GET_PAGE_OF_LOCATION@(format wp_location@ loc)

Method num = this.get_page_of_location@(format wp_location@ loc)

Arguments loc The passed location.

Description Returns the page number for the passed location in simple numeric (1-based) format. Location *must* be within the main document flow.

See also [WP DISPLAY LOCATION@](#)

WP_GET_PARA_BULLET@

Returns a bead for the paragraph's leading bullet

Format format wp_bead@ bullet = WP_GET_PARA_BULLET@(format [wp_range@](#) range, latinBullets, spacePolicy)

Method format wp_bead@ bullet = this.get_para_bullet@(format wp_range@ range, latinBullets, spacePolicy)

Arguments

- range The range to examine.
- latinBullets A string containing the Latin characters Applixware uses as bullets, for example, *, -, +, @, # and so on.
- spacePolicy Indicates how trailing and leading whitespace is treated. This is one of the following values:
 - 0 WP_BULLET_NO_TRAILING
 - 1 WP_BULLET_SPACES
 - 2 WP_BULLET_SPACES_OR_TAB

Description Examines range (which should represent an entire paragraph) and returns a single bead representing the paragraph's leading "bullet", or NULL if no

qualifying bullet is found. If found, the bullet (and possibly some trailing whitespace) is selected. Ideally, the caller will use [WP_GET_RANGE_OF_PARA@](#) to generate the passed range for the paragraph.

This function deselects the current range; if a bullet is found, it is selected.

NOTE: Any bead that falls partially within the selection is completely examined. Do not try to restrict the examined region to partial beads. Also, although the bullet and trailing whitespace may represent multiple and/or fractional beads, this function returns a single bead representing the bullet and whitespace.

See also [WP_SET_PARA_BULLET@](#)

WP_GET_PARA_NUMBERING@

Returns a bead for the paragraph's leading number

Format WP_GET_PARA_NUMBERING@(format [wp_range@](#) range, prefix, suffix, trailingWSFlag, series, value)

Method [this.get_para_numbering@](#)(format [wp_range@](#) range, prefix, suffix, trailingWSFlag, series, value)

Arguments

range	The range to examine.
prefix	The text Applixware Words expects before the number.
suffix	The text Applixware Words expects after the number.
trailingWSFlag	If set to TRUE, Applixware Words selects the tab or spaces that follow the suffix or number.
series	One of the following numeric series: WP#NUMBERING#NONE -1 WP#NUMBERING#ARABIC 0 /* 1, 2, 3... */ WP#NUMBERING#LETTER#UPPER 1 /* A, B, C... */ WP#NUMBERING#LETTER#LOWER 2 /* a, b, c... */ WP#NUMBERING#ROMAN#UPPER 3 /* I, II, III... */ WP#NUMBERING#ROMAN#LOWER 4 /* i, ii, iii... */
value	The value of the number discovered. 0 is returned if there is no number.

Description Examines the passed range (which should represent an entire paragraph) and returns a single bead representing the paragraph's leading "number". NULL is

returned if no qualifying number is found. If a qualifying number is found, the number (and possibly some leading text and trailing text or whitespace) is selected. This function deselects the current range; if a number is found, it is selected.

Ideally, the caller will use [WP_GET_RANGE_OF_PARA@](#) to generate the passed range for the paragraph. If the caller specifies a prefix or suffix, it must exist for this function to successfully return a number. If the caller does not specify a prefix or suffix, this function accepts a single-character, non-alphanumeric prefix (followed by any number of spaces) or suffix (preceded by any number of spaces) if such an item is present.

NOTE: Any bead that falls partially within the selection is completely examined. Do not try to restrict the examined region to partial beads. Also, although the number and leading/trailing text may represent multiple and/or fractional beads, this function returns a single bead representing the number and its text.

See also [WP_SET_PARA_NUMBERING@](#)

WP_GET_PARA_SETTINGS@

Returns paragraph attributes

Format format wp_para_attrs@ attr = WP_GET_PARA_SETTINGS@()

Method format wp_para_attrs@ attr = this.get_para_settings@

Description Returns the paragraph attributes for the currently selected material as a wp_para_attrs@ format. If an element is NULL, the value is mixed.

Here is the definition of wp_para_attrs@:

```
format wp_para_attrs@
  pageBreak,      ' int: TRUE (pagebreak before para) or FALSE
  keepWith,      ' int: TRUE (keep w/next para) or FALSE
  block,         ' int: TRUE (keep para on single page) or FALSE
  justify,       ' int: horiz alignment
                 (WP_ATTRS_JUSTIFY_XXXX)
  inTOC,         ' int: TRUE (put para in Tbl of Cont) or FALSE
  indentToLevel, ' int: TRUE (indent para to level) or FALSE
  noSpell,       ' int: TRUE (don't spellcheck para) or FALSE
  firstIndent,   ' int: left indent of 1st line in mils (1000 mils = 1 inch)
  leftIndent,    ' int: left indent of other lines in mils
  rightIndent,   ' int: right indent from right edge in mils
  preParaSpacing, ' mils before para
  postParaSpacing, ' mils after para
  lineSpacing,   ' mils after each line
```

level, ' int: para level number
hyphZone, ' int: percent whitespace
hyphMinFrag ' int: min chars before and/or after hyphen

See also [WP SET PARA SETTINGS@](#)

WP_GET_PREV_ROW@

Returns the Row Start bead of the previous row

Format rowBead = WP_GET_PREV_ROW@(bead)

Method rowBead = this.get_prev_row@(bead)

Arguments bead The bead you are examining.

Description Returns the Row Start bead of the previous row in the same table, or 0 if bead is in the first row. An error occurs if the bead is not in a row.

See also [WP_GET_NEXT_ROW@](#)

WP_GET_RANGE_OF_CELL@

Finds the range of a cell

Format flag = WP_GET_RANGE_OF_CELL@(bead, format [wp_range@](#) range)

Method flag = this.get_range_of_cell@(bead, format wp_range@ range)

Arguments bead The bead you are examining.
range The returned range of the cell containing bead.

Description If bead is in a cell, outputs the range of the cell which includes bead and returns TRUE. The output range will end with a Cell End. It returns FALSE if the bead is not in a cell. If FALSE is returned, range should be ignored because it is not guaranteed to be meaningful.

WP_GET_RANGE_OF_CELLS@

Finds the range of cells

Format flag = WP_GET_RANGE_OF_CELLS@(bead, count, format [wp_range@](#) range)

Method flag = this.get_range_of_cells@(bead, count, format wp_range@ ran

Arguments bead The bead you are examining.

count The number of cells to examine.
range The returned range of the cells.

Description If bead is in a cell, outputs a range which begins at the start of the cell containing bead and including count cells, and also returns TRUE. The output range ends with a Cell End. If the row does not contain a sufficient number of cells, the output range simply ends at the last cell in the row.
It returns FALSE if bead is not in a cell. If FALSE is returned, range should be ignored because it is not guaranteed to be meaningful.

WP_GET_RANGE_OF_COLUMN@

Returns the range of a bead's column

Format flag = WP_GET_RANGE_OF_COLUMN@(bead, format [wp_range@](#) range)

Method flag = this.get_range_of_column@(bead, format wp_range@ range)

Arguments bead The bead you are examining.
range The range of cell for bead.

Description Outputs the range of bead's column and returns TRUE. FALSE is returned if bead is not in a cell.

WP_GET_RANGE_OF_GLOSSARY@

Finds a glossary's range

Format format wp_range@ range = WP_GET_RANGE_OF_GLOSSARY@(name, inclStartEndFlag)

Method format wp_range@ range = this.get_range_of_glossary@(name, inclStartEndFlag)

Arguments name The name of the glossary.
inclStartEndFlag
A Boolean value, which if set to TRUE, tells the macro to include the glossary start and end beads in a range. If set to FALSE, the range only includes the glossary material (or a cursor if the glossary is empty).

Description Returns the range of the glossary definition as a [wp_range@](#) format variable. This range occurs before the main flow of the document. If the glossary does not exist, a cursor at bead 0 is returned.

WP_GET_RANGE_OF_LINE@

Finds a line's range

Format format wp_range@ range = WP_GET_RANGE_OF_LINE@(format wp_location@ loc, noEndingsFlag)

Method format wp_range@ range = this.get_range_of_line@(format wp_location@ loc, noEndingsFlag)

Arguments loc A location in the line.
noEndingsFlag A Boolean value, which if set to TRUE says the returned range will not include a terminating line break or paragraph marker (if one exists).

Description Returns the range (as a [wp_range@](#) format variable) of the line containing loc. The returned range may not be a legal range. For example, it could contain a Field Start without a Field End. This means you must be careful when using this range.

WP_GET_RANGE_OF_NTH_COLUMN@

Returns a column's range

Format flag = WP_GET_RANGE_OF_NTH_COLUMN@(bead, colNum, format [wp_range@](#) range)

Method flag = this.get_range_of_nth_column@(bead, colNum, format wp_range@ range)

Arguments bead The bead you are examining.
colNum The column number that you want.
range The range of the row for the bead.

Description Outputs the range of the desired column (in the same table as the bead) and returns TRUE. It returns FALSE if no row has that many columns (or the passed bead is not in a cell).

Note that column numbers are zero-based.

The range of a column is defined as follows:

- Starting at the top of the table, the first row that contains at least the desired number of columns is located.
- The column range starts at the first bead of the appropriate cell of that row.

- The range ends at the Cell End bead of the appropriate cell in the last row of the table to have at least the desired number of columns.

If FALSE is returned, you should ignore range because it is not meaningful in this case.

WP_GET_RANGE_OF_PARA@

Finds the range of a paragraph

Format flag = WP_GET_RANGE_OF_PARA@(bead, pilchroFlag, format [wp_range@](#) range)

Method flag = this.get_range_of_para@(bead, pilchroFlag, format wp_range@ range)

Arguments

bead	The bead you are examining.
pilchroFlag	A Boolean value which if set to TRUE tells the macro to include the terminating paragraph bead in the range.
range	The returned range of the paragraph.

Description Outputs the range of the paragraph which includes bead and returns TRUE. It returns FALSE if bead is not part of a paragraph (Cell End bead).

If the paragraph is completely empty and the calling macro excludes the terminating paragraph bead, there will be nothing in the returned range. In this case, FALSE is returned. If FALSE is returned, the caller should ignore the output range; it is not guaranteed to be meaningful.

WP_GET_RANGE_OF_ROW@

Finds a row's range

Format flag = WP_GET_RANGE_OF_ROW@(bead, format [wp_range@](#) range)

Method flag = this.get_range_of_row@(bead, format wp_range@ range)

Arguments

bead	The bead you are examining.
range	The returned row range.

Description Outputs the range of the row that includes bead and returns TRUE. The output range will start with a Row Start bead and end with a Cell End. It returns FALSE if bead is not in a row. If FALSE is returned, the caller should ignore the output range; it is not guaranteed to be meaningful.

WP_GET_RANGE_OF_SCREEN@

Returns range of displayed text

Format format [wp_range@](#) range = WP_GET_RANGE_OF_SCREEN@()

Method format wp_range@ range = this.get_range_of_screen@

Description Returns a variable in wp_range@ format that describes the range of the displayed screen. This range is for information with the main flow.

0 is returned if there is no main flow material being displayed. This can occur if a footnote or a header is filling the screen.

WP_GET_RANGE_OF_TABLE@

Finds a table's range

Format WP_GET_RANGE_OF_TABLE@(bead, format [wp_range@](#) range)

Method this.get_range_of_table@(bead, format wp_range@ range)

Arguments bead The bead you are examining.
range The returned table range.

Description Outputs the range of the table that includes bead and returns TRUE. The output range will start with a Row Start bead and end with a Cell End. It returns FALSE if bead is not in a table. If FALSE is returned, the caller should ignore the output range; it is not guaranteed to be meaningful.

WP_GET_RANGE_OF_WORD@

Returns the current word's range

Format format [wp_range@](#) range = WP_GET_RANGE_OF_WORD@(format [wp_location@](#) loc, trailFlag, endOKFlag, alphaNumFlag)

Method format wp_range@ range = this.get_range_of_word@(format wp_location@ loc, trailFlag, endOKFlag, alphaNumFlag)

Arguments loc Where you are looking for the word.
trailFlag A Boolean value, which if set to TRUE tells the macro that the returned range should include the trailing space after the word (if it exists).

endOKFlag A Boolean value which if loc is not before a word character but immediately follows indicates that the range of the preceding word should be returned if set to TRUE. The default is FALSE.

alphaNumFlag
A Boolean value which if set to TRUE indicates that only letters and numbers in word are used to defined a word. The default is FALSE.

Description Returns the range of the word that contains location loc.

A word is defined as follows:

- It can be a set of adjacent alphanumeric characters not containing an word starting or word ending characters (for example, space, comma, period, and so on). Characters can be in adjacent text beads or text beads separated by marker beads or a soft hyphen. However, they cannot be separated by any other bead type.
- A single symbol such as #, &, and so on.

If the passed location is not a text bead, the range of the word will be a simple cursor, indicating that there is no word. Otherwise, the location indicates a character and the word containing the character is expanded according to the above rules.

WP_GET_READ_ONLY@

Determines if a document is read-only

Format flag = WP_GET_READ_ONLY@()

Method flag = this.get_read_only@

Description Returns TRUE if the document is in read-only mode and FALSE for normal mode (that is, read-write mode).

WP_GET_ROW@

Retrieves a row's beads

Format format arrayof beads@ beads = WP_GET_ROW@(bead, materialFlag)

Method format arrayof beads@ beads = this.get_row@(bead, materialFlag)

Arguments bead The bead in the desired row.

materialFlag A Boolean value which if set to TRUE tells the macro to retrieve the entire contents of the row's cells.

Description Retrieves all beads of the row that contains bead. An error occurs if the bead is not in a row.

An array containing the Row Start and Cell End beads is returned. Optionally, the material within a cell can also be included. If not, each cell will only contain its terminating Paragraph bead.

See also [WP_GET_NEXT_ROW@](#)
[WP_GET_PREV_ROW@](#)

WP_GET_SECTION_PAGE_SETUP@

Returns page setup information

Format format wp_page_setup@ info = WP_GET_SECTION_PAGE_SETUP@
(sectionBeadNum)

Method format wp_page_setup@ info = this.get_section_page_setup@
(sectionBeadNum)

Arguments sectionBeadNum The section's bead number.

Description Returns a wp_page_setup@ format. The definition of this format is as follows:

format wp_page_setup@
pageOrientation, ' portrait, landscape
pageWidth, ' in mils (1000 mils = 1 inch)
pageHeight, ' in mils
leftMargin, ' in mils
rightMargin, ' in mils
topMargin, ' in mils
bottomMargin, ' in mils
bindingMargin ' in mils

WP_GET_SELECTED_CELL_ATTRS@

Returns selected cell's attributes

Format format wp_cell_attrs@ attrs = WP_GET_SELECTED_CELL_ATTRS@()

Method format wp_cell_attrs@ attrs = this.get_selected_cell_attrs@

Description Returns a [wp_cell_attrs@](#) format that describes the cell attributes of the current selection.

See also [WP_SET_CELL_ATTRS@](#)

WP_GET_SELECTED_FORM_ATTRS@

Returns cell attributes in the current selection

Format format [wp_cell_attrs@](#) cell = WP_GET_SELECTED_FORM_ATTRS@ (locked, unlocked)

Method format wp_cell_attrs@ cell = this.get_selected_form_attrs@ (locked, unlocked)

Arguments

locked	A Boolean value, where TRUE means retrieve locked cell attributes. Default value is FALSE.
unlocked	A Boolean value, where TRUE means retrieve unlocked cell attributes. Default value is FALSE.

Description Returns cell attributes of the currently selected cells in a form document. You can retrieve information for either locked, unlocked, or all cells.

See also [WP_GET_SELECTED_FORM_INFO@](#)
[WP_SET_SELECTED_FORM_ATTRS@](#)

WP_GET_SELECTED_FORM_INFO@

Returns form-specific information for selected cells

Format infoArray = WP_GET_SELECTED_FORM_INFO@ ()

Method infoArray = this.get_selected_form_info@

Description Returns an array containing form-specific information for the selected cells. infoArray contains the following information:

infoArray[0]	The number of locked cells in selection.
infoArray[1]	The number of unlocked cells in selection.
infoArray[2]	An array of unique cell IDs in selection.
infoArray[3]	An array of unique double-click macros in selection.
infoArray[4]	An array of unique Entry Macros in selection.
infoArray[5]	An array of unique Exit Macros in selection.
infoArray[6]	An array of unique Next Cell IDs in selection.

Null or empty information is not placed in the array.

See also [WP_GET_SELECTED_FORM_ATTRS@](#)

WP_GET_SELECTED_ROW_ATTRS@

Returns selected row's attributes

Format format wp_row_attrs@ attrs = WP_GET_SELECTED_ROW_ATTRS@()

Method format wp_row_attrs@ attrs = this.get_selected_row_attrs@

Description Returns a wp_row_attrs@ formatted variable that describes the row attributes of the row in which data is selected. The definition of this format is as follows:

format wp_row_attrs@

alignment, ' int: WP#ROW#ALIGN#LEFT, #CENTER, #RIGHT

indent, ' int: indent from left margin, in mils (1000 mils = 1 inch)

topRowMargin, ' int: in mils

bottomRowMargin,

' int: in mils

heightType, ' int: WP#ROW#HEIGHT#AUTO, #EXACT, #MIN, #MAX

height ' int: row height, in mils

NULL is returned if the selection does not contain a row. If a particular attribute is not common to all rows, then "alignment" and "heightType" are set to -1, and other members are set to NULL.

See also [WP_SET_SELECTED_ROW_ATTRS@](#)

[WP_SET_ROW_ATTRS@](#)

WP_GET_SELECTED_WORD@

Returns up to 30 characters from the first selection

Format text = WP_GET_SELECTED_WORD@()

Method text = this.get_selected_word@

Description Returns the first 30 characters of the first of any number of current selections in the current Words document.

See also [WP_GET_CURRENT_WORD@](#)

WP_GET_SELECTION_MARKS@

Returns the range of the desired selection number

Format flag = WP_GET_SELECTION_MARKS@(name, num, format [wp_range@](#)
range)

Method flag = this.get_selection_marks@(name, num, format wp_range@ range)

Arguments

name	The name used in WP_SET_SELECTION_MARKS@ .
num	The desired selection number.
range	The range of the requested selection.

Description Returns the range of the desired selection number. In most cases, there is only one selection (number 0). Only when performing intra-table selections are there "multiple" selections.

TRUE is returned if the selection exists, FALSE if the selection does not exist. Thus, the caller can write a loop to call this function until FALSE is returned. This macro assumes that WP_SET_SELECTION_MARKS@ has already been called.

See also [WP_SET_SELECTION_MARKS@](#)

WP_GET_SESSION_VAR@

Returns a session variable's value

Format value = WP_GET_SESSION_VAR@(name)

Method value = this.get_session_var@(name)

Arguments

name	The name of a session variable.
------	---------------------------------

Description Returns the value of a session variable. These values are accessible during the current session. However, they are not saved as part of the file. A session variable is a variable that is local to a particular Words file. That is, more than one Words document could have a session variable named name. Each of these variables could have a different value.

For information on setting session variable values, see [WP_SET_SESSION_VAR@](#).

WP_GET_SESSION_VAR_NAMES@

Returns a list of session variable names

Format nameArray = WP_GET_SESSION_VAR_NAMES@()

Method nameArray = this.get_session_var_names@

Description Returns a list of session variable names from Words. WP_GET_SESSION_VAR_NAMES@ will *not* return names that end in one of the reserved "hidden" characters.

WP_GET_SKIP_DICT@

Returns the name of the skip dictionary

Format name = WP_GET_SKIP_DICT@()

Method name = this.get_skip_dict@

Description Returns the name of the skip dictionary used with the spelling checker. If no skip dictionary exists, NULL is returned.

See also [WP_SET_SKIP_DICT@](#)

WP_GET_STATUS@

Returns miscellaneous Words status information

Format format wp_status@ stat = WP_GET_STATUS@()

Method format wp_status@ stat = this.get_status@

Description Returns a wp_status@ variable. This format is defined as follows:

format wp_status@	
pageNum,	'page number at cursor
pageCount,	'total number of pages
verticalPosition,	'position on page (from top) of cursor
pageHeight,	'height of page in mils (1000 mils = 1 inch)
inFootNote,	'TRUE if cursor is in a footnote
inHdrFtr,	'TRUE if cursor is in a header or footer
inFrame,	'TRUE if cursor is in a frame
inTable,	'TRUE if cursor is in a table
inField,	'TRUE if cursor is in a field

fieldMethod,	'field method string, if cursor is in a field. In nested field, this is the level one field.
overtimeMode,	'TRUE if typing mode is overtime.
macroEditing,	'TRUE if this is the macro editor
equationEditing,	'TRUE if this is the equation editor
paraNum,	'Paragraph number (line number in macro editor)
paraCount,	'Total number of paragraphs in current flow (line count in macro editor)
readOnly,	'TRUE if document in read-only mode
viewOnly,	'TRUE if document in view-only mode
zoomFactor,	'Zoom percentage
modified	'TRUE if document is modified

WP_GET_STYLE@

Returns style information

Format format wp_style@ info = WP_GET_STYLE@(name)

Method format wp_style@ info = this.get_style@(name)

Arguments name The name of a style.

Description Returns a wp_style@ formatted variable describing a style's definition. This format is as follows:

```
format wp_style@
  name,                ' Latest name for style
  orig_name,           ' Original name of style
  textFlag,            ' int: TRUE->locals,
                      ' FALSE->all inherits
                      ' array: Effective attrs
format wp_text_attrs@ text_attrs,
                      ' array: local/inherit attrs
format wp_text_attrs_flags@ text_attrs_flags,
  paraFlag,           ' int: TRUE->locals,
                      ' FALSE->all inherits
                      ' array: Effective paragraph attributes
format wp_para_attrs@ para_attrs,
                      ' array: local/inherit attrs
format wp_para_attrs_flags@ para_attrs_flags,
  frameFlag,          ' int: TRUE->local,
                      ' FALSE->inherit
                      ' array: frame attributes if local
format wp_frame@ frame,
  tabFlag,            ' int: TRUE->local,
                      ' FALSE->inherit
```

format wp_tab@ tabs, ' array: add and delete list for tabs
 ' array: inherited tabs
 format wp_tab@ inheritTabs,
 borderFlag, ' int: TRUE->local,
 ' FALSE->inherit
 ' array: Effective border attribs
 format wp_border_attrs@ border,
 ' arr: local/inherit atts
 format wp_border_attrs_flags@ border_flags,
 parent, ' name of parent style
 glossary, ' prefix glossary
 nextStyle, ' next style name
 changed ' TRUE -> values changed

See also [WP_SET_STYLE@](#)

WP_GET_STYLES_FROM_FILE@

Returns an array of all the styles

Format styleArray = WP_GET_STYLES_FROM_FILE@(file)

Method styleArray = this.get_styles_from_file@(file)

Arguments file The file you want to examine for styles.

Description Returns an array containing information for all the styles in file. Each array element is formatted as wp_style@. For the definition of this format, see [WP_GET_STYLE@](#).

WP_GET_STYLE_NAMES@

Returns the names of all styles in a file

Format nameArray = WP_GET_STYLE_NAMES@()

Method nameArray = this.get_style_names@

Description Returns an array containing the names of all styles contained in the current file.

WP_GET_TAG@

Returns tag information

Format array = WP_GET_TAG@(name)

Method array = this.get_tag@(name)

Arguments name The name of the tag.

Description Returns a three-element array whose contents are as follows:

- The name of the tag.
- The bead number.
- The bead offset.

For example:

```
var info
var wp_location@ loc
info = WP_GET_TAG@"foo"
name = info[0]
loc.bead = info[1]
loc.offset = info[2]
```

WP_GET_TAG_BEAD@

Retrieves bead portion of tag definition

Format beadNum = WP_GET_TAG_BEAD@(name)

Method beadNum = this.get_tag_bead@(name)

Arguments name The name of the tag.

Description Returns the bead number component of a tag's definition. A tag is defined as the following three-element array:

- The name of the tag.
- The bead number.
- The bead offset.

Restated, this macro returns the second element in the tag's definition.

WP_GET_TAG_LOCATION@

Returns the location associated with a tag

Format format wp_location@ loc = WP_GET_TAG_LOCATION@(name)

Method format wp_location@ loc = this.get_tag_location@(name)

Arguments name The name of the tag.

Description Returns the location associated with a tag. A tag is defined as the following three-element array:

- The name of the tag.
- The bead number.
- The bead offset.

This function combines the second and third components into a [wp_location@](#) formatted variable, then returns this variable.

WP_GET_TAG_NUM@

Returns tag information

Format array = WP_GET_TAG_NUM@(num)

Method array = this.get_tag_num@(num)

Arguments num The number currently associated with a tag.

Description Returns a three-element array whose contents are as follows:

- The name of the tag.
- The bead number.
- The bead offset.

When a tag is created, it is stored in an in-memory list. The tag's number simply represents the tag's position on this list. Because the size of the list changes as tags are added and deleted, a tag's number is constantly changing. Because many Words commands also use tags, you should never assume that a tag's number has not changed simply because you have not used a tag.

WP_GET_TEXT_ATTRS@

Returns text attributes for the current selection

Format format wp_text_attrs@ attr = WP_GET_TEXT_ATTRS@()

Method format wp_text_attrs@ attr = this.get_text_attrs@

Description Returns text attributes for the currently selected material. If an element is NULL, the value is mixed. The returned information is of type wp_text_attrs@, which is defined in the ELF include file wp_.am as follows:

```
format wp_text_attrs@
  bold,      ' int: TRUE or FALSE
  italic,    ' int: TRUE or FALSE
  strikethru,' int: TRUE or FALSE
  hidden,    ' int: TRUE or FALSE
  capitals,  ' int: WP#CAPS#XXXX_
  underline, ' int: WP#UNDER#XXXX_
  position,  ' An integer indicating the point offset from normal
```

' text position. Positive numbers indicate superscript.
 ' Negative numbers indicate subscript. Zero is normal text.
 hyphenate,
 ' int: TRUE or FALSE
 color,
 ' string: color name
 face
 ' string: Font face name
 size,
 ' int: in points
 tag
 ' string: tag name

See also [WP_SET_TEXT_ATTRS@](#)

WP_GET_TEXT_OF_CELL@

Returns text content of a cell

Format text = WP_GET_TEXT_OF_CELL@(beadNum)

[Method](#) text = this.get_text_of_cell@(beadNum)

Arguments beadNum The bead number of the cell.

Description Returns the text content of a cell. Spaces are inserted for pilchros and adjacent sets of non-Latin characters. To remove extraneous spaces, use this macro with [TRIM@](#).

To retrieve a number in a cell, pass text to [WP_GET_NUMBER_IN_TEXT@](#).

See also [WP_PUT_TEXT_IN_CELL@](#)

WP_GET_TEXT_OF_CURRENT_PARA@

Returns text of all selected paragraphs

Format rawText = WP_GET_TEXT_OF_CURRENT_PARA@(latinOnlyFlag)

[Method](#) rawText = this.get_text_of_current_para@(latinOnlyFlag)

Arguments latinOnlyFlag
 A Boolean value where TRUE indicates that only characters in the Latin fonts are returned. Each adjacent set of non-Latin characters (that is, those characters in the Symbol or Dingbat fonts) is replaced with a single space character.
 If this is FALSE, all characters are returned. However, there is no way of telling what the face of the character represented.

Description Returns the text in all selected paragraph. Even if a paragraph is partially selected, all text from the paragraph is returned.

If the selection is an empty paragraph, the empty string ("") is returned.

If the selection contains more than one paragraph, all paragraphs except the last paragraph have their paragraph markers replaced with a space character.

WP_GET_TEXT_OF_RANGE@

Retrieves text in a range

Format text = WP_GET_TEXT_OF_RANGE@(format [wp_range@](#) range, latinOnlyFlag)

Method text = this.get_text_of_range@(format [wp_range@](#) range, latinOnlyFlag)

Arguments range The range you are passing.

latinOnlyFlag

A Boolean value which if set to TRUE indicates that only characters in the Latin face are returned. Each non-Latin character (that is, a character in the Symbol or Dingbat fonts) is replaced with a space.

If this value is FALSE, all characters are returned. However, you will have no way of knowing the face of the original character.

Description Returns the text (in 8-bit values) contained within range. The empty string is returned if there is no text in range.

If more than one paragraph exists within range, the paragraph marker is represented in the returned string as a space.

WP_GET_TYPING_TEXT_ATTRS@

Returns typing text attributes for the current selection

Format format wp_text_attrs@ attr = WP_GET_TYPING_TEXT_ATTRS@()

Method format wp_text_attrs@ attr = this.get_typing_text_attrs@

Description Returns typing text attributes for the current selection. These are the local attributes that text would acquire if it were typed at the current location.

Macros that insert material into the document should consider setting the local text attributes of the inserted text beads to these attributes.

By convention, a non-Latin font (that is, the Symbol and Dingbat fonts) can never be specified as typing attributes.

The definition of the returned [wp_text_attrs@](#) format is as follows:

format wp_text_attrs@

bold, 'Boolean

italic, 'Boolean

striketru,	'Boolean
hidden,	'Boolean
capitals,	'WP#CAPS#NORMAIL_ 'WP#CAPS#UPPER_ 'WP#CAPS#LOWER_
underline,	'WP#UNDER#NONE_ 'WP#UNDER#SINGLE_ 'WP#UNDER#DOUBLE_ 'WP#UNDER#WORDS_ 'WP#UNDER#DOUBLE#WORDS 'WP#UNDER#NOTABS 'WP#UNDER#DOUBLE#NOTABS
position,	'WP#POS#NORMAL#POINTS 'WP#POS#SUB#POINTS_ 'WP#POS#SUPER#POINTS_
hyphenate,	'Boolean
color,	'color name
face,	'font name
size,	'point size
tag	'tag name

WP_GET_UNDO_NESTING@

Returns the *undo* nesting level

Format value = WP_GET_UNDO_NESTING@()

Method value = this.get_undo_nesting@

Description Returns the nesting level of the undo system. This number should be zero at the start and end of a high-level operation. Use this function to find mismatched UndoStart/UndoEnd bead pairs.

WP_GET_UNDO_STATUS@

Checks the status of the Undo/Redo system

Format array = WP_GET_UNDO_STATUS@()

Method array = this.get_undo_status@

Description Checks the status of the Undo/Redo system. It returns a five-member array:

- 0 Number of undos pending (i.e. number of times you can hit the Undo key at this point).
- 1 Text of next undo (as specified in original Undo Start).
- 2 Number of redos pending.

3 Text of next redo.

4 Nesting level (number of Undo Starts minus Undo Ends).

The array should be 0 at the start and end of a high-level operation. If it is not 0, you should ignore all other information in the returned array.

WP_GET_USER_DICT@

Returns the default user dictionary

Format name = WP_GET_USER_DICT@()

Method name = this.get_user_dict@

Description Returns the name of the default user dictionary used by the spell checker and the thesaurus.

See also [WP_SET_USER_DICT@](#)

WP_GET_VIEW_ONLY@

Returns *view-only* state

Format flag = WP_GET_VIEW_ONLY@()

Method flag = this.get_view_only@

Description Returns the *view-only* flag. If this flag is TRUE, the document is in read-only mode; FALSE indicates that the document is in read-write mode.

See also [WP_SET_VIEW_ONLY@](#)

WP_GET_VIEW_SCALE@

Returns the zoom factor

Format value = WP_GET_VIEW_SCALE@()

Method value = this.get_view_scale@

Description Returns the current zoom factor (which is one of the elements in the wp_status@ state). This returned value is expressed as a percentage of normal, which is defined as being 100%.

WP_GET_VIEW_STATE@

Returns the viewing state

Format format wp_view_state@ info = WP_GET_VIEW_STATE@()

Method format wp_view_state@ info = this.get_view_state@

Description Returns a variable in wp_view_state@ format that defines the current viewing state. The definition of this format is:

```
format wp_view_state@
pilchros,      ' view paragraph markers
ruler,         ' view ruler
breaks,        ' view page/column/section/line breaks
fieldMethods, ' view field methods (rather than field codes)
allBounds,     ' view all boundaries
cellBounds,    ' view cell boundaries
frameBounds,   ' view frame boundaries
pageBounds,    ' view page boundaries
zoomFactor,    ' percentage zoom (100 -> 100%, 200 -> 2X)
wysiwyg       ' 0-line positioning, 1-char positioning
```

See also [WP_SET_VIEW_STATE@](#)

WP_GET_WINDOW_HEIGHT@

Returns the height of the Words window

Format height = WP_GET_WINDOW_HEIGHT@()

Method height = this.get_window_height@

Description Returns the height of the Words window in mils. (1000 mils = 1 inch)

WP_GET_WINDOW_WIDTH@

Returns the width of the Words window

Format width = WP_GET_WINDOW_WIDTH@()

Method width = this.get_window_width@

Description Returns the width of the Words window in mils. (1000 mils = 1 inch)

WP_GET_XREF_SOURCE_NAMES@

Returns an array of all cross-reference source names

Format nameArray = WP_GET_XREF_SOURCE_NAMES@()

Method nameArray = this.get_xref_source_names@

Description Returns an array of all cross-reference source names found in cross-reference fields in a document. If no cross-references exist, NULL is returned. nameArray may contain duplicate entries if duplicate cross-reference sources exist.

WP_GOTO_BEGINNING@

Moves the cursor to the beginning of the Words document

Format WP_GOTO_BEGINNING@()

Method this.goto_beginning@

Description Places the cursor before the first character in the current Words document and updates the Words display to show the first page in the document. WP_GOTO_BEGINNING@ is called by Edit ® Go To ® Beginning.

See also [WP_GOTO_END@](#)

WP_GOTO_BEGIN_LINE@

Moves the cursor to the beginning of the current line

Format WP_GOTO_BEGIN_LINE@()

Method this.goto_begin_line@

Description Moves the cursor so that it is placed before the first character in the current line. WP_GOTO_BEGIN_LINE@ is called by Keys ® Beginning of line.

See also [WP_GOTO_END_OF_LINE@](#)

WP_GOTO_BOTTOM_OF_PAGE@

Moves the cursor to the bottom of the current page

Format WP_GOTO_BOTTOM_OF_PAGE@()

Method [this.goto_bottom_of_page@](#)

Description Moves the cursor so that it is placed after the last character on the current page and updates the display so that this position is visible. If the page ends with a paragraph marker, it is placed immediately before that paragraph marker.

See also [WP_GOTO_TOP_OF_PAGE@](#)

WP_GOTO_BOTTOM_OF_SCREEN@

Moves cursor to the bottom of the screen

Format WP_GOTO_BOTTOM_OF_SCREEN@()

Method [this.goto_bottom_of_screen@](#)

Description Moves the cursor to the bottom of the current screen.

See also [WP_GOTO_TOP_OF_SCREEN@](#)

WP_GOTO_END@

Moves the cursor to the end of the document

Format WP_GOTO_END@()

Method [this.goto_end@](#)

Description Places the cursor after the last character in the current Words document and updates the Words display to show this position. WP_GOTO_END@ is called by Edit ® Go To ® End.

See also [WP_GOTO_BEGINNING@](#)

WP_GOTO_END_OF_LINE@

Moves the cursor to the end of the current line

Format WP_GOTO_END_OF_LINE@()

Method [this.goto_end_of_line@](#)

Description Moves the cursor to the next end-of-line occurrence.

See also [WP_GOTO_BEGIN_LINE@](#)

WP_GOTO_END_OF_PARA@

Moves the cursor to the end of the current paragraph

Format WP_GOTO_END_OF_PARA@()

Method [this.goto_end_of_para@](#)

Description Places the cursor after the last character of the current paragraph in a Words document and, if necessary, updates the display.

See also [WP_GOTO_START_OF_PARA@](#)

WP_GOTO_EVEN_FOOTER@

Moves cursor to the even footer

Format WP_GOTO_EVEN_FOOTER@()

Method [this.goto_even_footer@](#)

Description Attempts to move the cursor to the even footer defined in the current section. It is up to the application to ensure that the footer has been defined before moving to it.

See also [WP_GOTO_ODD_FOOTER@](#)
[Defining Headers and Footers](#)

WP_GOTO_EVEN_HEADER@

Moves cursor to the even header

Format WP_GOTO_EVEN_HEADER@()

Method [this.goto_even_header@](#)

Description Attempts to move the cursor to the even header defined for the current section. It is up to the application to ensure that the header has been defined before moving to it.

See also [WP_GOTO_ODD_HEADER@](#)
[Defining Headers and Footers](#)

WP_GOTO_FIRST_CELL@

Moves cursor to the first cell in a row

Format WP_GOTO_FIRST_CELL@(bead)

Method [this.goto_first_cell@\(bead\)](#)

Arguments bead A bead in any cell of a table.

Description Moves the cursor to the first cell in the row containing bead. An error occurs if bead is not in a table.

WP_GOTO_FIRST_FOOTER@

Moves cursor to the first footer

Format WP_GOTO_FIRST_FOOTER@()

Method [this.goto_first_footer@](#)

Description Moves the cursor to first footer of the current section if the footer was defined as "first page different". It is up to the application to ensure that the footer has been defined before moving to it.

See also [WP_GOTO_LAST_FOOTER@](#)
[Defining Headers and Footers](#)

WP_GOTO_FIRST_HEADER@

Moves cursor to the first header

Format WP_GOTO_FIRST_HEADER@()

Method [this.goto_first_header@](#)

Description Moves the cursor to the first header of the current section if the header was defined as "first page different". It is up to the application to ensure that the header has been defined before moving to it.

See also [WP_GOTO_LAST_HEADER@](#)
[Defining Headers and Footers](#)

WP_GOTO_FIRST_PAGE@

Displays the first page

Format WP_GOTO_FIRST_PAGE@()

Method [this.goto_first_page@](#)

Description Moves the cursor to the top of the first page in the document and then displays this first page.

See also [WP_GOTO_LAST_PAGE@](#)
[WP_GOTO_PAGE@](#)

WP_GOTO_LAST_FOOTER@

Moves cursor to the last footer

Format WP_GOTO_LAST_FOOTER@()

Method [this.goto_last_footer@](#)

Description Moves the cursor to the last footer of the current section. It is up to the application to ensure that the footer has been defined before moving to it.

See also [WP_GOTO_FIRST_FOOTER@](#)
[Defining Headers and Footers](#)

WP_GOTO_LAST_HEADER@

Moves cursor to the last occurrence of a header

Format WP_GOTO_LAST_HEADER@()

Method [this.goto_last_header@](#)

Description Moves the cursor to the last header in a document. It is up to the application to ensure that the header has been defined before moving to it.

See also [WP_GOTO_FIRST_HEADER@](#)
[Defining Headers and Footers](#)

WP_GOTO_LAST_PAGE@

Displays the last page of your document

Format WP_GOTO_LAST_PAGE@()

Method [this.goto_last_page@](#)

Description Moves the cursor to the top of the last page of the document. It also displays this location.

See also [WP_GOTO_FIRST_PAGE@](#)
[WP_GOTO_PAGE@](#)

WP_GOTO_NEXT_INSET@

Moves the cursor to the next object field

Format WP_GOTO_NEXT_INSET@()

Method [this.goto_next_inset@](#)

Description Searches forward from the current cursor position for the next linked or embedded object field in the current Words document. If a linked or embedded object is field is found, the cursor is moved to the inset. In no objects are found between the current cursor position and the end of the document, an error is thrown. WP_GOTO_NEXT_INSET@ is called by Edit ® Go To ® Next Object.

See also [WP_GOTO_PREV_INSET@](#)

WP_GOTO_NEXT_CHANGE@

Move the cursor to the next change bar in the current section

Format WP_GOTO_NEXT_CHANGE@()

Description Moves the cursor to the next change bar in a flow.

WP_GOTO_NEXT_DOC@

Opens the next document in the history list

Format WP_GOTO_NEXT_DOC@()

Description Closes the current document and opens the next document in the history list. Applixware Words maintains a list of files that you have opened in the current session. You can move to the last file you opened by calling the function [WP_GOTO_PREV_DOC@](#). Once you have moved to a previously-opened document, you can use WP_GOTO_NEXT_DOC@ to move to the next document in the list.

WP_GOTO_NEXT_LINE@

Moves cursor to the beginning of the next line

Format WP_GOTO_NEXT_LINE@()

Method [this.goto_next_line@](#)

Description Moves the cursor to so that it precedes the first character on the *next* line of the document.

See also [WP_GOTO_PREV_LINE@](#)

WP_GOTO_NEXT_PAGE@

Moves the cursor to the next page in a Words document

Format WP_GOTO_NEXT_PAGE@()

Method [this.goto_next_page@](#)

Description Moves the cursor to the next page in the current Words document based on the page in which the cursor currently resides. The cursor is placed at the top of the next page and the page is displayed.

See also [WP_GOTO_PREV_PAGE@](#)
[WP_GOTO_PAGE@](#)

WP_GOTO_PREV_CHANGE@

Move the cursor to the previous change bar in the current flow

Format WP_GOTO_PREV_CHANGE@()

Description Moves the cursor to the previous change bar in a flow. Footers, headers, and footnotes are ignored.

WP_GOTO_PREV_DOC@

Opens the previous document in the history list

Format WP_GOTO_PREV_DOC@()

Description Closes the current document and opens the previous document in the history list. Applixware Words maintains a list of files that you have opened in the current session. You can move to the last file you opened by calling the function WP_GOTO_PREV_DOC@(). Once you have moved to a previously-opened document, you can use [WP_GOTO_NEXT_DOC@](#) to move to the next document in the list.

WP_GOTO_NEXT_PARA@

Moves the cursor to the next paragraph

Format WP_GOTO_NEXT_PARA@()

Method [this.goto_next_para@](#)

Description Moves the cursor so that it precedes the first character in the next paragraph. (Stated in another way, the cursor moves one character past the paragraph marker at the end of the current paragraph.) If this new position is not currently displayed, the display is adjusted so the cursor is visible.

See also [WP_GOTO_NEXT_PLACE@](#)
[WP_GOTO_PREV_PARA@](#)

WP_GOTO_NEXT_PLACE@

Moves the cursor to the next paragraph

Format flag = WP_GOTO_NEXT_PLACE@()

Method flag = this.goto_next_place@

Description Moves the cursor to the next paragraph. If the movement is successful, TRUE is returned. Otherwise, this macro returns FALSE.

See also [WP_GOTO_NEXT_PARA@](#)

WP_GOTO_NEXT_SCREEN@

Moves the cursor to the top of the next screen

Format WP_GOTO_NEXT_SCREEN@()

Method this.goto_next_screen@

Description Moves the cursor to the top of the next screen and then re-displays that location.

See also [WP_GOTO_PREV_SCREEN@](#)

WP_GOTO_NEXT_SECTION@

Moves the cursor to the next section

Format moveFlag = WP_GOTO_NEXT_SECTION@()

Method moveFlag = this.goto_next_section@

Description Searches forward from the current cursor location for the next section in the current Words document. If it is found, the cursor is placed at the start of the first line in the section and the window is updated to show the page in which the cursor

resides. `WP_GOTO_NEXT_SECTION@` is called by Edit ® Go To ® Next Section. If the current section is the last section, this macro beeps.

If this macro is successful, it returns TRUE. If it is not successful, it returns FALSE.

See also [WP_GOTO_PREV_SECTION@](#)

WP_GOTO_NEXT_WORD@

Moves the cursor to the next word

Format `WP_GOTO_NEXT_WORD@()`

Method `this.goto_next_word@`

Description Moves the cursor so that it is placed immediately before the first character of the next word.

See also [WP_GOTO_PREV_WORD@](#)

WP_GOTO_NTH_PARA@

Moves to *nth* paragraph

Format `WP_GOTO_NTH_PARA@(num)`

Method `this.goto_nth_para@(num)`

Arguments `num` The paragraph to move to.

Description Moves the cursor to a paragraph *num* in the document. The first paragraph in the document is paragraph 0, not paragraph 1. Use this function to move to a paragraph by its position relative to the beginning of the document rather than relative to the current cursor position.

If *num* is greater than the number of paragraphs in the document, this macro beeps; no other action will occur.

See also [WP_GO_PARA_NUM@](#)

WP_GOTO_ODD_FOOTER@

Moves the cursor to the odd footer

Format `WP_GOTO_ODD_FOOTER@()`

Method `this.goto_odd_footer@`

Description Attempts to move the cursor to the odd footer of the current section. It is up to the application to ensure that the footer is defined before moving to it.

See also [WP_GOTO_EVEN_FOOTER@](#)

WP_GOTO_ODD_HEADER@

Moves the cursor to the odd header

Format WP_GOTO_ODD_HEADER@()

Method [this.goto_odd_header@](#)

Description Attempts to move the cursor to the odd header of the current section. It is up to the application to ensure that the header is defined before moving to it.

See also [WP_GOTO_EVEN_HEADER@](#)

WP_GOTO_PAGE@

Moves the cursor to an absolute page

Format WP_GOTO_PAGE@(page)

Method [this.goto_page@\(page\)](#)

Arguments page The page you want to go to.

Description Moves the cursor to a page in the document. The first page in the document is page 0, not page 1. Use this function to move to a page by its position relative to the beginning of the document rather than relative to the current page.

If page is greater than the number of pages in the document, this macro beeps; no other action will occur.

WP_GOTO_PAGE@ is called by Edit ® Go To ® Page.

See also [WP_GOTO_FIRST_PAGE@](#)

[WP_GOTO_LAST_PAGE@](#)

[WP_GOTO_NEXT_PAGE@](#)

[WP_GOTO_PAGE_NUMBER@](#)

[WP_GOTO_PREV_PAGE@](#)

WP_GOTO_PAGE_NUMBER@

Moves the cursor to an absolute page

Format WP_GOTO_PAGE_NUMBER@(page)

Method [this.goto_page_number@\(page\)](#)

Arguments page The page you want to go to.

Description Moves the cursor to a page in the document. Use this function to move to a page by its position relative to the beginning of the document rather than relative to the current page.

If page is greater than the number of pages in the document, this macro beeps; no other action will occur.

This macro is a replacement macro for WP_GOTO_PAGE@.

See also [WP_GOTO_FIRST_PAGE@](#)
[WP_GOTO_LAST_PAGE@](#)
[WP_GOTO_NEXT_PAGE@](#)
[WP_GOTO_PREV_PAGE@](#)

WP_GOTO_PREV_INSET@

Moves the cursor to the previous object field

Format WP_GOTO_PREV_INSET@()

Method [this.goto_prev_inset@](#)

Description Moves backward from the current cursor location to the previous occurrence of an linked or embedded object field in the current Words document. If found, the cursor is placed at the start of the object field and the window is updated to show the page in which the cursor resides. WP_GOTO_PREV_INSET@ is called by Edit ® Go To ® Previous Field.

See also [WP_GOTO_NEXT_INSET@](#)

WP_GOTO_PREV_LINE@

Moves to the previous line

Format WP_GOTO_PREV_LINE@()

[Method](#) `this.goto_prev_line@`

Description Sets the cursor at the start of the previous line.

See also [WP_GOTO_NEXT_LINE@](#)

WP_GOTO_PREV_PAGE@

Moves the cursor to the previous page

Format `WP_GOTO_PREV_PAGE@()`

[Method](#) `this.goto_prev_page@`

Description Moves the cursor to the previous page in the current Words document based on the page in which the cursor currently resides. The cursor is placed at the top of the previous page and the page is displayed.

See also [WP_GOTO_PAGE@](#)
[WP_GOTO_NEXT_PAGE@](#)

WP_GOTO_PREV_PARA@

Moves the cursor to the previous paragraph

Format `WP_GOTO_PREV_PARA@()`

[Method](#) `this.goto_prev_para@`

Description Moves the cursor to the beginning of the previous paragraph. A beep occurs if the cursor is located within a document's first paragraph.

See also [WP_GOTO_NEXT_PARA@](#)

WP_GOTO_PREV_SCREEN@

Moves the cursor to the top of the previous screen

Format `WP_GOTO_PREV_SCREEN@()`

[Method](#) `this.goto_prev_screen@`

Description Moves the cursor to the top of the previous screen and then re-displays that location.

See also [WP_GOTO_NEXT_SCREEN@](#)

WP_GOTO_PREV_SECTION@

Moves the cursor to the previous section

Format WP_GOTO_PREV_SECTION@()

Method [this.goto_prev_section@](#)

Description Moves backward from the current cursor location to the previous occurrence of a section in the current Words document. If found, the cursor is placed at the start of the first line in the section and the window is updated to show the page in which the cursor resides. WP_GOTO_PREV_SECTION@ is called by Edit ® Go To ® Previous Section. A beep occurs if the cursor is in the document's first section.

See also [WP_GOTO_NEXT_SECTION@](#)

WP_GOTO_PREV_WORD@

Moves the cursor to the beginning of a word

Format WP_GOTO_PREV_WORD@()

Method [this.goto_prev_word@](#)

Description Moves the cursor back to the beginning of the current word. If the cursor is already at the beginning of a word, it is moved to the beginning of the previous word.

See also [WP_GOTO_NEXT_WORD@](#)

WP_GOTO_PREV_PLACE@

Goes to beginning of the paragraph

Format flag = WP_GOTO_PREV_PLACE@()

Method flag = [this.goto_prev_place@](#)

Description Move the cursor to the beginning of the current paragraph. If the cursor is already at the beginning of the paragraph, go to the beginning of the previous paragraph. TRUE is returned if the cursor moves to the previous place. FALSE means the cursor is at the beginning of the document.

See also [WP_GOTO_START_OF_PARA@](#)

WP_GOTO_START_OF_LINE@

Moves the cursor to the beginning of the line

Format WP_GOTO_START_OF_LINE@()

Method [this.goto_start_of_line@](#)

Description Moves the cursor from its current location so that it is placed immediately in front of the line's first character. If the cursor is already at the beginning of the line, no further action is taken.

See also [WP_GOTO_END_OF_LINE@](#)

WP_GOTO_START_OF_PARA@

Moves the cursor to the beginning of the paragraph

Format WP_GOTO_START_OF_PARA@()

Method [this.goto_start_of_para@](#)

Description Moves the cursor from its current location so that it is placed immediately in front of the paragraph's first character. If the cursor is already at the beginning of the paragraph, no further action is taken.

See also [WP_GOTO_END_OF_PARA@](#)
[WP_GOTO_PREV_PLACE@](#)

WP_GOTO_TOP_OF_PAGE@

Moves the cursor to the top of the current page

Format WP_GOTO_TOP_OF_PAGE@()

Method [this.goto_top_of_page@](#)

Description Moves the cursor from its current location so that it is placed immediately in front of the page's first character. If the cursor is already at the beginning of the page, no further action is taken.

See also [WP_GOTO_BOTTOM_OF_PAGE@](#)

WP_GOTO_TOP_OF_SCREEN@

Moves the cursor to the top of the screen

Format WP_GOTO_TOP_OF_SCREEN@()

Method [this.goto_top_of_screen@](#)

Description Moves the cursor to the top of screen. It will precede the first character displayed in the main flow. The cursor will not move into a header, footer, or footnote.

See also [WP_GOTO_BOTTOM_OF_SCREEN@](#)

WP_GO_PARA_NUM@

Moves cursor to a specified paragraph number

Format WP_GO_PARA_NUM@(number)

Method [this.go_para_num@\(number\)](#)

Arguments number The paragraph to which you are moving the cursor.

Description Moves the cursor to a paragraph. The first paragraph in the document is paragraph 0. This means WP_GO_PARA_NUM@(10) moves the cursor to the eleventh paragraph in the document.

See also [WP_GOTO_NTH_PARA@](#)

WP_GRAPHICS_INSET@

Places a new embedded Graphics object at the cursor position

Format WP_GRAPHICS_INSET@()

Method [this.graphics_inset@](#)

Description Creates a new Graphics object at the current cursor position in the current Words document. The Graphics window is displayed so that a graphic can be created.

See also [WP_INSERT_GR_INSET@](#)

WP_HARD_SPACE@

Places a non-breaking space at the current cursor location

Format WP_HARD_SPACE@()

Method [this.hard_space@](#)

Description When a non-breaking space is set, Words will not break a line at that space.

WP_HAVE_SELECTION@

Indicates if any text is selected

Format flag = WP_HAVE_SELECTION@()

Method [flag = this.have_selection@](#)

Description Returns TRUE if any text is selected in the current Words document. It returns FALSE if no text is selected.

See also [WP_GET_CURRENT_WORD@](#)

WP_HELP_HYPERLINK@

Requests a help topic from Applixware Help

Format WP_HELP_HYPERLINK@(topicString)

Arguments topicString A string containing the name of a topic in Applixware Help.

Description Requests a topic from Applixware Help. If the topic is found by the Applixware Help system, it opens a Help window and displays the topic. If the topic is not found. The following error appears on the screen:

No Help is available for: *your_topic_name*

Your_topic_name is the name of the help topic for which you requested information.

WP_HSB_TO_RGB@

Changes color from HSB system to RGB system

Format WP_HSB_TO_RGB@(hue, saturation, brightness, red, green, blue)

Method [this.hsb_to_rgb@](#)(hue, saturation, brightness, red, green, blue)

Arguments

hue	A color's hue value.
saturation	A color's saturation value.
brightness	A color's brightness value.
red	A color's red value. This is an output value.
green	A color's green value. This is an output value.
blue	A color's blue value. This is an output value.

Description Transforms a color expressed in the HSB system to a color expressed in the RGB system.

See also [WP CMYK TO RGB@](#)
[WP RGB CMYK@](#)
[WP RGB HSB@](#)

WP_HYPERLINK@

Locates a hypertext in the current document

Format flag = WP_HYPERLINK@(array)

Method [flag = this.hyperlink@](#)(array)

Arguments

array	A three-item array defining the hyperlink, as follows:
array[0]	The link name
array[1]	The path name
array[2]	The name of a profile variable indicating a location to be concatenated with element 1.

Description Locates a hypertext in the current document. If the hypertext exists, this macro returns TRUE and moves the cursor to the hypertext's location. If the hypertext does not exist, FALSE is returned and no cursor movement occurs. The hyperlink and hypertexts are usually inserted into a document using the Utilities® Special Fields ® Hyperlink and ® Hypertext commands.

Only the first element of array is required. If elements 2 and 3 are omitted, then Words assumes that the hypertext is in the current file. If you are only specifying the line name, the argument to this macro need not be an array. That is:

```
var foo
foo = "aLinkName"
WP_HYPERLINK@(foo)
```

If element 2 exists, Words assumes that the hypertext is in the file named by element 2.

In some cases, you may want to defer naming where the location of the hypertext will be. The optional third argument allows you to name a profile variable that contains the location of the file.

NOTE: Hypertargets are inserted using [WP_INSERT_FIELD@](#). For example:

```
method = WP#HYPER#TARGET#FIELD_STRING++
        " \'" ++ text ++ "'"
WP_INSERT_FIELD@(method)
```

NOTE: This macro, like all macros, passes the macro's bead number to ELF. This bead number is the second argument to this macro. However, it is not used and, consequently, is ignored.

WP_ILLEGALIZE_LOCATION@

Converts a legal location into an end-of-range

Format last = WP_ILLEGALIZE_LOCATION@(format wp_location@ loc)

Method last = this.illegalize_location@(format wp_location@ loc)

Arguments loc The location being converted.

Description Converts a "legal" location into an end-of-range location. This function is normally used as follows:

```
range.last = WP_ILLEGALIZE_LOCATION@( loc)
```

See also [WP_LEGALIZE_LOCATION@](#)

WP_ILLEGAL_FOR_TABLE@

Can a range be surrounded by a row or cell

Format string = WP_ILLEGAL_FOR_TABLE@([format [wp_range@](#) range])

Method string = this.illegal_for_table@([format wp_range@ range])

Arguments range An optional range specification.

Description Indicates if the passed range can be surrounded by a row or cell. If range is NULL, the current selection is examined.

If the examined material cannot be placed in a cell (e.g., it contains a page break, a cell, etc.), an error string is returned. If the material can be placed in a cell, this macro returns NULL or an empty string.

WP_IMPORT_*@

Imports a copy of the non-Appixware file into Appixware Words

Format WP_IMPORT_*(filename, revert)

Method [this.import_*](#)*(filename, revert)

Arguments filename The full path name of the file to import.
revert Indicates whether to place the imported document in the current Words window or a new Words window. If revert is set to TRUE, the imported document replaces the current document. The current Words document is not saved. If set to FALSE, a new Words window is created and the imported file is placed in the new window.

Description Imports a copy of the non-Appixware file you specify into Appixware Words. The following import file formats are supported:

WP_IMPORT_ASCII_LINES@ ASCII lines

WP_IMPORT_ASCII_PARAS@ ASCII paragraphs

WP_IMPORT_HTML@ HTML (HyperText Markup Language); this is discussed separately

WP_IMPORT_MSWORD70@ MS Word for Windows 7.0 and MS Word for Macintosh 6.0 formats.

WP_IMPORT_RTF@ Rich Text Format

WP_IMPORT_URL@ Imports a WEB .html file into Words; this macro is documented separately

WP_IMPORT_WINWORD60@ MS Word for Windows versions 6.0 and 7.0. Some newer 7.0 features may be ignored.

For more information about Words filters, refer to the *Words* manual. See also FILTER_*_@ and **WP_EXPORT_***@.

WP_IMPORT_HTML@

Imports a copy of an HTML document into Applixware Words

Format WP_IMPORT_HTML@(filename[, revert [, URL]])

Method [this.import_html@](#)(filename[, revert [, URL]])

Arguments

filename	The name of an HTML file.
revert	Indicates whether to place the imported document in the current Words window or a new Words window. If revert is set to TRUE, the imported document replaces the current document. The current Words document is not saved. If set to FALSE, a new Words window is created and the imported file is placed in the new window.
URL	The file's URL if it was accessed using WWW. The base URL is used to resolve any relative URLs contained within filename.

Description Converts an HTML file into a Words document and displays it within a Words window.

See also [FILTER HTML TO WP@](#)
[FILTER URL TO GR@](#)
[FILTER URL TO WP@](#)
[WP_IMPORT_URL@](#)
[WWW_FETCH_URL@](#)
[WWW_MERGE_URL@](#)

WP_IMPORT_URL@

Fetches and imports an HTML document

Format WP_IMPORT_URL@(url[, revert])

Method [this.import_url@](#)(url[, revert])

Arguments

url	The URL naming the location of an HTML document on the Web.
revert	Indicates whether to place the imported document in the current Words window or a new Words window. If revert is set to TRUE, the imported document replaces the current document. The current Words document is not saved. If set to FALSE, a new Words window is created and the imported file is placed in the new window.

Description Retrieves an HTML document from a Website, then converts it into a Words document and displays it within a Words window.

WP_IN_FOOTNOTE@

Indicates if cursor is in a footnote

Format flag = WP_IN_FOOTNOTE@()

Method flag = this.in_footnote@

Description Returns TRUE if the current selection or cursor is in a footnote.

See also [WP_IN_HDRFTR@](#)
[WP_IN_MAIN_FLOW@](#)
[WP_IN_SELECTION_MARKS@](#)

WP_IN_HDRFTR@

Indicates if cursor is in a header or footer

Format flag = WP_IN_HDRFTR@()

Method flag = this.in_hdrftr@

Description Returns TRUE if the current selection or cursor is in a header or footer.

See also [WP_IN_FOOTNOTE@](#)
[WP_IN_MAIN_FLOW@](#)
[WP_IN_SELECTION_MARKS@](#)

WP_IN_MAIN_FLOW@

Indicates if cursor is in the main flow

Format flag = WP_IN_MAIN_FLOW@()

Method flag = this.in_main_flow@

Description Returns TRUE if the current selection or cursor is in the main flow. That is, if it is not in a footnote, a header, or a footer.

See also [WP_IN_FOOTNOTE@](#)
[WP_IN_HDRFTR@](#)

WP_IN_SELECTION_MARKS@

WP_IN_SELECTION_MARKS@

Indicates if location is within a marked selection

Format flag = WP_IN_SELECTION_MARKS@(name, format wp_location@ loc, num)

Method flag = this.in_selection_marks@(name, format wp_location@ loc, num)

Arguments

name	The name used in WP_SET_SELECTION_MARKS@ .
loc	The location.
num	The selection number. This is a returned value.

Description Determines if the passed location is within a marked selection. It returns TRUE if it is, or FALSE if it is not. If TRUE, the selection number that the location falls within is output.

See also [WP_IN_FOOTNOTE@](#)
[WP_IN_HDRFTR@](#)
[WP_IN_MAIN_FLOW@](#)

WP_INDENT@

Indents the current text

Format WP_INDENT@()

Method this.indent@

Description Indents the current or selected paragraphs by one level. The amount by which the paragraphs is indented is specified using the Format ® Document Settings Indent per Paragraph Level text entry field. By default this value is .5 inches.

If changing the indent level sets a paragraph's to the same level it would have had if it had inherited the level, then the indentation is set as being *inherited* rather than *local*.

WP_INHERIT_TABSTOPS@

Forces all selected paragraphs to inherit their tabstops

Format WP_INHERIT_TABSTOPS@()

Method this.inherit_tabstops@

Description Changes all tabs in the selected paragraphs so they are only those that are allowed by the paragraph's style. Local tabs are removed.

WP_INIT_PARAGRAPH_BEAD@

Initializes and returns a paragraph bead

Format bead = WP_INIT_PARAGRAPH_BEAD@(localFlag)

Method bead = this.init_paragraph_bead@(localFlag)

Arguments localFlag If set to TRUE, all attributes are local. If set to FALSE all attributes are inherited.

Description Initializes and returns a paragraph bead. If localFlag is set to TRUE, all of the returned beads will be local. Because you have not yet had a chance to set them, they are set to a *reasonable* value. These default values are:

no bold	no italic	no underline	no strike through
no hidden	no underline	position 0	hyphenate
color = Black	font = Times	size = 12	no tag
no page break	no keep	justify left	not in TOC
indentToLevel	no spell	first indent = 0	left indent = 0
right indent = 0	para spacing 0	line spacing = 0	level = 0
hyphZone = 1000	hyph fragment = 3		

WP_INIT_PARA_ATTRS@

Initializes paragraph attributes to reasonable values

Format format wp_para_attrs@atts = WP_INIT_PARA_ATTRS@([localFlag])

Method format wp_para_attrs@atts = this.init_para_attrs@([localFlag])

Arguments localFlag A Boolean value which if set to TRUE indicates that all attributes are local. The default is FALSE, indicating that attributes are inherited.

Description Initializes paragraph attributes to reasonable values. (This function is called by [WP_INIT_PARAGRAPH_BEAD@](#)) The following paragraph attribute values are returned as a wp_para_attrs@ format:

pageBreak	= FALSE
keepWith	= FALSE
justify	= WP#JUSTIFY#LEFT_
inTOC	= FALSE
indentToLevel	= TRUE
noSpell	= FALSE
firstIndent	= 0
leftIndent	= 0
rightIndent	= 0
preParaSpacing	= 0
postParaSpacing	= 0
lineSpacing	= 0
level	= 0
hyphZone	= 1000
hyphMinFrag	= 3

See also [WP_GET_PARA_SETTINGS@](#)

WP_INIT_PARA_ATTR_FLAGS@

Initializes paragraph attribute flags

Format format wp_para_attrs_flags@ info = WP_INIT_PARA_ATTR_FLAGS@
(localFlag)

Method format wp_para_attrs_flags@ info = this.init_para_attr_flags@ (localFlag)

Arguments localFlag If set to TRUE, all attributes are local. If set to FALSE all attributes are inherited.

Description Initializes and returns a paragraph attributes flag format (wp_para_attrs_flags@). WP_INIT_PARA_ATTR_FLAGS@ sets the attributes to all inherit, or all local with reasonable values. For example, the following items are set as follows and are returned as a wp_para_attrs_flag@ format:

pageBreak	= localFlag
keepWith	= localFlag
justify	= localFlag
inTOC	= localFlag
indentToLevel	= localFlag
noSpell	= localFlag
firstIndent	= localFlag
leftIndent	= localFlag
rightIndent	= localFlag

preParaSpacing = localFlag
 postParaSpacing = localFlag
 lineSpacing = localFlag
 level = localFlag
 hyphZone = localFlag
 hyphMinFrag = localFlag

See also [WP_GET PARA SETTINGS@](#)

WP_INIT_STYLE_BEAD@

Initializes and returns a style bead

Format format wp_style@ style = WP_INIT_STYLE_BEAD@(name, parent)

Method format wp_style@ style = this.init_style_bead@(name, parent)

Arguments name The name of the style being created.
 parent The style from which it will inherit attributes.

Description Creates, initializes, and returns a style bead. The style's attributes are set to "all inherits"; it will be its own next style. The returned value has a wp_style@ format. This style must then be added using [WP_SET STYLE@](#).

WP_INIT_TEXT_ATTRS@

Initializes text attributes

Format format wp_text_attrs@ attrs= WP_INIT_TEXT_ATTRS@()

Method format wp_text_attrs@ attrs= this.init_text_attrs@

Description Initializes text attributes to reasonable values. It also returns these values as a wp_text_attrs@ format.

This format along with the default attributes set by this macro are as follows:

wp_text_attrs@	
bold,	FALSE
italic,	FALSE
strikethru,	FALSE
hidden,	FALSE
capitals,	WP#CAPS#NORMAL_
underline,	WP#UNDER#NONE
position,	0
hyphenate,	WP#HYPHENATE#NONE_
color,	Black

face,	WP#FONT#TIMES_
size,	12
tag,	""

WP_INIT_TEXT_ATTR_FLAGS@

Initializes text attribute flags

Format format wp_text_attrs_flags@ attrs = WP_INIT_TEXT_ATTR_FLAGS@
(localFlag)

Method format wp_text_attrs_flags@ attrs = this.init_text_attr_flags@ (localFlag)

Arguments localFlag If set to TRUE, all attributes are local. If set to FALSE all attributes are inherited.

Description Initializes and returns a wp_text_attrs_flags@ formatted variable. WP_INIT_TEXT_ATTR_FLAGS@ sets the attributes to all inherit, or all local with reasonable values. See [WP_INIT_TEXT_ATTRS@](#) for more information.

The wp_text_attrs_flags@ format is as follows. Each element is set to localFlag when it is returned by this macro.

```
format wp_text_attrs_flags@
    wp_bold,
    italic,
    strikethru,
    hidden,
    capitals,
    underline,
    position,
    hyphenate,
    color,
    face,
    size,
    tag
```

WP_INIT_TEXT_BEAD@

Initializes and returns a text bead

Format format wp_bead@ bead = WP_INIT_TEXT_BEAD@(localFlag[, text])

Method format wp_bead@ bead = this.init_text_bead@(localFlag[, text])

Arguments local If set to TRUE, all attributes are local. If set to FALSE all attributes are inherited.

text The starting string.

Description Initializes and returns a text bead (which has a format of wp_bead@). WP_INIT_TEXT_BEAD@ sets the attributes to all inherit, or all local with reasonable values.

WP_INITIALIZE_DOC@

Sets the language mode for a document

Format WP_INITIALIZE_DOC@()

Method [this.initialize_doc@](#)

Description Sets the language mode for a document that is presumed to have just been opened. This macro should be called by all macros that open and create new Applixware documents.

It also resets the document's *modified* flag so that if a document is closed at the point, the user will not be asked to save the file.

WP_INSERT_ADVISORY_HYPHEN@

Inserts a soft-hyphen field

Format WP_INSERT_ADVISORY_HYPHEN@()

Method [this.insert_advisory_hyphen@](#)

Description Inserts a soft-hyphen field at the cursor position.

WP_INSERT_AUDIO_INSET@

Inserts audio data

Format WP_INSERT_AUDIO_INSET@(audioFilename[, comment[, suppressIconFlag[, iconFileName[, appMacro]]]])

Method [this.insert_audio_inset@\(audioFilename\[, comment\[, suppressIconFlag\[, iconFileName\[, appMacro\] \] \] \]\)](#)

Arguments audioFilename

The file name in which the audio information is stored.

comment An optional comment describing the audio object

suppressIconFlag

An optional Boolean value indicating an audio icon is to be displayed.

iconFileName

The path name of the icon.

appMacro

A string indicating the name of an ELF macro that will be executed when a user double-clicks on the embedded object field. This field is optional

Description Creates and inserts an audio embedded object field that contains the passed audio data. The structure of this field is:

```
{local_insert "filename" -comment "text" -noicon}
```

or

```
{local_insert "filename" -comment "text" -iconFileName "text"}
```

Only the first argument (filename) is required.

See also [WP_INSERT_AUDIO_INSET_FILE@](#).

WP_INSERT_AUDIO_INSET_FILE@

Inserts audio file

Format WP_INSERT_AUDIO_INSET_FILE@(data[, comment[, suppressIconFlag[, iconFileName[, appMacro]]]])

Method [this.insert_audio_inset_file@](#)(data[, comment[, suppressIconFlag[, iconFileName[, appMacro]]]])

Arguments data

Audio data.

comment

An optional comment describing the audio insert

suppressIconFlag

An optional Boolean value indicating an audio icon is to be displayed.

iconFileName

The path name of the icon.

appMacro

A string indicating the name of an ELF macro that will be executed when a user double-clicks on the embedded object field. This field is optional

Description Creates and inserts an audio inset that contains the passed audio file. This object is inserted as an embedded field. The structure of this field is:

```
{local_insert "filename" -comment "text" -noicon}
```

or

```
{local_insert "filename" -comment "text" -iconFilename "text"}
```

Only the first argument (data) is required.

See also [WP_INSERT_AUDIO_INSET@](#).

WP_INSERT_BEAD@

Inserts a single bead

Format beadNum = WP_INSERT_BEAD@(format wp_bead@ bead, format [wp_location@](#) loc)

Method beadNum = this.insert_bead@(format wp_bead@ bead, format wp_location@ loc)

Arguments bead The bead to be inserted.
loc The location to insert the bead.

Description Returns the bead number of the inserted bead. Note that inserting the wrong beads at the wrong location will cause severe problems.

See also [WP_DELETE_BEAD@](#)
[WP_INSERT_BEADS@](#)

WP_INSERT_BEADS@

Inserts multiple beads

Format range = WP_INSERT_BEADS@(format arrayof wp_bead@ beads, format [wp_location@](#) loc)

Method range = this.insert_beads@(format arrayof wp_bead@ beads, format wp_location@ loc)

Arguments beads An array of format wp_bead@.
loc The location to insert the beads.

Description Insert an array of beads into the document beginning at location loc. This function returns the range of the new beads. (This returned value has a [wp_range@](#) format.) Note that inserting the wrong beads at the wrong location will cause severe problems.

[WP_DELETE_BEADS@](#)
[WP_INSERT_BEAD@](#)

WP_INSERT_CELLS@

Inserts cells into an existing row

Format range = WP_INSERT_CELLS@(format arrayof wp_beads@ beads, format [wp_location@](#) loc, beforeCellFlag)

Method range = this.insert_cells@(format arrayof wp_beads@ beads, format wp_location@ loc, beforeCellFlag)

Arguments beads An array of format wp_bead@.
loc The location where the cell will be inserted.
beforeCellFlag If set to TRUE, and loc is in a cell, then the cells are inserted before the cell. If set to FALSE, and loc is in a cell, then the cells are inserted after the cell.

Description Inserts cells into an existing row. The passed location must be within an existing cell. Do not use this function to create new rows. (Use [WP_INSERT_ROWS@](#).) The range of the new beads is returned. This range is of type [wp_range@](#).
The passed beads must represent a legal set of cells (for example, each cell must contain a paragraph marker, no frames in the cell, etc.) Insert an illegal set of beads or inserting beads at the wrong location can cause severe problems.

See also [WP_DELETE_CELLS@](#)

WP_INSERT_CELLS_AT_SELECTION@

Inserts empty copy of cells, rows, or columns

Format WP_INSERT_CELLS_AT_SELECTION@(insertType, insertNum, beforeFlag, colPolicy)

Method this.insert_cells_at_selection@(insertType, insertNum, beforeFlag, colPolicy)

Arguments insertType The type of insertion, as follows:
0 WP_INSCCELL_ROWS
insert entire rows
1 WP_INSCCELL_COLS
insert entire columns
2 WP_INSCCELL_CELLS
insert cells
insertNum The number of items to insert.

beforeFlag A Boolean value which if set to TRUE indicates information is inserted before or above.

colPolicy The column insertion policy, as follows:

- 0 WP_INSCCELL_COLS_SAME
Insert only desired number of cells
- 1 WP_INSCCELL_COLS_FILL
Backfill short rows to get proper number of columns

Description Inserts empty copies of an existing cell, row, or column into the table. This is the low level function that actually does the insertion. It is strongly recommended that you use higher level functions if possible.

WP_INSERT_CLIPART@

Inserts graphic as embedded graphic object

Format WP_INSERT_CLIPART@(gfx)

Method [this.insert_clipart@\(gfx\)](#)

Arguments gfx A graphics handle

Description Inserts the graphic pointed to by gfx into the current Words window. The graphic is inserted as an embedded graphic object.

WP_INSERT_COLUMN_BREAK@

Inserts a column break

Format beadNum = WP_INSERT_COLUMN_BREAK@()

Method [beadNum = this.insert_column_break@](#)

Description Inserts a column break at the current cursor position.

If the cursor is within a table, this command breaks (inserts a paragraph marker) the table after the current row and then inserts a column break before the newly inserted paragraph marker.

The bead position of the column break is returned.

WP_INSERT_CNDL_VAR_FIELD@

Converts the selected text into a conditional variable field

Format WP_INSERT_CNDL_VAR_FIELD@(variableName)

Method [this.insert_cndl_var_field@](#)(variableName)

Arguments variableName The name of the string variable that will control the selection's visibility.

Description Transforms the selected text into a conditional field dependent on the value of variableName.

WP_INSERT_EMBEDDED_OBJ_FIELD@

Inserts an embedded object field

Format WP_INSERT_EMBEDDED_OBJ_FIELD@ (Name_in, Inframe, Cvmacro, Appmacro, View, Lastsection, Shrinktofit, Doctype, Appname, Picture)

Arguments

Name_in	The file name of the source of the object. For example, if you are embedding
Inframe	A Boolean. If TRUE (-1), the object is inserted in a frame.
Cvmacro	The macro to run to convert the object from its foreign file format to Applixware file format.
Appmacro	A integer value indicating the application wiuth which you want to open this inset when it is double-clicked in the word processor.
View	A string. This argument can be "RANGE", "OBJECT", or "VIEW"
Lastsection	IF TRUE, the last section bead of an Applixware Words inset is included in the inset. If FALSE, the last section bead is excluded.
Shrinktofit	If TRUE, the inset is made smaller to fit on the page. If FALSE the inset may be truncated if it is too large to fit on the page.
Doctype	A number indicating the type of document you are embedding. These number are in the file fileinf_.am.
Picture	Boolean. IF TRUE, insert the object as a picture. IF FALSE Insert the object as a table.

Description Insets an embedded inset into a Words document.

WP_INSERT_EQUATION@

Inserts an equation bead

Format WP_INSERT_EQUATION@()

Method [this.insert_equation@](#)

Description Inserts an empty equation bead into the Words document at the current cursor position.

See also [WP_INSERT_BEAD@](#)

WP_INSERT_EQUATION_INSET@

Inserts an equation editor field

Format WP_INSERT_EQUATION_INSET@(appMacro)

Method [this.insert_equation_inset@\(appMacro\)](#)

Arguments appMacro A string indicating the name of an ELF macro that will be executed when a user double-clicks on the embedded object field. This field is optional

Description Inserts an equation editor embedded object field in a Words document at the current cursor position.

WP_INSERT_FIELD@

Inserts a field

Format beadNum = WP_INSERT_FIELD@(method)

Method [beadNum = this.insert_field@\(method\)](#)

Arguments method The type of field you want to create and any arguments to the field.

Description Creates any type of field. The first word in the method identifies the field type. WP_INSERT_FIELD@ returns the Field Start bead number of new field.

WP_INSERT_FIELD_DATE_*@

Inserts the date at the current cursor location

Format WP_INSERT_FIELD_DATE_*@()

Method [this.insert_field_date_*@](#)

Description Inserts dates having various formats in the current Words document at the current cursor position. The macros and the formats in which they display the date are as follows:

WP_INSERT_FIELD_DATE_0@	mm/dd/yy
WP_INSERT_FIELD_DATE_1@	Mmm dd, yyyy
WP_INSERT_FIELD_DATE_2@	dd.mm.yy
WP_INSERT_FIELD_DATE_3@	dd Mmm yy
WP_INSERT_FIELD_DATE_4@	Mmmm dd, yyyy
WP_INSERT_FIELD_DATE_5@	yyyy-mm-dd
WP_INSERT_FIELD_DATE_6@	yy-mm-dd
WP_INSERT_FIELD_DATE_7@	yyyy mm dd
WP_INSERT_FIELD_DATE_8@	yy mm dd
WP_INSERT_FIELD_DATE_9@	yyyymmdd
WP_INSERT_FIELD_DATE_10@	yymmdd
WP_INSERT_FIELD_DATE_11@	dd.mm.yyyy

WP_INSERT_FIELD_PAGE_COUNT@

Inserts the total number of pages field

Format WP_INSERT_FIELD_PAGE_COUNT@()

Method [this.insert_field_page_count@](#)

Description Inserts a field whose value represents the total number of pages in the document. This number is always displayed as an arabic number.

WP_INSERT_FIELD_PAGE_NUMBER@

Inserts a page number field

Format WP_INSERT_FIELD_PAGE_NUMBER@()

Method [this.insert_field_page_number@](#)

Description Inserts a page number field at the current cursor position. The style used when displaying this page number is set using **WP SET PAGE NUMBERING FORMAT@**. The value displayed will be the current page number unless it has been changed using the same WP_SET_-PAGE_NUMBERING_FORMAT@ macro.

WP_INSERT_FIELD_TIME_0@

Inserts a time string in 24 hour format

Format WP_INSERT_FIELD_TIME_0@()

Method [this.insert_field_time_0@](#)

Description Inserts a string into the current Words document in the following format:
24hh:mm:ss.

WP_INSERT_FOOTNOTE@

Inserts a footnote

Format WP_INSERT_FOOTNOTE@(method)

Method [this.insert_footnote@\(method\)](#)

Arguments method The footnote's method; that is, the field method string for the footnote reference field being inserted.

Description Inserts a footnote in a Words document at the current cursor position. This field is displayed as follows:

```
footnote [-enddoc] [ [ -set value ] [-series "name" ]      ]  
                [ -text "text"                ]
```

WP_INSERT_FRAME@

Inserts a new frame

Format WP_INSERT_FRAME@()

Method [this.insert_frame@](#)

Description Inserts a frame in a Words document at the current cursor position. This new frame is defined as follows:

type	WP#FRAME#FLOW#AROUND
xposType	WP#FRAME#XPOS#LEFT
xposRelative	WP#FRAME#XREL#COLUMN
xpos	0
yposType	WP#FRAME#YPOS#TOP
yposRelative	WP#FRAME#YREL#PARA
ypos	0
widthType	WP#FRAME#WIDTH#AUTO
width	0
heightType	WP#FRAME#HEIGHT#AUTO
height	0
leftMargin	150 mils (1000 mils = 1 inch)
rightMargin	leftMargin (150 mils)
topMargin	0
bottomMargin	topMargin (0)

If, however, a paragraph style named Frame exists, this macro obtains these attributes from the style. However, this style is not used, if the user is defining the position and size using the mouse.

WP_INSERT_GR_INSET@

Inserts an unframed Graphics field

Format id = WP_INSERT_GR_INSET@(appMacro[, inFrameFlag])

Method id = this.insert_gr_inset@(appMacro[, inFrameFlag])

Arguments appMacro A string indicating the name of an ELF macro that will be executed when a user double-clicks on the object. This field is optional.

inFrameFlag A Boolean value which if set to TRUE indicates that a frame will surround the object. If a default frame style exists, it is used. If this value is omitted, the default is FALSE.

Description Inserts an embedded Graphics object field in a Words document at the current cursor position and opens the Graphics window. The ELF process ID is returned.

See also [WP_INSERT_GR_INSET_IN_FRAME@](#)

WP_INSERT_GR_INSET_FROM_GFX@

Inserts embedded graphics object

Format WP_INSERT_GR_INSET_FROM_GFX@(gfx[, appMacro[, inFrameFlag]])

Method this.insert_gr_inset_from_gfx@(gfx[, appMacro[, inFrameFlag]])

Arguments gfx The handle to the graphic object.
appMacro The name of the macro that is used to convert the graphics into Applixware format.
inFrameFlag A Boolean value which if set to TRUE indicates the embedded object is placed in a frame.

Description Embeds a graphic object into the current Words document. The graphic is displayed at the current cursor position. The object will be displayed within a frame if the optional inFrameFlag is set to TRUE.

If the graphic object is not in Applixware graphics format, it can be converted to it by specifying the optional appMacro argument, which names the graphics filter to be used.

WP_INSERT_GR_INSET_IN_FRAME@

Inserts a framed Graphic field

Format WP_INSERT_GR_INSET_IN_FRAME@(appmacro)

Method this.insert_gr_inset_in_frame@(appmacro)

Arguments appMacro A string indicating the name of an ELF macro that will be executed when a user double-clicks on the object. This field is optional.

Description Inserts a framed Graphics embedded object field in a Words document at the current cursor position and opens the Graphics window. See also [**WP_INSERT_GR_INSET@**](#).

WP_INSERT_LINE_BREAK@

Inserts a line break

Format WP_INSERT_LINE_BREAK@()

Method this.insert_line_break@

Description Inserts a line break (a hard carriage return) at the current cursor position.

WP_INSERT_LINK_FILE_FIELD@

Inserts a linked file field

Format WP_INSERT_LINK_FILE_FIELD@(name, inFrameFlag[, cvtMacro[, appMacro[, view[, lastSectionFlag[, shrinkToFitFlag[, docType[, appName]]]]]]])

Method [this.insert_link_file_field@](#)(name, inFrameFlag[, cvtMacro[, appMacro[, view[, lastSectionFlag[, shrinkToFitFlag[, docType[, appName]]]]]]])

Arguments

name	The name of the file being linked.
inFrameFlag	A Boolean value which if set to TRUE indicates that a frame will surround the object. If a default frame style exists, it is used.
cvtMacro	The name of the macro used to convert the information in name to a different format (for example, Applixware Words format). This field is optional.
appMacro	A string indicating the name of an ELF macro that will be executed when a user double-clicks on the object. This field is optional.
view	An optional range, view, or chart desired. If view is specified only the view or range indicated is displayed in the inset. (Views and ranges typically refer to Spreadsheets views and ranges.)
lastSectionFlag	An optional Boolean value which if set to TRUE indicates that the inset (which should be text) uses the same section properties as the last section.
shrinkToFitFlag	An optional Boolean value which if set to TRUE indicates that the rows (for Applixware Spreadsheets and Data) will be shrunk to fit within the margins if they are too wide.
docType	The document type as returned from RECOGNIZE_FILE@ .
appName	The name of the application, as follows: Audio Data_ Equations_ Graphics_ Macros_ Spreadsheets_ Words_ Internal_ (for all others)

Description Places an object of a given file format at the current cursor position in a Words document. When inserted the field has the following form:

```
{link [-cvt "cvtMacro"] [-app "appMacro"] [-view]
      [-lastSection] [-shrinkToFit] }
```

If you are linking a graphics inset, the field is made editable (i.e., unprotected) so that a bitmap bead is saved to a file. (This is needed by the filters.)

See also [WP_CHANGE_LINK@](#)
[WP_CHANGE_LINK_INFO@](#)
[WP_GET_LINKS@](#)
[WP_LINK_EXT *@](#)
[WP_SET_LINKS_INFO@](#)

WP_INSERT_MACRO_INSET@

Creates a Macro Editor object

Format WP_INSERT_MACRO_INSET@(appMacro)

Method [this.insert_macro_inset@\(appMacro\)](#)

Arguments appMacro The name of a macro

Description Creates a Macro Editor embedded object field containing appMacro. The Macro Editor is also opened.

WP_INSERT_MARKER_BEAD@

Creates a marker bead at the passed location

Format actualName = WP_INSERT_MARKER_BEAD@ (name, format wp_location@ loc[, saveToFileFlag])

Method [actualName = this.insert_marker_bead@ \(name, format wp_location@ loc\[, saveToFileFlag \]\)](#)

Arguments name The name for the new marker bead.
loc The location where you want to place the marker bead.
saveToFileFlag
A Boolean value which if set to TRUE indicates that the marker bead will be preserved in the output file when it is written to disk. FALSE, which is the default, tells Words that it should not write a marker bead to disk.

Description Creates a marker bead at location loc. The caller passes a suggested name for the bead; the actual name used is returned. (If the name you use is already used, a number is appended to name that creates a unique name.)

Note that the selection changes as a side-effect.

See also [WP_DELETE_MARKER_BEAD@](#)

WP_INSERT_PAGE_BREAK@

Inserts a page break at the current cursor position

Format beadNum = WP_INSERT_PAGE_BREAK@(type)

Method beadNum = this.insert_page_break@(type)

Arguments type The type of page break. Possible values are:

0	next page
1	odd
2	even

Description Inserts a page break at the current cursor position. The bead number of the page break is returned.

If the current position is within a table, the break is placed after the current row.

The current selection is deleted if it is just a break.

See also [WP_INSERT_PAGE_BREAK_NEXT@](#)

WP_INSERT_PAGE_BREAK_NEXT@

Inserts a page break

Format beadNum = WP_INSERT_PAGE_BREAK_NEXT@()

Method beadNum = this.insert_page_break_next@

Description Inserts a page break at the current cursor position. This page break forces text to the *next* page. (Use the [WP_INSERT_PAGE_BREAK@](#) command to insert breaks to odd or even pages.) The bead number of the page break is returned.

If the current position is within a table, the break is placed after the current row.

The current selection is deleted if it is just a break.

WP_INSERT_QUERY_INSET@

Inserts a Data object

Format WP_INSERT_QUERY_INSET@(appMacro, inFrameFlag)

Method [this.insert_query_inset@](#)(appMacro, inFrameFlag)

Arguments appMacro A string indicating the name of the ELF macro that is executed when a user double-clicks on the embedded object field.
inFrameFlag A Boolean value, which if set to TRUE indicates that the embedded object field should be placed within a frame.

Description Inserts an Applixware Data embedded object field in a Words document at the current cursor position.

WP_INSERT_ROWS@

Inserts rows in a table

Format format [wp_range@](#) range = WP_INSERT_ROWS@(format arrayof wp_bead@ beads, format [wp_location@](#) loc, beforeRowFlag)

Method format wp_range@ range = [this.insert_rows@](#)(format arrayof wp_bead@ beads, format wp_location@ loc, beforeRowFlag)

Arguments beads An array of format wp_bead@.
loc The location where rows will be inserted.
beforeRowFlag
If set to TRUE and loc is a row, the row is inserted before the current row. If set to FALSE and loc is a row, the row is inserted after the current row.

Description Insert some rows before or after an existing row. If loc is not in a row, this macro creates a new table. The range of the new beads is returned.

See also [WP_DELETE_ROW@](#)

WP_INSERT_SECTION@

Inserts a section bead

Format beadNum = WP_INSERT_SECTION@(format wp_bead@ bead[, format wp_location@ loc])

Method beadNum = this.insert_section@(format wp_bead@ bead[, format wp_location@ loc])

Arguments bead A section bead.
loc The location at which the section bead will be inserted. If this argument is not used (or is NULL), the section bead is inserted at the cursor's location (or the start of a selection).

Description Inserts a section bead at the current cursor position and returns the bead number of the inserted section bead. In most cases, the section bead you are inserting is derived from the current section's section bead. The section bead may not be inserted at the current cursor position in some cases. For example, if the current cursor position is within a table's cell, the section bead will be inserted after the current row. (To obtain this bead, use **WP GET CURRENT SECTION BEAD@.**)

The portion of wp_bead@ structure that describes a section is as follows:

```
sectType,      ' int: WP#SECTION#CONTINUOUS,
...
nCols,        ' int: # of columns in section
gutterWidth,  ' int: space between snaking col-
               umns
pageWidth,    ' int: width of page, in mils (1000 mils
               = 1 inch)
pageHeight,   ' int: height of page, in mils
pageOrientation, ' int: WP_PAGE_PORTRAIT or
               '_LANDSCAPE
leftMargin,   ' int: left page margin, in mils
rightMargin,  ' int: right page margin, in mils
topMargin,    ' int: top page margin, in mils
bottomMargin, ' int: bottom page margin, in mils
bindingMargin, ' int: addl margin on "binding" side
               (2-sided)
oddHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
evenHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
firstHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
lastHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
oddFooterType, ' int: WP_HDRFTR_NORMAL, etc.
evenFooterType, ' int: WP_HDRFTR_NORMAL, etc.
firstFooterType, ' int: WP_HDRFTR_NORMAL, etc.
lastFooterType, ' int: WP_HDRFTR_NORMAL, etc.
oddHeaderName, ' string: name of hdrftr/glossary
evenHeaderName, ' string: name of hdrftr/glossary
firstHeaderName, ' string: name of hdrftr/glossary
lastHeaderName, ' string: name of hdrftr/glossary
oddFooterName, ' string: name of hdrftr/glossary
```

evenFooterName, ' string: name of hdrftr/glossary
 firstFooterName, ' string: name of hdrftr/glossary
 lastFooterName, ' string: name of hdrftr/glossary
 sectNumStyle, ' int: section numbering style (arabic,
 ...)
 controlSectNum, ' int: section # control (same, restart,
 ..)
 restartSectNum- ' int: number to restart section num-
 Val, bering at
 separator, ' string: section/page number separa-
 tor
 pageNumStyle, ' int: page numbering style (arabic,
 ...)
 restartPageNum, ' int: restart page numbering if TRUE
 restartPageNum- ' int: number to restart page number-
 Val, ing at

WP_INSERT_SECTION_BREAK@

Inserts a section break

Format WP_INSERT_SECTION_BREAK@(type)

Method [this.insert_section_break@\(type\)](#)

Arguments

type	An indicator of the relationship of the current section with the next section. type is one of the following values:
0	Continuous
1	Next column
2	Next Page
3	Odd Page
4	Even Page

Description Inserts a section break at the current cursor position. The section break may not be inserted at the current cursor position in some cases. For example, if the current cursor position is within a table's cell, the section break will be inserted after the current row.

The new section break's attributes are constructed by copying the existing section break to a new place, then updating its type so that page break is of the correct type.

WP_INSERT_SS_INSET@

Inserts a Spreadsheet object

Format WP_INSERT_SS_INSET@(appMacro, inFrameFlag)

Method [this.insert_ss_inset@](#)(appMacro, inFrameFlag)

Arguments appMacro A string indicating the name of the ELF macro that is executed when a user double-clicks on the embedded object field.

inFrameFlag A Boolean value, which if set to TRUE indicates that the embedded object field should be placed within a frame.

Description Inserts a Spreadsheet embedded object field in a Words document at the current cursor position.

WP_INSERT_TABLE@

Inserts a table at the cursor position

Format WP_INSERT_TABLE@(numRows, numCols, colWidth, rowHeightType, rowHeight, leftMargin, rightMargin, topMargin, bottomMargin)

Method [this.insert_table@](#)(numRows, numCols, colWidth, rowHeightType, rowHeight, leftMargin, rightMargin, topMargin, bottomMargin)

Arguments numRows The number of rows in the inserted table.

numCols The number of columns in the inserted table.

colWidth A column's width in mils (1000 mils = 1 inch). If the column width is 0, then the columns' width is determined by Words.

rowHeightType An indicator of how a row's height is determined. Possible values are:

0	Automatically
1	Exactly at
2	At least
3	No more than.

rowHeight For all options of rowHeightType except Automatically, specify the row's height in mils.

leftMargin The cell's left margin in mils.

rightMargin The cell's right margin in mils.

topMargin The cell's top margin in mils.
bottomMargin
The cell's bottom margin in mils.

Description Inserts a table (with empty cells) at the cursor position.

See also [WP_INSERT_TABLE_AT_LOC@](#)

WP_INSERT_TABLE_AT_LOC@

Inserts a table at the passed location

Format format wp_location@ loc = WP_INSERT_TABLE_AT_LOC@(numRows, numCols, colWidth, rowHeightType, rowHeight, leftMargin, rightMargin, topMargin, bottomMargin, format wp_location@ loc))

Method format wp_location@ loc = this.insert_table_at_loc@(numRows, numCols, colWidth, rowHeightType, rowHeight, leftMargin, rightMargin, topMargin, bottomMargin, format wp_location@ loc))

Arguments

numRows The number of rows in the inserted table.

numCols The number of columns in the inserted table.

colWidth A column's width in mils (1000 mils = 1 inch). If the column width is 0, then the columns' width is determined by Words.

rowHeightType
An indicator of how a row's height is determined. Possible values are:

0	Automatically
1	Exactly at
2	At least
3	No more than

rowHeight For all options of rowHeightType except Automatically, specify the row's height in mils.

leftMargin The cell's left margin in mils.

rightMargin The cell's right margin in mils.

topMargin The cell's top margin in mils.

bottomMargin
The cell's bottom margin in mils.

loc The location at which the table is inserted.

Description Inserts a table (with empty cells) at the passed location. The location of the first paragraph in the first cell of the table is returned. It is an error if loc is within a cell. However, if loc is a Row Start bead, the new rows are inserted before the existing row.

See also [WP_INSERT_TABLE@](#)

WP_INSERT_TABLE_CONT_TEXT@

Inserts a table continuation glossary

Format WP_INSERT_TABLE_CONT_TEXT@()

Method [this.insert_table_cont_text@](#)

Description Inserts a table continuation glossary at the current cursor position. This glossary is named TableContinuation@. This glossary allows use to insert continuation text on each page of a table after its first page.

If this glossary does not exist, it is created. When it is created, it will contain the string "(cont.)". However, you can change this glossary's contents in the same way you can change any other glossary.

WP_INSERT_TABSTOP@

Inserts a tabstop

Format WP_INSERT_TABSTOP@(milpos[, tabType[, decimalChar[, leaderStr]]])

Method [this.insert_tabstop@\(milpos\[, tabType\[, decimalChar\[, leaderStr\] \] \]\)](#)

Arguments

milpos	The position in mils (1000 mils = 1 inch) of the tab stop.
tabType	A value indicating the kind of tab to be created. These values are: WP#TAB#LEFT_ WP#TAB#CENTER_ WP#TAB#RIGHT_ WP#TAB#DECIMAL_ The default tabType is WP#TAB#LEFT_.
decimalChar	A one character string indicating the decimal character. The default decimal character is a period (.).
leaderStr	The set of characters making up the leader string.

Description Adds a tab to all paragraphs in the current selection. If a paragraph was inheriting its tabs, all tabs become local. The maximum number of tabstops is defined in wp_.am as WP#MAX#TABS. The current value is 50.

See also [WP_DELETE_TABSTOP@](#)

WP_INSERT_XRF_SRC_FIELD@

Inserts a cross-reference source field

Format WP_INSERT_XRF_SRC_FIELD@(identifier, liveFlag, text)

Method [this.insert_xrf_src_field@](#)(identifier, liveFlag, text)

Arguments

identifier	The name for the cross-reference source that will be used to refer to this source in the cross reference.
liveFlag	A Boolean value which if set to TRUE indicates that a dynamic link exists between the cross-reference source and the cross reference. When a link is live, changes to the text also appear in the cross reference.
text	The text of the cross reference

Description Defines a cross-reference source for which a cross reference will search.

WP_INSTALL_MACROS@

Installs all macros embedded within a Words file

Format WP_INSTALL_MACROS@()

Method [this.install_macros@](#)

Description Installs all macros that are embedded within a Words file. That is, this macro compiles all of the macros that were embedded within the document. If a .elo file is embedded, it will be placed into the ELF runtime environment.

As a security precaution, the wplInstallMacros profile setting must be set to TRUE. If it is FALSE, the macros will not be compiled.

WP_IS_BEAD_IN_CELL@

Indicates if a bead is in a cell

Format flag = WP_IS_BEAD_IN_CELL@(bead)

Method [flag = this.is_bead_in_cell@](#)(bead)

Arguments bead The bead being tested

Description Returns TRUE if bead is within a cell.

See also [WP_GET_CELL@](#)

WP_IS_BEAD_IN_TABLE@

Indicates if a bead is in a table

Format flag = WP_IS_BEAD_IN_TABLE@(bead)

Method flag = this.is_bead_in_table@(bead)

Arguments bead The bead being tested.

Description Returns TRUE if bead is within a table. Note that the Row Start bead is in the table but is not in any particular cell.

WP_IS_CHILD@

Determines if Words is opened as a child

Format flag = WP_IS_CHILD@()

Method flag = this.is_child@

Description Returns TRUE if the current instance of Words was opened as a child linked/embedded object.

WP_IS_CHILD_LINK@

Determines if Words is opened as a child object

Format flag = WP_IS_CHILD_LINK@()

Method flag = this.is_child_link@

Description Returns TRUE if the current instance of Words was opened as a child linked object.

WP_IS_CHILD_OBJECT@

Determines if Words is a child object

Format flag = WP_IS_CHILD_OBJECT@()

Method flag = this.is_child_object@

Description Returns TRUE if the current instance of Words was opened as a child embedded object.

WP_IS_EQN@

Indicates if in equation editing mode

Format flag = WP_IS_EQN@()

Method flag = this.is_eqn@

Description Returns TRUE if Words is in equation editing mode.

WP_IS_FACE_LATIN@

Returns TRUE if the typeface is not Symbol or Zapf Dingbats

Format flag = WP_IS_FACE_LATIN@(fontname)

Method flag = this.is_face_latin@(fontname)

Arguments fontname The name of the font being tested. Symbolic values for these names (found in wp_.am) are as follows:

WP#FONT#TIMES_
WP#FONT#BOOKMAN_
WP#FONT#SCHOOL_
WP#FONT#CHANCERY_
WP#FONT#PALATINO_
WP#FONT#COURIER_
WP#FONT#HELVETICA_
WP#FONT#HELVNARROW_
WP#FONT#AVANT_
WP#FONT#SYMBOL_
WP#FONT#DINGBATS_
WP#FONT#MINCHOU_
WP#FONT#GOTHIC

WP_IS_HELP@

Indicates if in Help mode

Format flag = WP_IS_HELP@()

Method flag = this.is_help@

Description Returns TRUE if the current word processor is running in help mode.

WP_IS_HTML@

Indicates if the current application is the HTML Author

Format flag = WP_IS_HTML@()

Description Returns TRUE if the current application is the Applixware HTML Author.

WP_IS_ME@

Indicates if in Macro Editor mode

Format flag = WP_IS_ME@()

Method flag = this.is_me@

Description Returns TRUE if the current word processor is the macro editor.

WP_IS_WP@

Indicates if using Words

Format flag = WP_IS_WP@()

Method flag = this.is_wp@

Description Returns TRUE if the current word processor is Words; that is, the current document is not a macro editor document and you are not in equation mode.

WP_ITALICS@

Makes text italic

Format WP_ITALICS@()

Method this.italics@

Description Changes the currently selected text to italic. If no text is selected, the italic attribute is applied to whatever you type after invoking the macro.
If the selection is already italicized, WP_ITALICS@ removes the italic attribute from the selection.

See also [WP_BOLD@](#)
[WP_UNDERLINE@](#)

WP_JOIN@

Combines adjacent text blocks

Format WP_JOIN@()

[Method](#) this.join@

Description Combines two adjacent text blocks in the current Words document. The text block in which the cursor resides is joined with the block above it. The areas to be joined must be compatible in format. For example, you can combine text areas from two adjacent columns to form a single column, but you cannot join text in a paragraph format with text in a two-column format. When adjacent text areas are joined, the attributes of the first text area, such as margins and text size, are applied to the joined text.

When Words joins paragraphs, it removes leading and trailing whitespace. It also removes glossaries if they are part of a paragraph's style definition.

WP_JUSTIFY_CENTER@

Center aligns paragraphs

Format WP_JUSTIFY_CENTER@()

[Method](#) this.justify_center@

Description Centers the text of the paragraphs within the current selection between the left and right margins.

See also [WP_JUSTIFY_FULL@](#)
[WP_JUSTIFY_LEFT@](#)
[WP_JUSTIFY_RIGHT@](#)

WP_JUSTIFY_FULL@

Aligns paragraph text against left and right border

Format WP_JUSTIFY_FULL@()

[Method](#) `this.justify_full@`

Description Sets the text of the paragraphs within the current selection so that it is aligned with both the right and left margins.

See also [WP_JUSTIFY_CENTER@](#)
[WP_JUSTIFY_LEFT@](#)
[WP_JUSTIFY_RIGHT@](#)

WP_JUSTIFY_LEFT@

Aligns paragraph text with left margin

Format `WP_JUSTIFY_LEFT@()`

[Method](#) `this.justify_left@`

Description Sets the text of the paragraphs within the current selection so that it is aligned against the left margin. Because the text is aligned to the left, the paragraphs have a ragged right border.

See also [WP_JUSTIFY_CENTER@](#)
[WP_JUSTIFY_FULL@](#)
[WP_JUSTIFY_RIGHT@](#)

WP_JUSTIFY_RIGHT@

Aligns paragraph text with right margin

Format `WP_JUSTIFY_RIGHT@()`

[Method](#) `this.justify_right@`

Description Sets the text of the paragraphs within the current selection so that it is aligned against the right margin. Because the text is aligned to the right, the paragraphs have a ragged left border.

See also [WP_JUSTIFY_CENTER@](#)
[WP_JUSTIFY_FULL@](#)
[WP_JUSTIFY_LEFT@](#)

WP_LEGALIZE_LOCATION@

Converts the end-of-range location into a legal location

Format WP_LEGALIZE_LOCATION@(format [wp_location@](#) loc)

Method [this.legalize_location@](#)(format wp_location@ loc)

Arguments loc The location to be legalized.

Description Insures that a location is legal. This macro is primarily used for converting the end-of-range location into a legal location.

See also [WP_ILLEGALIZE_LOCATION@](#)

WP_LEFT_ARROW_KEY@

Moves left one character

Format WP_LEFT_ARROW_KEY@()

Method [this.left_arrow_key@](#)

Description Moves the cursor one character to the left.

See also [WP_RIGHT_ARROW_KEY@](#)

WP_LINK_EXT_*@

Places a linked object field of a file in a Words document

Format WP_LINK_EXT_*@ (filename, macro[, conversion[, view[, frameFlag]]])

Method [this.link_ext_*@](#)(filename, macro[, conversion[, view[, frameFlag]]])

Arguments

filename	The path name of the file you want to display in the linked object field.
macro	A string indicating the name of the ELF macro that is executed when a user double-clicks on the linked object field.
conversion	An optional conversion program for converting the file specified by filename to a different format (Applixware Words format, for example).

view	An optional range, view, or chart desired. If view is specified, only the view or range indicated will be displayed in the inset. (Views and ranges typically refer to Spreadsheets views and ranges.)
frameFlag	A Boolean value which if set to TRUE indicates that the object field will be framed.

Description Places a linked object field of a given file at the current cursor location. When a user double-clicks on the field, the macro specified by macro is executed. Typically, macro will contain statements for starting the application that created filename and loading the file displayed in the linked object field. When macro is completed, processing returns to the calling macro.

The WP_LINK_EXT_*@ macros include:

WP_LINK_EXT_ASCII_- GRID@	ASCII Spreadsheet file
WP_LINK_EXT_ASCII_- LINES@	ASCII file
WP_LINK_EXT_ASCII_- PARAS@	ASCII file
WP_LINK_EXT_AUDIO@	Audio file
WP_LINK_EXT_AXBITMAP@	Applixware Bitmap File Format
WP_LINK_EXT_CGM@	Computer Graphics Metafile
WP_LINK_EXT_CSV@	Comma Separated Value
WP_LINK_EXT_DIF@	DIF/XDIF File Format
WP_LINK_EXT_DXF@	Document Interchange Format
WP_LINK_EXT_EPSI@	Encapsulated PostScript Interchange Format
WP_LINK_EXT_EQN@	Applixware Equation Editor File Format
WP_LINK_EXT_FAX@	CCITT Group 3 FAX Format
WP_LINK_EXT_GEM@	GEM Image Format
WP_LINK_EXT_GIF@	Compuserve Graphics Interchange Format
WP_LINK_EXT_GP4@	CCITT Group 4 FAX format.
WP_LINK_EXT_GR@	Applixware Graphics File Format
WP_LINK_EXT_HPGL@	Hewlett-Packard Graphics Language
WP_LINK_EXT_HTML@	HTML Format
WP_LINK_EXT_ILBM@	Amiga IFF ILBM
WP_LINK_EXT_IRIS@	Silicon Graphics IRIS format
WP_LINK_EXT_MAC_- PAINT@	MACPaint file format

WP_LINK_EXT_ME@	Applixware Macro Editor file format
WP_LINK_EXT_MISC@	Miscellaneous
WP_LINK_EXT_MSWBMP@	Microsoft Windows Bitmap
WP_LINK_EXT_MS_WORD@	Microsoft Word file format
WP_LINK_EXT_MSWORD70@	Microsoft Word 7.0 for Windows and Microsoft Word 6.0 for Macintosh File format
WP_LINK_EXT_PBM@	X11 release's PBM (Portable Bitmap) Plus
WP_LINK_EXT_PCX@	PC Paintbrush
WP_LINK_EXT_PGM@	X11 release's PBM (Portable Greymap) Plus
WP_LINK_EXT_PICT2@	Mac PICT2 file
WP_LINK_EXT_PICT@	Mac PICT file
WP_LINK_EXT_PPM@	X11 release's PBM (Portable Pixmap) Plus
WP_LINK_EXT_QUERY@	Applixware Data Format
WP_LINK_EXT_RAW@	Raw graphics
WP_LINK_EXT_RTF@	Microsoft Rich Text Format
WP_LINK_EXT_SS@	Applixware Spreadsheet file
WP_LINK_EXT_SSGRID@	Applixware Spreadsheet grid file
WP_LINK_EXT_SUN_- RASTER@	SUN Raster file
WP_LINK_EXT_SYLK@	Symbolic Link File (Excel, Multiplan)
WP_LINK_EXT_TGA@	TrueVision Targa File
WP_LINK_EXT_TIFF@	Tagged Image File
WP_LINK_EXT_WFW2@	5. 0
WP_LINK_EXT_WK3@	Lotus 1-2-3 Spreadsheet File Format - wk3
WP_LINK_EXT_- WINWORD60@	Microsoft Word for Windows, Version 6
WP_LINK_EXT_WK4@	Lotus 1-2-3 Spreadsheet File Format - wk4
WP_LINK_EXT_WKS@	Lotus 1-2-3 Spreadsheet File Format - wk1
WP_LINK_EXT_WMF@	Windows Metafile
WP_LINK_EXT_WORD- PERFECT@	WordPerfect File Format
WP_LINK_EXT_WP@	Applixware Words document
WP_LINK_EXT_WPG@	Word Perfect Graphic
WP_LINK_EXT_XBM@	X Windows Bitmap

WP_LINK_EXT_XLS@	Microsoft Excel Spreadsheet File Format
WP_LINK_EXT_XPM@	X pixmap
WP_LINK_EXT_XWD@	X Window Dump

See also [WP_CHANGE_LINK@](#)
[WP_CHANGE_LINK_INFO@](#)
[WP_GET_LINKS@](#)
[WP_INSERT_LINK_FILE_FIELD@](#)
[WP_SET_LINKS_INFO@](#)

WP_LINK_ADD_DOC@

Adds a document to the Words history list

Format WP_LINK_ADD_DOC@(filename, target, doc_type, read_only)

Arguments

filename	The full pathname or URL of the document
target	The target within the document
doc_type	The Applixware document type
read_only	TRUE if the document is to be opened as read-only

Description Adds a document to the Words history list. You can display the entire Words history list by running the macro WP_LINK_HISTORY_DLG@.

See also [WP_GOTO_NEXT_DOC@](#)
[WP_GOTO_PREV_DOC@](#)

WP_LIST_GLOSSARY@

Returns an array of glossary names

Format nameArray = WP_LIST_GLOSSARY@()

Method nameArray = this.list_glossary@

Description Returns an array of glossary names; that is, each element in the array contains the name of one glossary element.

WP_LIST_SERIES@

Returns all series information

Format format arrayof wp_series@ =WP_LIST_SERIES@()

Method format arrayof wp_series@ series = this.list_series@

Description Returns an array of series information. Each element in the array is formatted as a wp_series@ variable. The definition of this format is:

```
format wp_series@
    name,          ' name of series
    style,         ' array of 1-char str
    multiLevel,   ' if true, show multiple levels
    showFrom,     ' array, but 1st element not used.
    seperator,    ' array, but 1st element not used.
    leaderText,
    trailerText
```

WP_LOAD_FILE@

Loads a file in the current Words window

Format WP_LOAD_FILE@(filename[,changeBarModeFlag][, noHookFlag][, readOnly][, trueName])

Method this.load_file@(filename[,changeBarModeFlag][, noHookFlag][, readOnly][, trueName])

Arguments

filename	The name of the Words file you want to load. If the file is not in the current directory, filename must be a full path name.
changeBarModeFlag	This flag is obsolete and ignored. Set this to FALSE.
noHookFlag	A Boolean value which if set to TRUE indicates that hook macros are not executed.
readOnly	If TRUE, the file is opened in read-only mode. If FALSE, ELF attempts to open file in read-write mode.
trueName	If <i>name</i> is a URL, or a file in a foreign file format, this is the name of the original file. This is needed to resolve relative links. For example, if you load a file called http://www.Applixware.com, and this file has a graphic, the macro uses trueName to retrieve the linked graphic. When you are loading a local Words file, this value can be NULL.

Description Loads the specified file into the current Words window. The current document is not saved before it is replaced, so any changes made to the document since it was last saved are lost.

When this macro executes, it runs the ELF window hook with a value of 1 unless the value of the noHookFlag parameter is TRUE.

Immediately after executing this macro, you should also execute **WP_INITIALIZE_DOC@** if your application will need to use dictionaries.

WP_LOCAL_TEXT_ATTRS_FOR_ARRAY@

Applies local text attributes

Format format wp_text_attrs@ newBeads =
WP_LOCAL_TEXT_ATTRS_FOR_ARRAY@(format arrayof
wp_bead@ beads, format wp_text_attrs@ attrs)

Method format wp_text_attrs@ newBeads = this.local_text_attrs_for_array@(format
arrayof wp_bead@ beads, format wp_text_attrs@ attrs)

Arguments beads The beads upon which the text attributes will be applied. For a definition of wp_bead@, consult the wp_.am file.
attrs The attributes being applied to an array of text beads.

Description Imposes local text attributes upon an array of text beads. That is, for every text bead in the beads array, if the text bead is inheriting a certain text attribute and the passed text attrs structure has a nonNULL value for this attribute, it is made local to the text bead.

The updated array of beads is returned.

The **WP_GET_TYPING_TEXT_ATTRS@** macro returns a wp_text_attrs@ format containing the typing text attributes for the current selection.

The definition of the returned wp_text_attrs@ format is as follows:

```
format wp_text_attrs@
    bold,          'Boolean
    italic,        'Boolean
    strikethru,    'Boolean
    hidden,        'Boolean
    capitals,      'WP#CAPS#NORMAIL_
                  'WP#CAPS#UPPER_
                  'WP#CAPS#LOWER_
    underline,    'WP#UNDER#NONE_
                  'WP#UNDER#SINGLE_
                  'WP#UNDER#DOUBLE_
                  'WP#UNDER#WORDS_
                  'WP#UNDER#DOUBLE#WORDS
```

	'WP#UNDER#NOTABS
	'WP#UNDER#DOUBLE#NOTABS
position,	'WP#POS#NORMAL#POINTS
	'WP#POS#SUB#POINTS_
	'WP#POS#SUPER#POINTS_
hyphenate,	'Boolean
color,	'color name
face,	'font name
size,	'point size
tag	'tag name

WP_LOCALIZE_ALL_LINKS@

Localizes all linked files

Format WP_LOCALIZE_ALL_LINKS@(localizeInOrigFormat[, appMacro])

Method [this.localize_all_links@](#)(localizeInOrigFormat[, appMacro])

Arguments localizeInOrigFormat

A Boolean value which if set to TRUE leaves the linked object fields in its native format. If set to FALSE, links to foreign applications are converted to Applixware format before being localized.

appMacro If an application macro is specified, the localized embedded object will call this macro to open the object.

Description Copies an externally linked file into the current Words document. The linked file is changed into an embedded local object. If localizeInOrigFormat is set to FALSE, this macro also converts the linked file into an Applixware formatted file.

See also [WP_CHANGE_LINK@](#)

[WP_CHANGE_LINK_INFO@](#)

[WP_GET_LINKS@](#)

[WP_INSERT_LINK_FILE_FIELD@](#)

[WP_LOCALIZE_LINK@](#)

[WP_SET_LINKS_INFO@](#)

WP_LOCALIZE_LINK@

Embeds a linked object

Format WP_LOCALIZE_LINK@(index, localizeInOrigFormat[, appMacro])

Method [this.localize_link@](#)(index, localizeInOrigFormat[, appMacro])

- Arguments**
- index The index of the link to be localized in a list of links. (this list is usually returned using WP_GET_LINKS_INFO@.)
 - localizeInOrigFormat A Boolean value which if set to TRUE leaves the embedded object in its native format. If set to FALSE, links to foreign applications are converted to Applixware format before being localized.
 - appMacro If an application macro is specified, the localized fields will call this macro to open the embedded object.

Description Copies an externally linked file into the current Words document. The linked file is changed into an embedded object.

The arrayOffset is the index of the link to be localized in the list of links. This list is created using the [WP_GET_LINKS_INFO@](#) macro. Each element in this array is formatted as a wp_link_info@ variable. The internal link list is dynamic. After a link is localized, the list is adjusted to reflect the new link status. This means that the following code fragment will not work:

```
links_list = WP_GET_LINKS_INFO@()
for i = 0 to ARRAY_SIZE@(links_list)-1
    'Incorrect after first localize
    WP_LOCALIZE_LINK@(i, FALSE)
next i
```

If all links need to be localized, use [WP_LOCALIZE_ALL_LINKS@](#).

If localizeInOrigFormat is set to FALSE, this macro also converts the linked file into an Applixware formatted file.

- See also** [WP_CHANGE_LINK@](#)
[WP_CHANGE_LINK_INFO@](#)
[WP_GET_LINKS@](#)
[WP_INSERT_LINK_FILE_FIELD@](#)
[WP_LOCALIZE_ALL_LINKS@](#)
[WP_SET_LINKS_INFO@](#)

WP_LOCATION_IN_RANGE@

Determines if the location is in a range

Format flag = WP_LOCATION_IN_RANGE@(format [wp_location@](#) loc, format [wp_range@](#) range)

Method flag = this.location_in_range@(format wp_location@ loc, format wp_range@ range)

Arguments loc An arbitrary location.

range An arbitrary range.

Description Returns TRUE if the passed location is within the passed range; otherwise it returns FALSE.

WP_LOCATION_OF_CURSOR@

Returns the location of the cursor

Format format wp_location@ loc = WP_LOCATION_OF_CURSOR@()

Method format wp_location@ loc = this.location_of_cursor@

Description Returns the location of the cursor (or the start of the selection). This returned value is in [wp_location@](#) format.

See also [WP_DISPLAY_LOC_OF_CURSOR@](#)

WP_MAIL@

Mails the current Words document

Format WP_MAIL@()

Method this.mail@

Description Mails the current Words document. If there are linked objects, these linked objects are also sent.

WP_MAKE_PASSED_ROW_FIT_STD@

Resizes row to fit within margin

Format WP_MAKE_PASSED_ROW_FIT_STD@(bead)

Method this.make_passed_row_fit_std@(bead)

Arguments bead A bead in the current row.

Description Scales all the cells in the row containing bead to fit within the current margins. This function is the same as the following function:

[WP_MAKE_ROWS_FIT@](#)(bead, 0 , 0, 0, FALSE, TRUE, TRUE)

WP_MAKE_ROW_FIT@

Resizes a row within margin

Format flag = WP_MAKE_ROW_FIT@(row, firstCell, nResize, resizeType, resizeMargins, expand, useAllSpace)

Method [this.make_row_fit@](#)(row, firstCell, nResize, resizeType, resizeMargins, expand, useAllSpace)

Arguments

row	The bead in the row to resize.
firstCell	The column number (zero-based) of the first cell being resized.
nResize	Number of cells to resize. 0 means that all cells will be resized. It is not an error if the current row has fewer cells than this number.
resizeType	The way in which cells should be resized. This can be one of the following values: 0 WP_MRF_CONSTANT_SCALE Apply constant scale to cells 1 WP_MRF_SAME_SIZE Make cells the same size
resizeMargins	A Boolean value which if set to TRUE tells that macro to resize the cell row if there is room. If set to FALSE, just the width is changed.
expand	A Boolean value which if set to TRUE indicates that the macro can expand the row if there is row.
useAllSpace	A Boolean value which if set to TRUE indicates that the rows bounding rectangle should be fully filled.

Description Makes a row fit into the current column width. The cells in the row are scaled evenly (up or down) so that the row fits exactly. The scaling affects the margins and the material area in the cells. You can request that all space in the column be used, or only as much space as is available *after* frames are taken into account.

TRUE is returned if the request was carried out as desired. FALSE is returned if there was some problem. For example, if the indicated cells cannot be shrunk sufficiently to make the row fit, ELF does the best it can, but the row will still be too large.

See also [WP_MAKE_ROW_FIT_QUICK@](#).

WP_MAKE_ROW_FIT_QUICK@

Repeats the previous WP_MAKE_ROW_FIT@

Format WP_MAKE_ROW_FIT_QUICK@()

Method [this.make_row_fit_quick@](#)

Description Repeats the previous [WP_MAKE_ROW_FIT@](#) operation.

WP_MAKE_ROW_FIT_SHRINK@

Shrinks the selection to fit

Format flag = WP_MAKE_ROW_FIT_SHRINK@([noErrFlag])

noErrFlag A Boolean value which if set to TRUE indicates that if the selection does not start in a row and does not contain a row, do not post an error. The default is FALSE; that is, an error is posted.

Method [flag = this.make_row_fit_shrink@\(\[noErrFlag \]\)](#)

Description Shrinks the selection to fit; no expansion will occur. All cells in a row will be resized by the same scale factor. Margins may also be resized.

This is a convenience function that calls [WP_MAKE_ROWS_FIT@](#)

WP_MAKE_ROWS_FIT@

Resizes selected rows to fit

Format WP_MAKE_ROWS_FIT@(whichCells, resizeType, resizeMarginsFlag, expandFlag, useAllSpaceFlag[, noErrFlag])

Method [this.make_rows_fit@\(whichCells, resizeType, resizeMarginsFlag, expandFlag, useAllSpaceFlag\[, noErrFlag \]\)](#)

Arguments which_cells An indicator of which cells should be resized. This can be one of the following values:

- 0 WP_MRF_RESIZE_SELECTED
Resize only the selected cells
- 1 WP_MRF_RESIZE_ALL
Resize all cells in the row

resizeType The way in which cells should be resized. This can be one of the following values:

- 0 WP_MRF_CONSTANT_SCALE
Apply constant scale to cells
- 1 WP_MRF_SAME_SIZE
Make cells the same size

resizeMarginsFlag

A Boolean value which if set to TRUE tells that macro to resize the cell row if there is room. If set to FALSE, just the width is changed.

expandFlag

A Boolean value which if set to TRUE indicates that the macro can expand the row if there is row.

useAllSpaceFlag

A Boolean value which if set to TRUE indicates that the rows bounding rectangle should be fully filled.

noErrFlag

A Boolean value which if set to TRUE indicates that if the selection does not start in a row and does not contain a row, do not post an error. The default is FALSE; that is, an error is posted.

Description Resizes the rows in the current selection. Each row in the selection is individually sent to **WP_MAKE_ROWS_FIT@**.

WP_MARK_POINT@

Marks a point in an Applixware Words document

Format WP_MARK_POINT@()

Method [this.mark_point@](#)

Description Marks a point in a words document. If you then move to a different place in the document, a second call to this macro selects the region between the two points.

WP_MENU_STATUS_CALLBACK@

Defines a callback macro invoked when a Words menu is displayed

Format WP_MENU_STATUS_CALLBACK@(callback)

Method [this.menu_status_callback@\(callback\)](#)

Arguments callback The name of the callback macro.

Description Names the callback macro that will be invoked when a user-defined function is placed on the menu bar. The first argument to the callback macro is the name of the menu item.

The callback macro should return one of the menu status defined in `menubar_.am` or -1 if it will not control the status of the menu item.

The most commonly used entries are:

MENUSTAT#NORMAL Display normally
MENUSTAT#DIMMED Display grayed
MENUSTAT#TOGGLE_ON Toggle on to left
MENUSTAT#TOGGLE_OFF Is a toggle, but is off
MENUSTAT#RADIO_OFF Is a radio, currently off
MENUSTAT#RADIO_ON Radio on to left
MENUSTAT#NO_SHOW Suppress display

Here is an extract from the callback macro used with the Help software:

```
function HELP_EXPRESSLINE_CALLBACK$(funcName)
    var format HELPdomainData@ currentData
    var format HELPlinks@ historyList

    case of (funcName)
    case "HELP_CONTENTS@"
        currentData = system_var@("HELFCurrentDomainData$")
        if (is_null@(currentData.contentsFilename))
            return (MENUSTAT#DIMMED)
        else
            return (MENUSTAT#NORMAL)

    case "HELP_BACK@"
    case "HELP_HISTORY_DLG@"
        historyList = system_var@(HELPhistoryList$)
        if (is_null@(historyList))
            return (MENUSTAT#DIMMED)
        if (array_size@(historyList) = 1)
            return (MENUSTAT#DIMMED)
        return (MENUSTAT#NORMAL)

    ...
    default
        return (-1)
    endcase
endmacro
```

WP_MERGE_CELLS@

Merges cell

Format WP_MERGE_CELLS@(format [wp_range@](#) range, firstFlag)

Method [this.merge_cells@](#)(format [wp_range@](#) range, firstFlag)

Arguments range The cells being merged.
firstFlag A Boolean value which if set to TRUE merges cells into the first cell.
If set to FALSE, it preserves the total width of all cells.

Description Merges cells in a passed range. This range must have a set of adjacent cells, beginning at the first bead of the first cell and ending with the Cell End bead of the last cell. Passing a range containing only one cell is not an error; however, nothing will occur. This macro does *not* alter the current selection.

See also [WP_MERGE_CELLS_FIRST@](#)
[WP_MERGE_CELLS_WIDE@](#)

WP_MERGE_CELLS_FIRST@

Merges cells into the first cell

Format WP_MERGE_CELLS_FIRST@()

Method [this.merge_cells_first@](#)

Description Merges all selected cells in a row into the first selected cell of the row. If more than one row is in the selection, each row is merged separately.

See also [WP_MERGE_CELLS@](#)
[WP_MERGE_CELLS_WIDE@](#)

WP_MERGE_CELLS_WIDE@

Merges cells into a wide cell

Format WP_MERGE_CELLS_WIDE@()

Method [this.merge_cells_wide@](#)

Description Merges all selected cells in a row into a single cell. The width of the combined cell is the width of the cells being merged. That is, if you are merging three 1" cells, the merged cell will be 3" wide.

See also [WP_MERGE_CELLS@](#)
[WP_MERGE_CELLS_FIRST@](#)

WP_MERGE_INIT@

Establishes merge information

Format WP_MERGE_INIT@(filename, nFields, rowStart)

Method [this.merge_init@\(filename, nFields, rowStart\)](#)

Arguments filename The file containing merge information.
nFields The number of merge fields in the document. This value is returned.
rowStart The Row Start bead of the first bead of the first record. This value is returned.

Description Loads a merge data file into the merge glossary and performs an analysis of the merge fields in the document. After analyzing the information, it returns the Row Start bead number of the first non-header record in this table and the number of fields in the document.

The merge data should be in a table. The first row of the table should contain the data field names. These names must match the names of the Merge Fields in the documents. All other rows contain the data being merged.

See also [WP_MERGE_NUM_RECORDS@](#)
[WP_MERGE_RECORD@](#)
[WP_MERGE_RECORD_NUM@](#)

WP_MERGE_NUM_RECORDS@

Returns number of merge records

Format numRec = WP_MERGE_NUM_RECORDS@()

Method numRec = this.merge_num_records@

Description Returns the number of records in the Merge Data table. This number excludes the header (field name) row of the table. If 0 is returned, then no Merge Data table exists.

See also [WP_MERGE_INIT@](#)
[WP_MERGE_RECORD@](#)
[WP_MERGE_RECORD_NUM@](#)

WP_MERGE_RECORD@

Merges one record into document

Format beadnum = WP_MERGE_RECORD@(nfields, rowStart)

Method beadnum = this.merge_record@(nfields, rowStart)

Arguments nfields The number of merge fields in the document.
rowStart The Row Start bead of the current record.

Description Merges the current Merge record into the document. This operation can merge more than one record into the document if the Merge fields indicate record offsets or advances.

The Row Start bead number of the next record is returned. If no more records are found in the merge table, this macro returns 0 (zero).

See also [WP_MERGE_INIT@](#)
[WP_MERGE_NUM_RECORDS@](#)
[WP_MERGE_RECORD_NUM@](#)

WP_MERGE_RECORD_NUM@

Merges record into a document

Format flag = WP_MERGE_RECORD_NUM@(nFields, recNum)

Method flag = this.merge_record_num@(nFields, recNum)

Arguments nFields The number of merge fields in the document
recNum The record number to merge. This number is one-based rather than zero-based.

Description Merges a particular merge record into the document. This macro could actually merge several records into the document if the merge fields within the document indicate record offsets or advances.

The record number of the current record is input. The first record is number 1; this corresponds to the second row of the Merge Data table since the first row contains the field names.

This macro returns TRUE if the record was merge. FALSE indicates that the record number is too big or there is no Merge Data table.

See also [WP_MERGE_INIT@](#)
[WP_MERGE_NUM_RECORDS@](#)
[WP_MERGE_RECORD@](#)

WP_MERGE_ROW@

Merges two table rows

Format WP_MERGE_ROW@(bead)

Method this.merge_row@(bead)

Arguments bead The passed bead.

Description Merges the row containing the passed bead with the following row (if any) by removing the following row's Row Start bead.

WP_MERGE_SELECTED_CELLS@

Merges selected cells

Format WP_MERGE_SELECTED_CELLS@(firstFlag)

Method [this.merge_selected_cells@\(firstFlag\)](#)

Arguments firstFlag If set to TRUE, cells are merged into the first cell. If set to FALSE, the total width of all cells is preserved.

Description Merges selected cells.

WP_MERGE_SELECTED_ROWS@

Merges selected rows

Format WP_MERGE_SELECTED_ROWS@()

Method [this.merge_selected_rows@](#)

Description Merges selected rows into one row. If the selection contains more than one table, each table's selection is merged separately.

WP_MERGE_TERMINATE@

Performs clean-up operations after a mail-merge

Format WP_MERGE_TERMINATE@(nfields)

Method [this.merge_terminate@\(nfields\)](#)

Arguments nfields The number of merge fields in the document.

Description Removes special markers and glossaries created by the mail merge operation.

WP_MODIFIED@

Indicates if a document is in a modified state

Format flag = WP_MODIFIED@()

Method flag = this.modified@

Description Returns TRUE if the current Words document has been modified since it was last saved. It returns FALSE if no modifications have been made. If the document is a new document that has never been saved to a file, TRUE is returned if any editing occurred in the Words window.

See also [WP_SET_MODIFIED@](#)

WP_MODIFY_BORDER_THICKNESS@

Changes the border's thickness

Format WP_MODIFY_BORDER_THICKNESS@(thickness)

Method this.modify_border_thickness@(thickness)

Arguments thickness A number in mils (1000 mils = 1 inch) specifying a rule line's thickness.

Description Changes the thickness of all selected borders to thickness. If a border does not exist, it is not set. That is, this macro only changes existing borders in the selection; it does not create them.

See also [WP_BORDER@](#)

WP_MODIFY_COLOR@

Changes a color's definition

Format WP_MODIFY_COLOR@(colorName, cyan, magenta, yellow, black)

Method this.modify_color@(colorName, cyan, magenta, yellow, black)

Arguments colorName A color's name.
cyan colorName's new cyan component.
magenta colorName's new magenta component.
yellow colorName's new yellow component.
black colorName's new black component.

Description Modifies the definition of a color that is associated with colorName. This color is expressed in the CMYK color system.

See also [WP_GET_COLOR@](#)

WP_MOVE_BEADS@

Moves a set of beads within the document

Format format wp_range@ range = WP_MOVE_BEADS@(format [wp_range@](#) range, format [wp_location@](#) loc)

Method format wp_range@ range = this.move_beads@(format wp_range@ range, format wp_location@ loc)

Arguments range The set of beads to move.
loc The location to which the beads will be moved.

Description Moves a set of beads within the document. This function returns the output range.

NOTE: Moving the wrong beads to the wrong location will cause severe problems.

WP_MOVE_TABSTOP@

Changes a tabstop's position

Format WP_MOVE_TABSTOP@(fromPos, toPos)

Method this.move_tabstop@(fromPos, toPos)

Arguments fromPos The old tabstop's position expressed in mils. (1000 mils = 1 inch)
toPos The new tabstop's position expressed in mils.

Description Resets the tabstops that exists as fromPos so that it is now at toPos.

WP_NEW@

Creates a new Words document

Format WP_NEW@()

Method this.new@

Description Creates a new Words document. This new document will have a name chosen by Words. In addition, the style used for this document will be the same as that used by the document that existed in the Words window immediately before

WP_NEW@ executed. WP_NEW@ forces a legal language and finds the default custom dictionary and also forces the next Save to be treated as a Save As.

If a file hook is set, it is run with a value of 1.

WP_NEW_WINDOW@

Opens a new Words window

Format pid = WP_NEW_WINDOW@(id, quiet, readOnly, width, height[, windowTitle[, iconTitle[, iconID]]])

Method pid = this.new_window@(id, quiet, readOnly, width, height[, windowTitle[, iconTitle[, iconID]]])

Arguments

id	The menu id, use the standard Words menus or a custom menu.
quiet	If set to TRUE, the new window runs in <i>windowless</i> mode. See the applix -quiet command line option. If set to FALSE, the window runs normally.
readOnly	If set to TRUE, the new window runs in Read-Only mode. If set to FALSE, the window runs normally.
width	A non-zero value (in mils, which are set 1000 to the inch) that overrides the standard window width.
height	A non-zero value (in mils, which are set 1000 to the inch) that overrides the standard window height.
windowTitle	An optional window title. This title will not be changed when new files are read into the window.
iconTitle	An optional title to be used when the window is minimized.
iconID	An optional icon id; this icon will be used when the window is minimized.

Description Opens a new Words window.

If a file hook is set, it is run with a value of 1.

The task number for the new window is returned.

WP_NEW_EQN_PICTURE@

Adds a new Equation editor inset to an Applixware Words document

Format WP_NEW_EQN_PICTURE@()

Description Adds a new Equation editor inset to an Applixware Words document. The object is inserted at the location of the cursor.

WP_NEW_GR@

Adds a new Graphics inset to an Applixware Words document

Format WP_NEW_GR@()

Description Adds a new Graphics inset to an Applixware Words document. The object is inserted at the location of the cursor.

WP_NEW_GR_FRAMED@

Adds a new framed Graphics inset to an Applixware Words document

Format WP_NEW_GR_FRAMED@()

Description Adds a new framed Graphics inset to an Applixware Words document. The object is inserted at the location of the cursor.

WP_NEW_INSET@

Adds a new inset to an Applixware Words document

Format WP_NEW_INSET@(doctype, cvtmacro, appmacro, picture, frame)

Arguments

docType	A constant indicating the type of inset to embed. These constants are defined in the ELF include file app_ids.am .
cvtMacro	A string name of the macro to run to convert the new inset into an Applixware document type. This is required if the new inset is not an Applixware document type. For example, if the new inset is a Microsoft Word inset, the cvtMacro field should be set to "WP_IMPORT_MSWORD@." This field should be set to NULL if you are running Applixware 4.3 or later, since this information is read out of the hooks.dat file.
appMacro	A string name of a customized macro to run to launch the foreign application when the user double-clicks the inset. This field should be set to NULL if you are running Applixware 4.3 or later, since this information is read out of the hooks.dat file.
picture	TRUE (-1) if the new inset is to be displayed as a picture. FALSE (0) if the inset is to be embedded as text. Only Applixware Words and Spreadsheets insets can be displayed as pictures.
frame	TRUE (-1) if the new inset is to be displayed in a frame. FALSE (0) if the inset is to be displayed without a frame.

Description Adds a new inset to an Applixware Words document. The inset can be in the either an Applixware document, or a foreign document type.

WP_NEW_ME_PICTURE@

Add a macro editor inset to an Applixware Words document

Format WP_NEW_ME_PICTURE@()

Description Adds a new macro editor inset to an Applixware Words document, and opens the macro editor so that you can add information to the inset. The inset is inserted into the document at the location of the cursor. Macro Editor insets are embedded as pictures in Words.

WP_NEW_QUERY@

Adds an Applixware Data query inset to an Applixware Words document

Format WP_NEW_QUERY@()

Description Adds an Applixware Data query inset to an Applixware Words document, and opens Applixware Data. The inset is inserted into the Words document at the location of the cursor.

WP_NEW_QUERY_FRAMED@

Adds a framed Applixware Data query inset to an Applixware Words document

Format WP_NEW_QUERY_FRAMED@()

Description Adds an Applixware Data query inset to an Applixware Words document, and opens Applixware Data. The inset is inserted into the Words document at the location of the cursor.

WP_NEW_SS@

Adds an Applixware Spreadsheets inset to an Applixware Words document

Format WP_NEW_SS@()

Description Adds an Applixware Spreadsheets inset to an Applixware Words document, and opens Applixware Applixware Spreadsheets. The inset is inserted into the Words document at the location of the cursor. The Spreadsheets inset is displayed as text.

WP_NEW_SS_FRAMED@

Adds a framed Applixware Spreadsheets inset to an Applixware Words document

Format WP_NEW_SS_FRAMED@()

Description Adds a framed Applixware Spreadsheets inset to an Applixware Words document, and opens Applixware Applixware Spreadsheets. The frame containing the inset is inserted into the Words document at the location of the cursor. The Spreadsheets inset is displayed as text.

WP_NEW_SS_FRAMED_PICTURE@

Adds a framed Applixware Spreadsheets inset to an Applixware Words document

Format WP_NEW_SS_FRAMED_PICTURE@()

Description Adds a framed Applixware Spreadsheets inset to an Applixware Words document, and opens Applixware Spreadsheets. The frame containing the inset is inserted into the Words document at the location of the cursor. The Spreadsheets inset is displayed as a picture.

WP_NEW_SS_PICTURE@

Adds an Applixware Spreadsheets inset to an Applixware Words document

Format WP_NEW_SS_PICTURE@()

Description Adds a framed Applixware Spreadsheets inset to an Applixware Words document, and opens Applixware Spreadsheets. The frame containing the inset is inserted into the Words document at the location of the cursor. The Spreadsheets inset is displayed as a picture.

WP_NEW_WP@

Adds an Applixware Words inset to an Applixware Words document

Format WP_NEW_WP@()

Description Adds an Applixware Words inset to an Applixware Words document, and opens Applixware Words. The inset is inserted into the Words container document at the location of the cursor. The Words inset is displayed as text.

WP_NEW_WP_FRAMED@

Adds a framed Applixware Words inset to an Applixware Words document

Format WP_NEW_WP_FRAMED@()

Description Adds a framed Applixware Words inset to an Applixware Words document, and opens Applixware Words. The frame containing the inset is inserted into the Words container document at the location of the cursor. The inset is displayed as text.

WP_NEW_WP_FRAMED_PICTURE@

Adds a framed Applixware Words inset to an Applixware Words document

Format WP_NEW_WP_FRAMED_PICTURE@()

Description Adds a framed Applixware Words inset to an Applixware Words document, and opens Applixware Words. The frame containing the inset is inserted into the Words document at the location of the cursor. The inset is displayed as a picture.

WP_NEW_WP_PICTURE@

Adds an Applixware Words inset to an Applixware Words document

Format WP_NEW_WP_PICTURE@()

Description Adds an Applixware Words inset to an Applixware Words document, and opens Applixware Words. The inset is displayed as a picture.

WP_NEXT_FIELD_GLOSS@

Replaces one glossary reference with another

Format format wp_bead@ beads = WP_NEXT_FIELD_GLOSS@(inNames, inFS)

Method format wp_bead@ beads = this.next_field_gloss@(inNames, inFS)

Arguments inNames Set of glossaries to alternate among.
inFS Field Start bead number.

Description Replaces a glossary reference field in a field value with the next glossary in a glossary list.

This macro expects to be called by a Macro Beads field in Words. It should be passed an array of glossary names. It will then examine the value in the field (which will hopefully be one of the glossary names in the array), and return a set of beads representing the next glossary.

If the Field Value is empty or contains unexpected material, the first glossary name in the array is returned.

WP_NEXT_FIELD_STRING@

Replaces one Field Value string with another

Format format wp_bead@ beads = WP_NEXT_FIELD_STRING@(inNames, inFS)

Method format wp_bead@ beads = this.next_field_string@(inNames, inFS)

Arguments inNames Set of strings to alternate among.
inFS Field Start bead number.

Description Replaces a string in a field value with the next string in a string list.

This macro expects to be called by a Macro String field in Words. It should be passed an array of strings. It will then examine the value in the field (which will hopefully be one of the strings in the array), and return a set of beads representing the next string.

If the Field Value is empty or contains unexpected material, the first glossary name in the array is returned.

WP_NEXT_KEY@

Scrolls to next screen

Format WP_NEXT_KEY@()

Method [this.next_key@](#)

Description Scrolls to the next screen. The cursor's position does not change.

See also [WP_PREV_KEY@](#)

WP_NEXT_PAGE@

Moves forward to the specified page

Format WP_NEXT_PAGE@(number)

Method [this.next_page@\(number\)](#)

Arguments number The page number to which the cursor will move.

Description Moves the cursor forward to the specified page. If number is less than or equal to the current page number, no action (exception a BEEP@) occurs.

Because of this macro's limitations, you should use [WP_GOTO_PAGE@](#).

Note This macro may be eliminated in a future release.

WP_NEXT_VISIBLE_BEAD@

Returns the bead number of the next visible bead

Format nextNum = WP_NEXT_VISIBLE_BEAD@(beadNum)

Method [nextNum = this.next_visible_bead@\(beadNum\)](#)

Arguments beadNum The sequence number of the bead to start searching at.

Description Returns the bead number of the next visible bead in the current flow. Visibility can depend on field method/value.

0 is returned if the end of the current flow is found before the macro finds a visible bead.

See also [WP_GET_BEAD@](#)

WP_NUM_CELLS_IN_ROW@

Returns the number of cells in a row

Format num = WP_NUM_CELLS_IN_ROW@(bead)

Method num = this.num_cells_in_row@(bead)

Arguments bead The passed bead.

Description Returns the number of cells in a row in which the passed bead resides. 0 (zero) is returned if the passed bead is not in a row.

WP_NUM_COLS_SELECTED@

Returns the number of selected table columns

Format num = WP_NUM_COLS_SELECTED@()

Method num = this.num_cols_selected@

Description Returns the number of table columns selected. A non-zero result is returned only when the current selection falls entirely within a single table *and* if the selection has more than one cell. In some cases (where rows have different numbers of cells) the number of selected columns cannot be deduced from the selected range, so it is stored explicitly.

A non-zero result means that not all beads that fall within the selection range are considered to be selected. 0 (zero) indicates a simple selection, where all beads in the range are selected.

WP_NUM_ROWS_IN_TABLE@

Returns the number of rows in a table

Format numRows = WP_NUM_ROWS_IN_TABLE@ (beadNum)

Method numRows = this.num_rows_in_table@(beadNum)

Arguments beadNum The number of a bead located anywhere in the table.

Description Returns the number of rows in the table containing bead. 0 (zero) is returned if beadNum is not within a table.

WP_NUM_TAGS@

Returns the number of tags

Format numTags = WP_NUM_TAGS@()

Method numTags = this.num_tags@

Description Returns the number of defined tags within the current document.

WP_NUMBERING@

Adds or removes numbers from the front of a paragraph

Format WP_NUMBERING@(format wp_bead@ newNum, starting, series, prefix, suffix, whitespaceTxt, trailingWSFlag, numPolicy, noNumPolicy, doAdjustFlag, adjustmentFlag)

Method this.numbering@(format wp_bead@ newNum, starting, series, prefix, suffix, whitespaceTxt, trailingWSFlag, numPolicy, noNumPolicy, doAdjustFlag, adjustmentFlag)

Arguments

newNum	The bead with (first) new number to insert or add.
starting	The starting number value (an integer).
series	A number series. The series can be one of the following values: -1 WP_NUMBERING_NONE 0 WP_NUMBERING_1 '1, 2, 3, ... 1 WP_NUMBERING_SMALL_A 'a, b, c, ... 2 WP_NUMBERING_CAP_A 'A, B, C, ... 3 WP_NUMBERING_SMALL_I 'i, ii, iii, iv, ... 4 WP_NUMBERING_CAP_I 'I, II, III, IV, ...
prefix	New prefix being used (and looked for) in existing numbers.
suffix	New suffix being used (and looked for) in existing numbers.
whitespaceTxt	Whitespace that follows the number.
trailingWSFlag	If set to TRUE, whitespace following the existing number is

replaced/deleted along with number. If set to FALSE, whitespace following the existing number is not deleted.

numPolicy Used for paragraphs that already have numbers. This is one of the following values:

0 WP_NUMBERING_NO_CHANGE
2 WP_NUMBERING_REPLACE
3 WP_NUMBERING_DELETE

noNumPolicy

Used for paragraphs that do not have numbers. This is one of the following values:

1 WP_NUMBERING_ADD
0 WP_NUMBERING_NO_CHANGE

doAdjustFlag

If set to TRUE indicates that paragraph margins should be adjusted.

adjustmentFlag

The offset of the left margin from the first margin; this parameter is only used if doAdjustFlag is set to TRUE.

Description This macro adds or removes numbers from the front of the paragraphs in the selected range.

WP_OBJECT_ALLOW_UNREF@

Allows embedded object to be saved if unreferenced

Format WP_OBJECT_ALLOW_UNREF@(name, allowUnrefFlag)

Method [this.object_allow_unref@\(name, allowUnrefFlag\)](#)

Arguments name The name of an embedded object within the Words document.

allowUnrefFlag

A Boolean value which if set to TRUE indicates that unreferenced objects are saved within the Words file. FALSE indicates that the object will be deleted.

Description This macro tells Words if it should save an unreferenced object when the Words file is saved.

Unreferenced objects are those that are not referenced by embedded object filed or cell custom shading. Normally, unreferenced objects are deleted when the document is saved.

See also [WP_OBJECT_IS_REFERENCED@](#)

WP_OBJECT_FILTER_MACRO@

Changes the embedded object filter macro

Format WP_OBJECT_FILTER_MACRO@(name, filterMacro)

Method [this.object_filter_macro@](#)(name, filterMacro)

Arguments name The name of an object within the Words document.
filterMacro The name of the filter macro that is used when object name is displayed.

Description Changes the filter macro that is used to display an object.

WP_OBJECT_IS_REFERENCED@

Indicates if the embedded object is referenced

Format flag = WP_OBJECT_IS_REFERENCED@(name)

Method [flag = this.object_is_referenced@](#)(name)

Arguments name The name of an object with the Words document.

Description Returns a TRUE/FALSE value indicating if object name is referenced.

See also [WP_OBJECT_ALLOW_UNREF@](#)

WP_OBJECT_LAUNCH_MACRO@

Changes an embedded object's launch macro

Format WP_OBJECT_LAUNCH_MACRO@(name, launchMacro)

Method [this.object_launch_macro@](#)(name, launchMacro)

Arguments name The name of an object within the Words document.
launchMacro The macro to be launched when a user double-clicks on an object.

Description Changes the launch macro associated with name. That is, the launch macro identifies the name of the application that is invoked when a user double-clicks on an object.

WP_OBJECT_NAME@

Changes an embedded object's name

Format WP_OBJECT_NAME@(name, newName)

Method [this.object_name@](#)(name, newName)

Arguments name The object name.
newName The new object name.

Description Changes the object name from name to newName. If references to name existed, they are changed to refer to newName.

WP_OBJECT_SET_TEMP_FILE@

Sets the temporary file associated with an embedded object

Format WP_OBJECT_SET_TEMP_FILE@(objectName, filename)

Method [this.object_set_temp_file@](#)(objectName, filename)

Arguments objectName The object's name.
filename The temporary file's file name.

Description Sets the temporary file associated with an object. If this macro is used, filename is opened when the local inset field that references this object is double-clicked. If the field has a user specified start macro, this temporary file is watched for changes.

For example, if the object is in WK3 format, and the user wants to launch Lotus 1-2-3 on a double click, the user can define the start macro that saves the object to disk (using [WP_SAVE_OBJECT@](#)), sets the object's temporary file to this file, and then launches Lotus 1-2-3. When that file is saved from Lotus 1-2-3, the object will be automatically updated.

WP_OPEN@

Loads and initializes a file

Format WP_OPEN@(filename)

Method [this.open@](#)(filename)

Arguments filename The name of the file being opened.

Description Opens and initializes an Applixware file. The following actions are performed.

- The file is loaded using **WP_LOAD_FILE@**.
- The file is initialized using **WP_INITIALIZE_DOC@**.
- If installable macros exist, they are installed using **WP_INSTALL_MACROS@**.

WP_OPEN_INSET@

Opens an inset

Format id = WP_OPEN_INSET@([bead])

Method id = this.open_inset@([bead])

Arguments bead The bead containing the inset.

Description Opens an application window containing an inset. If you omit the optional bead argument, Words opens the inset that is at the start of the current range or which is located at the current cursor. WP_OPEN_INSET@ can open embedded or linked objects.

The task id for the inset process is returned.

WP_OUTDENT@

Decreases a paragraph's level

Format WP_OUTDENT@()

Method this.outdent@

Description Decreases a paragraph's level by 1. In addition, if the paragraph's level is associated with an indentation, the paragraph will be reformatted to fit within these margins.

WP_OUTPUT_BEADS@

Writes beads to a document

Format WP_OUTPUT_BEADS@(fname, format arrayof wp_beads@ beads)

Method this.output_beads@(fname, format arrayof wp_beads@ beads)

Arguments fname The name of the file in which the beads will be placed.

beads An array wp_bead@ formats.

Description Writes the passed beads to the file named fname.

WP_OVERTYPE_TOGGLE@

Places user in overtype mode

Format WP_OVERTYPE_TOGGLE@()

Method [this.overtyp_toggle@](#)

Description If the user is in overtype mode, this macro places Words into normal mode. If the user is in normal mode, this macro places Words into overtype mode. ELF macro writers should not use this macro. Instead, they should use [WP SET OVERTYPE MODE@](#).

WP_PAGE_NUMBER_OF_LOC@

Returns combined section/page information

Format string = WP_PAGE_NUMBER_OF_LOC@(format [wp_location@](#) loc)

Method string = [this.page_number_of_loc@](#)(format wp_location@ loc)

Arguments loc A location in the section.

Description Returns the combined section/page number string for location loc. This location must be in the main flow.

See also [WP PAGE OF LOCATION@](#)

WP_PAGE_NUMBER_STRS_OF_LOC@

Returns individual section/page information

Format WP_PAGE_NUMBER_STRS_OF_LOC@(format wp_location@ loc, sectStr, separator, pageStr)

Method [this.page_number_strs_of_loc@](#)(format wp_location@ loc, sectStr, separator, pageStr)

Arguments loc A location in the section.
sectStr The section's section number (as a string). This is a returned value.

separator The separator between a section's section number and page string. This is a returned value.
pageStr The section's page number. This is a returned value.

Description Returns information describing the passed location's page number, its section separator, and any section designator.

WP_PAGE_OF_LOCATION@

Returns the page number for the passed location

Note This macro is obsolete. Use [WP_GET_PAGE_OF_LOCATION@](#) instead.

Format WP_PAGE_OF_LOCATION@(format wp_location@ loc)

Arguments loc The passed location.

Description Returns the page number for the passed location in a simple numeric zero-based format. loc *must* be within the main document flow.

See also [WP_PAGE_NUMBER_OF_LOC@](#)

WP_PAGE_SETUP@

Defines a section's page setup

Format WP_PAGE_SETUP@(format wp_page_setup@ page, applyType)

[Method](#) this.page_setup@(format wp_page_setup@ page, applyType)

Arguments page The new page definition.

applyType One of the following values:

WP#APPLY#WHOLE#DOCUMENT
WP#APPLY#THIS#POINT#FORWARD
WP#APPLY#CURRENT#SECTION
WP#APPLY#SELECTED#SECTIONS
WP#APPLY#SELECTED#TEXT

Description Defines the structure of a Words page. The definition for wp_page_setup@ is as follows. All units are in mils (1000 mils = 1 inch):

format wp_page_setup@

pageOrientation, 'WP#PAGE#LANDSCAPE_
'WP#PAGE#PORTRAIT_

pageWidth,
pageHeight,
leftMargin,

rightMargin,
topMargin,
bottomMargin,
bindingMargin.

WP_PASTE@

Pastes the contents of the clipboard

Format WP_PASTE@()

Method [this.paste@](#)

Description Pastes the contents of the clipboard at the current cursor position in the current Words document. WP_PASTE@ is called by Edit Y Paste.

Note WP_PASTE@ is not a synchronous call. That is, the next macro may execute before the material is actually pasted. This can occur if the clipboard's contents are owned by a program other than Applixware.

Example

See also [WP_COPY@](#)

[WP_CUT@](#)

WP_PASTE_COLUMNS@

Pastes a column of cells into a table

Format WP_PASTE_COLUMNS@()

Method [this.paste_columns@](#)

Description Pastes a column of cells into the current table.

Note WP_PASTE_COLUMNS@ is not a synchronous call. That is, the next macro may execute before the material is actually pasted. This can occur if the clipboard's contents are owned by a program other than Applixware.

See also [WP_PASTE_OVERWRITE@](#)

[WP_PASTE_COLS_AFTER@](#)

WP_PASTE_OVERWRITE@

Pastes a column of cells into a table, overwriting existing cells

Format WP_PASTE_OVERWRITE@()

Method [this.paste_overwrite@](#)

Description Pastes a column of cells into a table. The cells being written into the table replace existing cells.

Note WP_PASTE_OVERWRITE@ is not a synchronous call. That is, the next macro may execute before the material is actually pasted. This can occur if the clipboard's contents are owned by a program other than Applixware.

See also [WP_PASTE_COLUMNS@](#)
[WP_PASTE_COLS_AFTER@](#)

WP_PASTE_COLS_AFTER@

Inserts columns after current column

Format WP_PASTE_COLS_AFTER@()

Method [this.paste_cols_after@](#)

Description Pastes cells into the table by inserting a column (or columns) after the current column.

Note WP_PASTE_COLS_AFTER@ is not a synchronous call. That is, the next macro may execute before the material is actually pasted. This can occur if the clipboard's contents are owned by a program other than Applixware.

See also [WP_PASTE_COLUMNS@](#)
[WP_PASTE_OVERWRITE@](#)

WP_PLAY_AUDIO@

Plays an audio file

Format WP_PLAY_AUDIO@(filename)

Method [this.play_audio@\(filename\)](#)

Arguments filename The name of the audio file you want to play.

Description Plays an audio file on the computer's speaker.

WP_PRESERVE_FIELD_ATTR_EDITS@

Preserves attribute edits for Spreadsheets and Data objects

Format WP_PRESERVE_FIELD_ATTR_EDITS@()

Method [this.preserve_field_attr_edits@](#)

Description Starts preserving field value attributes for Spreadsheet and Data linked and embedded object fields. That is, the current state of the field value for these applications is preserved. After this done, all attribute edits to the field value are preserved; for example, boldness, point size, typeface, borders, shading, and so on. If the field is updated, these attributes are applied to the new data.

This action occurs automatically when a field is unprotected.

See also [WP_CLEAR_FIELD_ATTR_EDITS@](#)

WP_PRESERVE_SELECTION@

Preserves the current table selection

Format ncols = WP_PRESERVE_SELECTION@(name)

Method [ncols = this.preserve_selection@\(name\)](#)

Arguments name A unique name for the markers.

Description Saves the current selection by placing of a pair of named markers into the file at the beginning and end of the selection. The same marker name will be passed to [WP_RESTORE_SELECTION@](#) when you restore the selection. After calling this function, the selection *must* be restored before this function is called again.

This function returns the number of table columns currently selected (which must also be passed to the restore function).

WP_PREV_KEY@

Displays the previous page

Format WP_PREV_KEY@()

Method [this.prev_key@](#)

Description Scrolls to the previous screen. This is usually bound to the 'PgUp' key. This macro does not change the position of the cursor.

See also [WP_NEXT_KEY@](#)

WP_PREV_PAGE@

Moves backward to a specified page

Format WP_PREV_PAGE@(number)

Method [this.prev_page@\(number\)](#)

Arguments number The page number to which the cursor will move.

Description Moves the cursor to the indicated page. If number is greater than or equal to the current page number, no action (except a BEEP@) occurs.

Because of its inherent limitations, you should use the [WP_GOTO_PAGE@](#) macro.

WP_PREV_VISIBLE_BEAD@

Returns the bead number of the previous visible bead

Format prevBead = WP_PREV_VISIBLE_BEAD@(beadnum)

Method [prevBead = this.prev_visible_bead@\(beadnum\)](#)

Arguments beadnum The sequence number of the bead to start searching at.

Description Returns the bead number of the previous visible bead in the current flow. "Visibility" may depend on field method/value. 0 is returned if ELF finds the start of the current flow before finding a visible bead.

See also [WP_FIND_VISIBLE_BEAD@](#)

WP_PRINT@

Prints the current Words document

NOTE: While this macro is supported at this time, you should be using **WP_PRINT_DEVICE_SPECIFIC@**.

Format WP_PRINT@(printer, colorFlag, copies, bannerFlag, allPagesFlag, startPage, endPage, FALSE, mergeFile, tempFile, [, NULL] [, NULL] [, NULL] [, NULL] [, NULL] [, background] [,class] [, baggage])

Method this.print@(printer, colorFlag, copies, bannerFlag, allPagesFlag, startPage, endPage, FALSE, mergeFile, tempFile, [, NULL, NULL, NULL, NULL, NULL, background[, class]])

Arguments

printer	A string giving the printer name to print the document. If you want to print to a file rather than to a printer, set printer to NULL; the print file is placed in the directory specified by tempFile, or in the Applixware temporary directory if no directory is specified in tempFile.
colorFlag	Indicates whether the document is being printed on a color printer. If you are printing to a color printer, set colorFlag to TRUE; otherwise, set colorFlag to FALSE, which is the default. If this flag is set to TRUE and the job is sent to a black and white printer, colors are displayed in gray scale intensities.
copies	The number of copies to print. Default is 1.
bannerFlag	Indicates whether to include a banner page with the printed document. Set to TRUE if you want a banner page; FALSE if you don't, (which is the default).
allPagesFlag	Indicates whether to print all pages of the document. Set to TRUE to print all pages; FALSE if you want to print a range of pages.
startPage	A number indicating the first page to print if you are printing a range of pages. If allPagesFlag is TRUE, startPage is ignored.
endPage	A number indicating the last page to print if you are printing a range of pages. If allPagesFlag is TRUE, endPage is ignored.
mergeFile	The name of the .aw file to merge with the document when printing. Specify NULL if you are not using a merge file.
tempFile	The name of the print file. This file is created to print the document and is deleted when printing is completed (unless printer is set to NULL). If you do not supply a path name for tempFile, the print file is created in the Applixware temporary directory.
background	Indicates whether to print the document "in the background." Set to FALSE to not print in the background. If TRUE, a new axmain (an

additional Applixware server process) will be created, without checking out a new license. That new process will be terminated upon completing the print job.

class A constant indicating if printing will be performed on a PostScript or PCL device, as follows:

PostScript
PCL5

baggage an array of format [print baggage@](#).

Description Prints the current Words document using the attributes you specify. It is best to assign a FALSE value to background unless you are printing a very long document (approximately 20 pages or more). Otherwise, it may take longer to invoke the new Applixware process than to print in the foreground.

To collate copies (i.e., cause multiple copies to print as sets rather than printing all page 1's followed by all page 2's, and so on), set the axPrintCollate preference to TRUE before calling WP_PRINT@ as follows:

CHANGE_PREFS@("axPrintCollate", TRUE)

WP_PRINT_DEVICE_SPECIFIC@

Prints a file

Format WP_PRINT_DEVICE_SPECIFIC@(format wp_print_info@ info)

Method [this.print_device_specific@](#)(format wp_print_info@ info)

Arguments format wp_print_info@ info

Description Prints the current Words document using the attributes you specify. It is best to assign a FALSE value to background (within the wp_print_info@ format) unless you are printing a very long document (approximately 20 pages or more). Otherwise, it may take longer to invoke the new Applixware process than to print in the foreground.

To collate copies (i.e., cause multiple copies to print as sets rather than printing all page 1's followed by all page 2's, and so on), set the axPrintCollate preference to TRUE before calling WP_PRINT@ as follows:

CHANGE_PREFS@("axPrintCollate", TRUE)

The definition of wp_print_info@ is as follows:

format wp_print_info@

printername, 'Printer name or empty if to file.

A string giving the printer name to print the document. If you want to print to a file rather than to a printer, set printer to the empty string.

color, 'Boolean: print in color?

Indicates whether the document is being printed on a color

printer. If you are printing to a color printer, set colorFlag to TRUE; otherwise, set colorFlag to FALSE, which is the default.

If this flag is set to TRUE and the job is sent to a black and white printer, colors are displayed in gray scale intensities.

copies,	'Number of copies
banner,	'Indicates whether to include a banner page with the printed document. Set to TRUE if you want a banner page; FALSE if you don't (which is the default).
allpages,	'Boolean: print all pages Indicates whether to print all pages of the document. Set to TRUE to print all pages; FALSE if you want to print a range of pages.
range,	'Not used
merge,	'A string containing the name of the .aw file to merge with the document when printing. Specify NULL if you are not using a merge file.
file,	'The name of the temporary print file. This file is created to print the document and is deleted when printing is completed (unless printer is set to NULL). If you do not supply a path name for the temporary file, the print file is created in the Applixware temporary directory.
class,	'Printer class, which is one of the following strings: PCL5 PostScript
type,	'Printer Type
page1,	'A number indicating the first page to print if you are printing a range of pages. If allPages is TRUE, startPage is ignored.
pageN,	'A number indicating the last page to print if you are printing a range of pages. If allPages is TRUE, endPage is ignored.
draft,	'Obsolete
contins,	'Obsolete
changed,	'Obsolete
pgeExt1,	'Obsolete
pageExtN,	'Obsolete
background	'Boolean: Print in background? Indicates whether to print the document ``in the background." Set to FALSE to not print in the background. If TRUE, a new axmain (an additional Applixware server process) will be created, without checking out a new license. That new process will be terminated upon completing the print job.

WP_PROCESS_BEAD_ACTIONS@

Forces updates to occur before task can continue

Format WP_PROCESS_BEAD_ACTIONS@()

Method [this.process_bead_actions@](#)

Description Instructs Applixware Words to process all outstanding events and activities before performing additional activities. Operation that edit some type of beads in a document have side-effects. For example, editing a live glossary definition alters all matching glossary reference fields. Updating the glossary fields is a side-effect that occurs when Applixware Words is in its idle loop.

If an ELF macro continually performs actions upon a document, Words may not enter its idle loop until the macro finishes executing. In most cases, this is the what you want; that is, the macro makes a number of changes and the side effects are then all displayed.

In some cases, a macro may immediately need to use a side effect. Using this macro causes a side-effect to happen immediately. This also slows down performance.

Here are some examples of side effects:

- Revising other series fields following the insertion or deletion of a series field.
- Updating inherited headers and footers following a change to a section bead.
- Updating conditional fields because a document variable changed.
- Updating glossary fields because the glossary definition changed.
- Updating a page count field because information was inserted or deleted.
- Updating a cross reference field because the cross reference source changed.
- Updating a field value because the field method changed.

WP_PROCESS_SHIFT_TAB_KEY@

Selects the cell preceding the selection

Format WP_PROCESS_SHIFT_TAB_KEY@()

Method [this.process_shift_tab_key@](#)

Description If the selection is in a table, <shift/tab> selects the cell preceding the start of the current selection. If the selection is not in a table this macro inserts a tab.

WP_PROCESS_TAB_KEY@

Selects the cell following the selection

Format WP_PROCESS_TAB_KEY@()

Method [this.process_tab_key@](#)

Description If the selection is in a table, the TAB key selects the cell following the end of the current selection. If the selection is not in a table, this macro inserts a tab.

WP_PROMOTE@

Promotes a paragraph one level

Format WP_PROMOTE@()

Method [this.promote@](#)

Description Promotes the currently selected paragraphs, sections, or numbered list items to the next level. For example, if you promote a level two numbered list, it becomes a level one list. WP_PROMOTE@ is called by Attributes Ý Promote.

See also [WP_DEMOTE@](#)

WP_PROTECT_CURRENT_FIELD@

Protects (or unprotects) the current field

Format WP_PROTECT_CURRENT_FIELD@(protectFlag)

Method [this.protect_current_field@\(protectFlag\)](#)

Arguments protectFlag A Boolean value which if set to TRUE prevents changes to the field.

Description Protects (or unprotects) the current field. This prevents (or permits) casual edits to a field's value. If there is no current field, no action occurs.

WP_PROTECT_FIELD@

Protects (or unprotects) a field

Format WP_PROTECT_FIELD@(beadNum, protectFlag)

Method `this.protect_field@(beadNum, protectFlag)`

Arguments `beadNum` The field_start bead number.
`protectFlag` A Boolean value which if set to TRUE prevents changes to the field.

Description Protects (or unprotects) the passed Field Start (bead number). This prevents (or permits) casual edits to a field's value.

WP_PROTECT_QUERY@

Determines if a field is protected

Format `flag = WP_PROTECT_QUERY@(beadNum)`

Method `flag = this.protect_query@(beadNum)`

Arguments `beadNum` The field_start bead number.

Description Returns TRUE if the passed field indicated by beadNum is protected. This means that the value cannot be easily edited. If the passed Field Start bead is not protected, FALSE is returned.

WP_PUT_RANGE_IN_ROW@

Surrounds the passed range with a row

Format `WP_PUT_RANGE_IN_ROW@(format wp_range@ range[, sepChar[, width]])`

Method `this.put_range_in_row@(format wp_range@ range[, sepChar[, width]])`

Arguments `range` The passed range.
`sepChar` An optional argument that indicates the character used as a separator between columns in the row.
`width` An optional argument that specifies a row's width in millimeters. A value of 0 (zero) indicates that the row is stretched to fit within the current margins.

Description Transforms the text within a range into a row. If range contains more than one cell, indicate the character separating the cells with the optional sepChar argument. When the text is parsed by this macro, it replaces the separator character with a paragraph marker and a Cell End bead.

An optional width for the new row can be indicated. If you pass 0, the row is stretched between the page margins.

The new Row Start bead number is returned.

An error occurs if you perform this operation upon material already in a table, or upon the last paragraph in the document, or upon half a field, and so on.

WP_PUT_TEXT_IN_CELL@

Replaces cell contents with passed text

Format WP_PUT_TEXT_IN_CELL@(beadNum, text)

Method [this.put_text_in_cell@](#)(beadNum, text)

Arguments beadNum The bead number of the cell or the Entry field's Field Start bead number.
text The text string.

Description Replaces the cell contents with passed text.

See also [WP_GET_TEXT_OF_CELL@](#)

WP_QUICK_BULLETS@

Adds bullets to paragraphs

Format WP_QUICK_BULLETS@()

Method [this.quick_bullets@](#)

Description Repeats the most recently performed bullet operation. If bullets have not as yet been added to paragraphs, this macro will select a set of defaults before it adds them. All added information is added as text.

WP_QUICK_INSERT_FOOTER@

Adds an odd footer to the section

Format WP_QUICK_INSERT_FOOTER@()

Method [this.quick_insert_footer@](#)

Description Inserts a new odd footer into the current section and places the cursor within this footer. This command throws an error if the cursor is not in the main flow or if non-linked footer is defined for the current section.

WP_QUICK_INSERT_HEADER@

Adds an odd header to the section

Format WP_QUICK_INSERT_HEADER@()

Method [this.quick_insert_header@](#)

Description Inserts a new odd header into the current section and places the cursor within this header. This command throws an error if the cursor is not in the main flow or if non-linked footer is defined for the current section.

WP_QUICK_NUMBERING@

Adds list numbers to paragraphs

Format WP_QUICK_NUMBERING@()

Method [this.quick_numbering@](#)

Description Repeats the most recently performed add numbers operation. If numbers have not as yet been added to paragraphs, this macro selects a set of defaults before it adds them. All added information is added as text.

WP_RANGE_IS_CURSOR@

Checks range is a cursor or a selection

Format flag = WP_RANGE_IS_CURSOR@(format wp_range@ range)

Method flag = [this.range_is_cursor@\(format wp_range@ range\)](#)

Arguments range A range in the document.

Description Returns a Boolean value indicating if range is a cursor. TRUE indicates that it is a cursor; FALSE indicates that it is a selection.

WP_RECORD_MACRO@

Records a macro

Format WP_RECORD_MACRO@()

[Method](#) this.record_macro@

Description Begins recording a macro within a Words document.

WP_REDO@

Restores the last edit

Format WP_REDO@()

[Method](#) this.redo@

Description Replaces the last edit (or series of edits) performed by an Undo operation. WP_REDO@ is only valid if a WP_UNDO@ was previously executed. If there is no action to restore, an error occurs. WP_REDO@ is called by Edit ® Redo.

See also [WP_UNDO@](#)

WP_REDUCE_VIEW_SCALE@

Reduces the display size

Format WP_REDUCE_VIEW_SCALE@()

[Method](#) this.reduce_view_scale@

Description Changes the size at which information is displayed in the Words window by one level. (A *level* is one of the predefined sizes that are displayed on the View ® Zoom pull down menu.)

See also [WP_VIEW_SCALE@](#)

WP_REMOVE_DICT@

Removes a dictionary

Format WP_REMOVE_DICT@(name)

[Method](#) this.remove_dict@(name)

Arguments name The name of the dictionary that will be deleted.

Description Removes dictionary file name from the list of dictionaries and deletes the dictionary file from disk.

See also [WP_ADD_DICT@](#)

WP_GET_DICTS@

WP_REMOVE_ROW@

Changes a row to paragraphs

Format WP_REMOVE_ROW@(bead, sepString)

Method [this.remove_row@](#)(bead, sepString)

Arguments

bead	A bead in the row being deleted.
sepString	String that will be inserted in place of all intermediate Cell End Beads. A common choice is the string ", " (comma-space).

Description Changes the row containing bead into text. Each cell in the row will become a separate paragraph. In more technical terms, the Row Start and Cell End beads are deleted, but the material in the cells is left unchanged.

See also [WP_INSERT_ROWS@](#)

WP_RENAME@

Changes a file's name

Format WP_RENAME@(name[, NULL, grpAccess[, allAccess]])

Method [this.rename@](#)(name[, NULL, grpAccess[, allAccess]])

Arguments

name	The current file's new name.
NULL	Not used.
grpAccess	A number indicating the read and write permissions for the file for a member of the same group. This can be one of the following values: 0 No read or write permissions for the file. (default) 1 Read permission for the file 2 Read and write permissions for the file.
allAccess	A number indicating the read and write permissions for any user. This can be one of the following values: 0 No read or write permissions for the file. (default) 1 Read permission for the file 2 Read and write permissions for the file.

Description Renames the file in the current window to name and optionally sets the group and all access file privileges.

See also [WP_SAVE_RENAME@](#)

WP_RENAME_COLOR@

Changes a color's name

Format WP_RENAME_COLOR@(oldColorName, newColorName)

Method [this.rename_color@](#)(oldColorName, newColorName)

Arguments oldColorName The current color name.
newColorName The new color name.

Description Changes the name of an existing color in the document's color table from oldColorName to newColorName.

WP_REPAINT_WINDOW@

Redisplays the window

Format WP_REPAINT_WINDOW@()

Method [this.repaint_window@](#)

Description Redisplays (repaints) the window. This is only necessary if there is a display error.

WP_REPLACE_ALL@

Replaces one string with another

Format flag = WP_REPLACE_ALL@(string, wholeWordsFlag, matchCaseFlag, format wp_para_attrs@ paraAtts, format wp_text_attrs@ textAtts, replaceString, preserveCaseFlag)

Method flag = [this.replace_all@](#)(string, wholeWordsFlag, matchCaseFlag, format wp_para_attrs@ paraAtts, format wp_text_attrs@ textAtts, replaceString, preserveCaseFlag)

Arguments string The search string.
wholeWordsFlag
A Boolean value, where TRUE means that the search only

succeeds if the letters of the word are not contained within another word.

matchCaseFlag

A Boolean value, where TRUE means that the search only succeeds if the case of the letters is the same as the case of the letters in string.

paraAtts

A variable of type `wp_para_attrs@` that indicates paragraph attributes of the search string.

textAtts

A variable of type `wp_text_attrs@` that indicates characters attributes of the search string. For example, you could request that the search string be in bold.

replaceString

The replacement string.

preserveCaseFlag

A Boolean value, where TRUE means that the case of the replacement string will be the same as the original string.

Description Replaces all instances of string with `replaceString`.

For a discussion of pattern matching, see [Search Patterns](#).

WP_REPLACE_BEAD@

Replaces a bead

Format `WP_REPLACE_BEAD@(format wp_bead@ bead, beadNum)`

Method `this.replace_bead@(format wp_bead@ bead, beadNum)`

Arguments `bead` A new version of a bead.

`beadNum` The bead number being replaced.

Description Replaces an existing bead with another bead. Note that altering the wrong beads with the wrong data will cause severe problems.

WP_REPLACE_SELECTED@

Replaces selected text with the string specified

Format `WP_REPLACE_SELECTED@(string, preserveFlag)`

Method `this.replace_selected@(string, preserveFlag)`

Arguments `string` The string to replace selected text with.

preserveFlag

Indicates whether to preserve case when replacing the selected word. If preserve is TRUE, string will have the same case as the selected text. For example, if "Many" is the selected text and "few" is string, the replacement will appear in text as "Few."

If preserve is FALSE, string is inserted exactly as specified by string, regardless of the case of the selected text.

Description Inserts the specified string in place of selected text. string can include a pattern that indicates that the replacement text is located in the clipboard.

WP_REPLACE_STYLE@

Replaces one paragraph style with another

Format WP_REPLACE_STYLE@(oldStyleName, newStyleName, noReformat)

Method [this.replace_style@](#)(oldStyleName, newStyleName, noReformat)

Arguments

oldStyleName	The name of the paragraph style to replace
newStyleName	The name of the paragraph style that replaces OldStyle-Name
noReformat	A boolean. If TRUE, the macro reformats oldStyleName paragraphs with the newStyleName paragraph style.

Description Applies the paragraph style specified in newStyleName with the paragraph style specified in oldStyleName.

If the noReformat flag is TRUE, local formatting for each paragraph is retained. If noReformat is FALSE, local formatting is overridden. For example, if you want to retain the italics in your reformatted paragraphs, set noReformat to TRUE. If you wish to discard italics, set noReformat to FALSE.

For more information on paragraph styles in Applixware Words, see [Style Paragraph Attributes](#).

WP_RESET_BACKGROUND_COLOR@

Resets the background color of a Words document to the system default

Format WP_RESET_BACKGROUND_COLOR@()

Method [this.reset_background_color@](#)()

Description Resets the background color to the system default. Under UNIX, the system defaults are determined by the Applixware preferences axColorWorkBackGreen,

axColorWorkBackRed, and axColorWorkBackBlue. In NT, the default background color is set through the Control Panel, and stored in the registry.

WP_RESET_PARA_SETTINGS@

Resets paragraph attribute flags

Format WP_RESET_PARA_SETTINGS@(format wp_para_attrs_flags@ attrs)

Method [this.reset_para_settings@](#)(format wp_para_attrs_flags@ attrs)

Arguments attrs The paragraph attributes you are changing.

Description Resets paragraph attribute flags for all selected paragraphs. If a flag is set to TRUE, the attribute's flag is reset to inherit. If a flag is FALSE, the attribute's flag is unchanged.

See also [WP_SET_PARA_SETTINGS@](#)

WP_RESET_TEXT_ATTRS@

Resets the text attribute flags

Format WP_RESET_TEXT_ATTRS@()

Method [this.reset_text_attrs@](#)

Description Resets all text attributes back to their default state (which is that defined in the style). This macro removes all font changes, point size changes, bolding, underlining, and so on, that were applied.

See also [WP_SET_TEXT_ATTRS@](#)

WP_RESOLVE_PARA_TABS@

Returns an array of tab positions

Format format wp_tab@ tabArray = WP_RESOLVE_PARA_TABS@(beadNum)

Method [format wp_tab@ tabArray = this.resolve_para_tabs@](#)(beadNum)

Arguments beadNum A bead number contained within a paragraph.

Description Returns an array of tabs (in wp_tab@ format) representing the paragraph's tab positions. A paragraph's tabs are either all inherited or all local. For example, assume that a paragraph has a tab set at .75 inch. If you set another tab at 1.25 inch, the paragraph now has two local tabs, not a local and an inherited tab.

The returned array of tabs does not include tabs at the default .5 inch positions. beadNum can be the bead number of any bead in the paragraph including the paragraph's paragraph bead. If the cursor is within the paragraph, you can always use the bead number of the current cursor as an argument.

If you do not pass the value of a paragraph bead, this macro returns an empty array. Similarly, an empty array is also returned if no tabs are set for this paragraph.

If no tabs are set for a paragraph, Words will use its default rule of tab stops every .5 inch.

The definition of wp_tab@ is as follows:

```
format wp_tab@
    type,          'One of the following:
                   WP#TAB#LEFT_
                   WP#TAB#RIGHT_
                   WP#TAB#CENTER_
                   WP#TAB#DECIMAL_
                   WP#TAB#INDENT_
    deleted,       'Boolean: TRUE if tab deleted
    xpos,          'Indent of tab in mils (1000 mils = 1 inch)
    leaderStr,     'String: leadering for tab
    decimalChar    `Character: decimal control character
```

WP_RESOLVE_STYLE_TABS@

Returns tab array representing tabs that are defined by or inherited for a style

Format format wp_tab@ tabs = WP_RESOLVE_STYLE_TABS@(beadNum)

Method format wp_tab@ tabs = this.resolve_style_tabs@(beadNum)

Arguments beadNum The style's bead number.

Description Returns an array of tabs representing the tab stops that are either defined for or inherited by the style.

The actual position of the tabstops in a paragraph may be shifted due to the paragraph level; that is, the paragraph may have been promoted or demoted.

A pseudo-tab representing the left indent position is included in the returned array since a paragraph behaves as if tabstops is at this position.

The definition of wp_tab@ is as follows:

```
format wp_tab@
    type,          'One of the following:
                   WP#TAB#LEFT_
                   WP#TAB#RIGHT_
                   WP#TAB#CENTER_
```

	WP#TAB#DECIMAL_
	WP#TAB#INDENT_
deleted,	'Boolean: TRUE if tab deleted
xpos,	'Indent of tab in mils (1000 mils = 1 inch)
leaderStr,	'String: leadering for tab
decimalChar	'Character: decimal control character

See also [WP_GET_STYLE@](#)

WP_RESTORE_SELECTION@

Restores the current selection

Format WP_RESTORE_SELECTION@(name, nCols)

Method [this.restore_selection@](#)(name, nCols)

Arguments

name	The name specified in WP_PRESERVE_SELECTION@.
nCols	The number of columns as returned by WP_PRESERVE_SELECTION@.

Description Restores the selection created by [WP_PRESERVE_SELECTION@](#). Note that this operation deletes the markers inserted into the document that delimit the selection.

WP_RESTORE_TAGGED_SELECTION@

Restores the tagged selection

Format WP_RESTORE_TAGGED_SELECTION@(startTag, endTag, nCols)

Method [this.restore_tagged_selection@](#)(startTag, endTag, nCols)

Arguments

startTag	The starting tag name.
endTag	The ending tag name.
nCols	The number of columns in original selection.

Description Deletes the tags and restores the tagged selection within a table. See [WP_TAG_SELECTION@](#) for information on tagging a selection.

WP_RETRIEVE_TAGGED_RANGE@

Returns the tagged range

Format format [wp_range@](#) range = WP_RETRIEVE_TAGGED_RANGE@(startTag, endTag, noDelete)

Method format [wp_range@](#) range = this.retrieve_tagged_range@(startTag, endTag, noDelete)

Arguments

startTag	The starting tag name.
endTag	The ending tag name.
noDelete	A Boolean value which if set to TRUE indicates that the tags are not deleted.

Description Returns the tagged range between startTag and endTag and optionally deletes the tags.

WP_RETURN_KEY@

Inserts new paragraph

Format WP_RETURN_KEY@()

Method [this.return_key@](#)

Description Inserts a new paragraph at the current location. If the cursor is in the middle of a paragraph, the attributes of the new paragraph are the same as the original paragraph.

WP_REVERT@

Restores a Words document to its previous saved state

Format WP_REVERT@()

Method [this.revert@](#)

Description Restores a Words document to its state when last saved. This eliminates all changes made to the document since the last save. The document remains open. Unlike File ® Revert, WP_REVERT@ does not prompt to verify that the user really wants to restore the document, and discard any edits.

WP_RGB_TO_CMYK@

Converts a color from RGB to CMYK system

Format WP_RGB_TO_CMYK@(red, green, blue, cyan, magenta, yellow, black)

Method [this.rgb_to_cmyk@](#)(red, green, blue, cyan, magenta, yellow, black)

Arguments

red	A color's red component.
green	A color's green component.
blue	A color's blue component.
cyan	The color's converted cyan component.
magenta	The color's converted magenta component.
yellow	The color's converted yellow component.
black	The color's converted black component.

Description Converts a color expressed in RGB notation into a color expressed in CMYK notation.

See also [WP_CMYK_TO_RGB@](#)
[WP_HSB_TO_RGB@](#)
[WP_RGB_HSB@](#)

WP_RIGHT_ARROW_KEY@

Moves the cursor right one character

Format WP_RIGHT_ARROW_KEY@()

Method [this.right_arrow_key@](#)

Description Move the cursor right one character or item. For example, a paragraph marker or a line break are one character (if they are displayed).

See also [WP_LEFT_ARROW_KEY@](#)

WP_RGB_TO_HSB@

Converts a color from RGB to HSB system

Format WP_RGB_TO_HSB@(red, green, blue, hue, saturation, brightness)

Method [this.rgb_to_hsb@](#)(red, green, blue, hue, saturation, brightness)

Arguments

red	A color's red component.
green	A color's green component.
blue	A color's blue component.
hue	The color's converted hue component.
saturation	The color's converted saturation component.
brightness	The color's converted brightness component.

Description Converts a color expressed in RGB notation into a color expressed in HSB notation.

See also [WP CMYK TO RGB@](#)
[WP HSB TO RGB@](#)
[WP RGB CMYK@](#)

WP_ROW_NUM@

Returns a row number

Format rowNum = WP_ROW_NUM@(bead)

Method rowNum = this.row_num@(bead)

Arguments bead Any bead in a table.

Description Returns the row number of the row in which a passed bead resides. Row numbers are zero-based An error occurs if the passed bead is not in a row.

See also [WP_GET_ROW@](#)

WP_SAVE@

Saves the current Words document

Format WP_SAVE@()

Method this.save@

Description Writes the current Words document to a file. If the document has not been saved before, it will be saved to a file having the name displayed in the window title.

WP_SAVE_ATTTS@

Preserves attributes of current bead

Format WP_SAVE_ATTTS@()

Method [this.save_atts@](#)

Description Saves all paragraph attributes internally. These attributes can be applied to the current selection using [WP_APPLY_SAVED_ATTTS@](#).

WP_SAVE_OBJECT@

Writes an object to a file

Format WP_SAVE_OBJECT@(name, filename)

Method [this.save_object@\(name, filename\)](#)

Arguments

name	The object's internal name.
filename	The name of the file to which the object will be written.

Description Writes an object to a file and saves the file.

WP_SAVE_RENAME@

Saves the current file to a new name

Format WP_SAVE_RENAME@(name[, NULL, grpAccess[, allAccess]])

Method [this.save_rename@\(name\[, NULL, grpAccess\[, allAccess\]\]\)](#)

Arguments

name	The current file's new name.
NULL	Not used.
grpAccess	A number indicating the read and write permissions for the file for a member of the same group. This can be one of the following values: 0 No read or write permissions for the file. (default) 1 Read permission for the file 2 Read and write permissions for the file.
allAccess	A number indicating the read and write permissions for any user. This can be one of the following values:

- 0 No read or write permissions for the file. (default)
- 1 Read permission for the file
- 2 Read and write permissions for the file.

Description Renames the current file to name and saves it to disk. The file's access privileges are set using grpAccess and allAccess.

Example

See also [WP_RENAME@](#)

WP_SAVE_VERSION_3@

Saves the document as a Words 3.11 format document

Format WP_SAVE_VERSION_3@ ()

Method [this.save_version_3@](#)

Description Saves the document as a Words 3.11 format document.

WP_SCROLL_TO_FIRST_PAGE@

Scrolls to first page

Format WP_SCROLL_TO_FIRST_PAGE@ ()

Method [this.scroll_to_first_page@](#)

Description Scrolls the display to the first page. (The cursor does not move.)

See also [WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_LAST_PAGE@

Scrolls to last page

Format WP_SCROLL_TO_LAST_PAGE@ ()

Method [this.scroll_to_last_page@](#)

Description Scrolls the display to the last page. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_LEFT_SCREEN@

Scrolls to the left screen

Format WP_SCROLL_TO_LEFT_SCREEN@()

Method [this.scroll_to_left_screen@](#)

Description Scrolls the display so that undisplayed text to the left is displayed. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_NEXT_SCREEN@

Scrolls to the next screen

Format WP_SCROLL_TO_NEXT_SCREEN@()

Method [this.scroll_to_next_screen@](#)

Description Scrolls the display vertically to the next screen. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_PAGE_NUMBER@

Scrolls to a page number

Format WP_SCROLL_TO_PAGE_NUMBER@(page)

Arguments page The number of the page you want to scroll to.

Description Moves the Words display to the beginning of the specified page. The cursor is not moved.

WP_SCROLL_TO_PREV_SCREEN@

Scrolls to the previous screen

Format WP_SCROLL_TO_PREV_SCREEN@()

Method [this.scroll_to_prev_screen@](#)

Description Scrolls the display vertically to the previous screen. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_RIGHT_SCREEN@

Scrolls to the right screen

Format WP_SCROLL_TO_RIGHT_SCREEN@()

Method [this.scroll_to_right_screen@](#)

Description Scrolls the display so that undisplayed text to the right is displayed. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)

WP_SCROLL_TO_PREV_SCREEN@

WP_SELECTION_IS_MULTIPLE@

Indicates if columns are selected

Format flag = WP_SELECTION_IS_MULTIPLE@()

Method flag = this.selection_is_multiple@

Description Returns TRUE if one or more table columns are selected. Otherwise, FALSE is returned.

WP_SELECTION_LARGER@

Increases the selection of text in a Words document

Format WP_SELECTION_LARGER@()

Method this.selection_larger@

Description If no text is selected, WP_SELECTION_LARGER@ selects the character to the right of the cursor. Subsequent uses of WP_SELECTION_LARGER@ increase the selection of text in the following sequence: word, sentence, field, paragraph, entire document. WP_SELECTION_LARGER@ is called by Keys ® Enlarge selection.

See also WP_SELECTION_SMALLER@

WP_SELECTION_SMALLER@

Reduces the selection of text in a Words document

Format WP_SELECTION_SMALLER@()

Method this.selection_smaller@

Description Reduces the selection of text that has been selected using WP_SELECTION_LARGER@. The selection is reduced in the following sequence: paragraph, field, sentence, word, character, nothing selected. If the selected text was not selected as a result of using WP_SELECTION_LARGER@, the selection is reduced to a cursor. WP_SELECTION_SMALLER@ is called by Keys ® Reduce selection.

WP_SELECTION_TYPE_FOR_BORDERS@

Indicates the current border type

Format value =WP_SELECTION_TYPE_FOR_BORDERS@()

Method value = this.selection_type_for_borders@

Description Returns a value indicating the currently selected material's border type. This value is one of the following:

For paragraph borders

WP#BORDER#PARAGRAPH

For cell borders

WP#BORDER#SINGLE#ROW#SINGLE#COL

WP#BORDER#SINGLE#ROW#MULTI#COL

WP#BORDER#MULTI#ROW#SINGLE#COL

WP#BORDER#MULTI#ROW#MULTI#COL

For example, if this macro returns WP#BORDER#PARAGRAPH, then WP_SET_CURRENT_BORDER_ATTRS@ will apply paragraph borders to the current selection.

WP_SELECT_ALL@

Selects all material in the document

Format WP_SELECT_ALL@()

Method this.select_all@

Description Selects the entire document contents; that is, all information in the main flow is selected. WP_SELECT_ALL@ is called by Edit ® Select All.

WP_SELECT_CELL@

Selects current cell

Format WP_SELECT_CELL@()

Method this.select_cell@

Description Selects the entire contents of the current cell.

See also [WP_SELECT_NEXT_CELL@](#)

WP_SELECT_PREV_CELL@

WP_SELECT_COLUMN@

Selects table column

Format WP_SELECT_COLUMN@()

Method [this.select_column@](#)

Description Selects the entire column for each column in the selection. For example, suppose three cells in a row are selected. In this case, WP_SELECT_COLUMN@ selects three columns.

WP_SELECT_CURRENT_HDRFTR@

Selects current header or footer

Format WP_SELECT_CURRENT_HDRFTR@()

Method [this.select_current_hdrftr@](#)

Description Selects the entire contents of the header or footer in which the current cursor resides. If the cursor is not within a header or footer, no text is selected.

WP_SELECT_CURRENT_PARA@

Selects current paragraph

Format WP_SELECT_CURRENT_PARA@()

Method [this.select_current_para@](#)

Description Selects the paragraph in which the cursor resides. If more than one paragraph is in the selection, only the first paragraph is selected.

WP_SELECT_DOWN_ARROW@

Moves the current
selection down one line

Format WP_SELECT_DOWN_ARROW@()

Description If a selection exists that was created with a stroke from left to right, the selection is extended down one line. The new selection includes the current line from the cursor position to the end of the line, and all of the following line up to the cursor position. For example, the underlined portion of the following paragraph shows the result of calling `WP_SELECT_DOWN_ARROW@` with the cursor at the end of the word *integrated*.

Appixware is an integrated suite of applications that allows you to quickly and easily create compound documents.

If a selection exists that was created with a stroke from right to left, this macro shrinks the selection by one line.

See also [WP_SELECT_UP_ARROW@](#)

WP_SELECT_FIELD_VALUE@

Selects the field value

Format `WP_SELECT_FIELD_VALUE@()`

Method `this.select_field_value@`

Description Selects the contents of the field value based on the current cursor location. Before executing this macro, either the field must be selected or the cursor must be in a field.

See [WP_SELECT_FIELD_VALUE_OF_FIELD@](#) for more information.

WP_SELECT_FIELD_VALUE_OF_FIELD@

Selects the contents of a field value

Format `WP_SELECT_FIELD_VALUE_OF_FIELD@(fstart)`

Method `this.select_field_value_of_field@(fstart)`

Arguments `fstart` The passed field start bead.

Description Selects the contents of a field value. The contents of the field value of the passed Field Start bead is selected, not including the Field Split and Field End beads. If the field value is empty, the selection is a cursor. An error occurs if some other bead is passed.

See also [WP_SELECT_FIELD_VALUE@](#)

WP_SAVE_OBJECT@

Writes an object to a file

Format WP_SAVE_OBJECT@(name, filename)

Method [this.save_object@\(name, filename\)](#)

Arguments

name	The object's internal name.
filename	The name of the file to which the object will be written.

Description Writes an object to a file and saves the file.

WP_SAVE_RENAME@

Saves the current file to a new name

Format WP_SAVE_RENAME@(name[, NULL, grpAccess[, allAccess]])

Method [this.save_rename@\(name\[, NULL, grpAccess\[, allAccess\]\]\)](#)

Arguments

name	The current file's new name.
NULL	Not used.
grpAccess	A number indicating the read and write permissions for the file for a member of the same group. This can be one of the following values: 0 No read or write permissions for the file. (default) 1 Read permission for the file 2 Read and write permissions for the file.
allAccess	A number indicating the read and write permissions for any user. This can be one of the following values: 0 No read or write permissions for the file. (default) 1 Read permission for the file 2 Read and write permissions for the file.

Description Renames the current file to name and saves it to disk. The file's access privileges are set using grpAccess and allAccess.

Example

See also [WP_RENAME@](#)

WP_SAVE_VERSION_3@

Saves the document as a Words 3.11 format document

Format WP_SAVE_VERSION_3@()

Method [this.save_version_3@](#)

Description Saves the document as a Words 3.11 format document.

WP_SCROLL_TO_FIRST_PAGE@

Scrolls to first page

Format WP_SCROLL_TO_FIRST_PAGE@()

Method [this.scroll_to_first_page@](#)

Description Scrolls the display to the first page. (The cursor does not move.)

See also [WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_LAST_PAGE@

Scrolls to last page

Format WP_SCROLL_TO_LAST_PAGE@()

Method [this.scroll_to_last_page@](#)

Description Scrolls the display to the last page. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_LEFT_SCREEN@

Scrolls to the left screen

Format WP_SCROLL_TO_LEFT_SCREEN@()

Method [this.scroll_to_left_screen@](#)

Description Scrolls the display so that undisplayed text to the left is displayed. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_NEXT_SCREEN@

Scrolls to the next screen

Format WP_SCROLL_TO_NEXT_SCREEN@()

Method [this.scroll_to_next_screen@](#)

Description Scrolls the display vertically to the next screen. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_PAGE_NUMBER@

Scrolls to a page number

Format WP_SCROLL_TO_PAGE_NUMBER@(page)

Arguments page The number of the page you want to scroll to.

Description Moves the Words display to the beginning of the specified page. The cursor is not moved.

WP_SCROLL_TO_PREV_SCREEN@

Scrolls to the previous screen

Format WP_SCROLL_TO_PREV_SCREEN@()

Method [this.scroll_to_prev_screen@](#)

Description Scrolls the display vertically to the previous screen. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_RIGHT_SCREEN@](#)

WP_SCROLL_TO_RIGHT_SCREEN@

Scrolls to the right screen

Format WP_SCROLL_TO_RIGHT_SCREEN@()

Method [this.scroll_to_right_screen@](#)

Description Scrolls the display so that undisplayed text to the right is displayed. (The cursor does not move.)

See also [WP_SCROLL_TO_FIRST_PAGE@](#)
[WP_SCROLL_TO_LAST_PAGE@](#)
[WP_SCROLL_TO_LEFT_SCREEN@](#)
[WP_SCROLL_TO_NEXT_SCREEN@](#)
[WP_SCROLL_TO_PREV_SCREEN@](#)

WP_SELECTION_IS_MULTIPLE@

Indicates if columns are selected

Format flag = WP_SELECTION_IS_MULTIPLE@()

Method flag = this.selection_is_multiple@

Description Returns TRUE if one or more table columns are selected. Otherwise, FALSE is returned.

WP_SELECTION_LARGER@

Increases the selection of text in a Words document

Format WP_SELECTION_LARGER@()

Method this.selection_larger@

Description If no text is selected, WP_SELECTION_LARGER@ selects the character to the right of the cursor. Subsequent uses of WP_SELECTION_LARGER@ increase the selection of text in the following sequence: word, sentence, field, paragraph, entire document. WP_SELECTION_LARGER@ is called by Keys ® Enlarge selection.

See also [WP_SELECTION_SMALLER@](#)

WP_SELECTION_SMALLER@

Reduces the selection of text in a Words document

Format WP_SELECTION_SMALLER@()

Method this.selection_smaller@

Description Reduces the selection of text that has been selected using [WP_SELECTION_LARGER@](#). The selection is reduced in the following sequence: paragraph, field, sentence, word, character, nothing selected. If the selected text was not selected as a result of using WP_SELECTION_LARGER@, the selection is reduced to a cursor. WP_SELECTION_SMALLER@ is called by Keys ® Reduce selection.

WP_SELECTION_TYPE_FOR_BORDERS@

Indicates the current border type

Format value = WP_SELECTION_TYPE_FOR_BORDERS@()

Method value = this.selection_type_for_borders@

Description Returns a value indicating the currently selected material's border type. This value is one of the following:
For paragraph borders
WP#BORDER#PARAGRAPH
For cell borders
WP#BORDER#SINGLE#ROW#SINGLE#COL
WP#BORDER#SINGLE#ROW#MULTI#COL
WP#BORDER#MULTI#ROW#SINGLE#COL
WP#BORDER#MULTI#ROW#MULTI#COL
For example, if this macro returns WP#BORDER#PARAGRAPH, then WP_SET_CURRENT_BORDER_ATTRS@ will apply paragraph borders to the current selection.

WP_SELECT_ALL@

Selects all material in the document

Format WP_SELECT_ALL@()

Method [this.select_all@](#)

Description Selects the entire document contents; that is, all information in the main flow is selected. WP_SELECT_ALL@ is called by Edit ® Select All.

WP_SELECT_CELL@

Selects current cell

Format WP_SELECT_CELL@()

Method [this.select_cell@](#)

Description Selects the entire contents of the current cell.

See also [WP_SELECT_NEXT_CELL@](#)
[WP_SELECT_PREV_CELL@](#)

WP_SELECT_COLUMN@

Selects table column

Format WP_SELECT_COLUMN@()

Method `this.select_column@`

Description Selects the entire column for each column in the selection. For example, suppose three cells in a row are selected. In this case, `WP_SELECT_COLUMN@` selects three columns.

WP_SELECT_CURRENT_HDRFTR@

Selects current header or footer

Format `WP_SELECT_CURRENT_HDRFTR@()`

Method `this.select_current_hdrftr@`

Description Selects the entire contents of the header or footer in which the current cursor resides. If the cursor is not within a header or footer, no text is selected.

WP_SELECT_CURRENT_PARA@

Selects current paragraph

Format `WP_SELECT_CURRENT_PARA@()`

Method `this.select_current_para@`

Description Selects the paragraph in which the cursor resides. If more than one paragraph is in the selection, only the first paragraph is selected.

WP_SELECT_DOWN_ARROW@

Moves the current selection down one line

Format `WP_SELECT_DOWN_ARROW@()`

Description If a selection exists that was created with a stroke from left to right, the selection is extended down one line. The new selection includes the current line from the cursor position to the end of the line, and all of the following line up to the cursor position. For example, the underlined portion of the following paragraph shows the result of calling `WP_SELECT_DOWN_ARROW@` with the cursor at the end of the word *integrated*.

Applixware is an integrated suite of applications that allows you to quickly and easily create compound documents.

If a selection exists that was created with a stroke from right to left, this macro shrinks the selection by one line.

See also [WP_SELECT_UP_ARROW@](#)

WP_SELECT_FIELD_VALUE@

Selects the field value

Format WP_SELECT_FIELD_VALUE@()

Method [this.select_field_value@](#)

Description Selects the contents of the field value based on the current cursor location. Before executing this macro, either the field must be selected or the cursor must be in a field.

See [WP_SELECT_FIELD_VALUE_OF_FIELD@](#) for more information.

WP_SELECT_FIELD_VALUE_OF_FIELD@

Selects the contents of a field value

Format WP_SELECT_FIELD_VALUE_OF_FIELD@(fstart)

Method [this.select_field_value_of_field@\(fstart\)](#)

Arguments fstart The passed field start bead.

Description Selects the contents of a field value. The contents of the field value of the passed Field Start bead is selected, not including the Field Split and Field End beads. If the field value is empty, the selection is a cursor. An error occurs if some other bead is passed.

See also [WP_SELECT_FIELD_VALUE@](#)

WP_SELECT_LEFT@

Selects left one character

Format WP_SELECT_LEFT@()

Method [this.select_left@](#)

Description If no selection exists, selects the character to the left of the cursor. If a selection exists, adds the character to the left of the selection to the current selection.

See also [WP_SELECT_RIGHT@](#)

WP_SELECT_LEFT_ARROW@

Moves a part of the selection one character to the left

Format WP_SELECT_LEFT_ARROW@()

Method [this.select_left_arrow@](#)

Description If a selection exists, this macro moves the rightmost part of the selection one character to the left. If no selection exists, the character to the left of the cursor is selected.

If a selection exists that was initially created by a stroke from right to left, this macro moves the leftmost part of the selection one character to the left.

If a selection exists that was initially created by a stroke from left to right, this macro moves the rightmost part of the selection one character to the left.

See also [WP_SELECT_RIGHT_ARROW@](#)

WP_SELECT_LINE_DOWN@

Selects the current and next lines

Format WP_SELECT_LINE_DOWN@()

Method [this.select_line_down@](#)

Description Selects from the current cursor position to the end of the current line, then to that portion of the next line that starts at the beginning of the line to a point directly underneath the original cursor position.

For example, the underlined portion of this paragraph indicates the text that would be selected if the cursor was immediately in front of the word "h" in the word "this".

See also [WP_SELECT_LINE_UP@](#)

WP_SELECT_LINE_UP@

Selects the current and previous lines

Format WP_SELECT_LINE_UP@()

Method [this.select_line_up@](#)

Description Selects from the current cursor position to the beginning of the current line, then to the portion of the next line that is directly above the current cursor position.

For example, the underlined portion of this paragraph indicates the text that would be selected if the cursor was immediately after the word "text".

See also [WP_SELECT_LINE_DOWN@](#)

WP_SELECT_NEXT_CELL@

Selects the cell following the end of a selection

Format flag = WP_SELECT_NEXT_CELL@([wrapFlag])

Method [flag = this.select_next_cell@\(\[wrapFlag \]\)](#)

Arguments wrapFlag An optional Boolean value which if set to TRUE indicates that if the cursor is in the last cell of the table, it will be moved to the first cell of the table.

Description If the current selection is within a table (not whole rows), WP_SELECT_NEXT_CELL@ selects the cell following the end of the selection.

If the end of the table is reached and wrapFlag is TRUE, the first cell of the table is selected. Otherwise, the cursor is moved to a position after the table.

If wrapFlag is TRUE, TRUE is returned if the current selection is within a table. FALSE is only returned if the table has only one cell or all other cells are locked in Forms mode. TRUE means that a next cell was selected.

FALSE is returned if the current selection is not within a table (in which case the current selection is not altered).

In Form Mode, locked cells are skipped over.

See also [WP_SELECT_CELL@](#)
[WP_SELECT_PREV_CELL@](#)

WP_SELECT_NEXT_HYPERLINK@

Selects next
hyperlink in a document

Format WP_SELECT_NEXT_HYPERLINK@()

Description Selects the first hyperlink after the cursor in an Applixware Words document.

WP_SELECT_NEXT_SECTION@

Moves to the next section

Format flag = WP_SELECT_NEXT_SECTION@()

Method [flag = this.select_next_section@](#)

Description Selects the next section. WP_SELECT_NEXT_SECTION@ returns TRUE if there is one.

See also [WP_SELECT_PREV_SECTION@](#)

WP_SELECT_PARA_DOWN@

Selects the current and next paragraphs

Format WP_SELECT_PARA_DOWN@()

Method [this.select_para_down@](#)

Description Selects the current paragraph and extends the selection to include the next paragraph. If the cursor is already in the last paragraph of the flow, this macro selects this paragraph and beeps.

See also [WP_SELECT_PARA_UP@](#)

WP_SELECT_PARA_UP@

Selects the current and previous paragraphs

Format WP_SELECT_PARA_UP@()

Method [this.select_para_up@](#)

Description Selects the current paragraph and extends the selection to include the previous paragraph. If the cursor is already in the first paragraph of the flow, this macro selects this paragraph and beeps.

See also [WP_SELECT_PARA_DOWN@](#)

WP_SELECT_PREV_CELL@

Selects the cell preceding the start of the selection

Format flag = WP_SELECT_PREV_CELL@([wrapFlag])

Method flag = this.select_prev_cell@([wrapFlag])

Arguments wrapFlag An optional Boolean value which if set to TRUE indicates that if the cursor is in the first cell of the table, it will be moved to the last cell of the table.

Description If the current selection is within a table (not whole rows), WP_SELECT_PREV_CELL@ selects the cell preceding the start of the selection.

If the beginning of the table is reached and wrapFlag is TRUE, the last cell of the table is selected. Otherwise, the cursor is moved to a position before the table.

If wrapFlag is TRUE, TRUE is returned if the current selection is within a table. FALSE is only returned if the table has only one cell or all other cells are locked in Forms mode. TRUE means that a previous cell was selected.

FALSE is returned if the current selection is not within a table (in which case the current selection is not altered).

In Form Mode, locked cells are skipped over.

See also [WP_SELECT_CELL@](#)
[WP_SELECT_NEXT_CELL@](#)

WP_SELECT_PREV_HYPERLINK@

Selects previous hyperlink
in a document

Format WP_SELECT_PREV_HYPERLINK@()

Description Selects the first hyperlink before the cursor in an Applixware Words document.

WP_SELECT_PREV_SECTION@

Selects the previous section

Format WP_SELECT_PREV_SECTION@()

Method [this.select_prev_section@](#)

Description Selects the previous (entire) section. WP_SELECT_PREV_SECTION@ returns TRUE if there is one.

See also [WP_SELECT_NEXT_SECTION@](#)

WP_SELECT_RANGE@

Selects the passed range

Format WP_SELECT_RANGE@(format [wp_range@](#) range)

Method [this.select_range@\(format wp_range@ range\)](#)

Arguments range The passed range.

Description Selects the passed range.

WP_SELECT_RANGE_CELLS@

Selects the passed range of cells

Format WP_SELECT_RANGE_CELLS@(format [wp_range@](#) range, nCols)

Method [this.select_range_cells@\(format wp_range@ range, nCols\)](#)

Arguments range The passed range.
nCols The number of columns.

Description Selects the passed range consisting of a number of columns.

WP_SELECT_RANGE_FRAME@

Selects a frame's range

Format WP_SELECT_RANGE_FRAME@(format [wp_range@](#) range)

Method [this.select_range_frame@](#)(format wp_range@ range)

Arguments range The passed range.

Description Selects the passed range which should just be a frame.

WP_SELECT_RIGHT@

Extends selection one character to the right

Format WP_SELECT_RIGHT@()

Method [this.select_right@](#)

Description If no selection exists, selects the character to the right of the cursor. If a selection exists, adds the character to the right of the selection to the current selection.

See also [WP_SELECT_LEFT@](#)

WP_SELECT_RIGHT_ARROW@

Moves part of a selection one character to the right

Format WP_SELECT_RIGHT_ARROW@()

Method [this.select_right_arrow@](#)

Description If no selection exists, the character to the right of the cursor is selected.
If a selection exists that was initially created by a stroke from right to left, this macro moves the leftmost part of the selection one character to the right.
If a selection exists that was initially created by a stroke from left to right, this macro moves the rightmost part of the selection one character to the right.

See also [WP_SELECT_LEFT_ARROW@](#)

WP_SELECT_ROW@

Expands the selection so that no row is partially selected

Format WP_SELECT_ROW@()

Method [this.select_row@](#)

Description Expands the selection, if necessary, so that no row is partially selected. Note this macro leaves a single selection.

WP_SELECT_TABLE@

Expands the selection so that no table is partially selected

Format WP_SELECT_TABLE@()

Method [this.select_table@](#)

Description Expands the selection, if necessary, so that no table is partially selected. Note this macro leaves a single selection.

WP_SELECT_TO_FLOW_END@

Selects to the end of a flow

Format WP_SELECT_TO_FLOW_END@()

Method [this.select_to_flow_end@](#)

Description Selects from the start of the current selection to end of the current flow. That is, all information from the current location to the end of the current header, footer, footnote, or main flow.

WP_SELECT_TRAILING_SPACES@

Alters current selection to include/exclude trailing spaces

Format WP_SELECT_TRAILING_SPACES@(incSpacesFlag)

Method [this.select_trailing_spaces@\(incSpacesFlag\)](#)

Arguments incSpacesFlag If set to TRUE, trailing spaces are included.

Description Adjusts the end of the current selection to include or exclude trailing spaces. The selection is expanded or reduced. This function does nothing (and is harmless) in an intra-table ("multiple") selection since paragraphs do not span cells and therefore, cannot have trailing spaces.

WP_SELECT_UP_ARROW@

Moves the existing selection up one line

Format WP_SELECT_UP_ARROW@()

Description If a selection exists that was created with a stroke from right to left, the selection is extended up one line. The new selection includes the current line from the cursor position to the beginning of the line, and all of the preceding line up to the cursor position. For example, the underlined portion of the following paragraph shows the result of calling WP_SELECT_UP_ARROW@ with the cursor at the end of the word *compound*.

Applixware is an integrated suite of applications that allows you to quickly and easily create compound documents.

If a selection exists that was created with a stroke from left to right, this macro shrinks the selection by one line.

See also [WP_SELECT_DOWN_ARROW@](#)

WP_SELECT_WHOLE_LINES@

Selects entire lines

Format WP_SELECT_WHOLE_LINES@()

Method [this.select_whole_lines@](#)

Description Expands a selection so that no line in the current selection is partially selected. If the selection is a cursor, this macro selects the line containing the cursor. Note that the selection of a line may select more than that line due to legality rules (if a field starts on the line and ends elsewhere). Therefore, this function may end up selecting more than you expected, and the extra selections may be fractions of lines.

See also [WP_SELECT_WHOLE_LINE_DOWN@](#)
[WP_SELECT_WHOLE_LINE_UP@](#)

WP_SELECT_WHOLE_LINE_DOWN@

Selects the current line and the entire next line

Format WP_SELECT_WHOLE_LINE_DOWN@()

Method [this.select_whole_line_down@](#)

Description Selects the current line and then extends the selection to include the entire next line.

See also [WP_SELECT_WHOLE_LINES@](#)
[WP_SELECT_WHOLE_LINE_UP@](#)

WP_SELECT_WHOLE_LINE_UP@

Selects the current line and the entire previous line

Format WP_SELECT_WHOLE_LINE_UP@()

Method [this.select_whole_line_up@](#)

Description Selects the current line and then extends the selection to include the entire previous line.

See also [WP_SELECT_WHOLE_LINES@](#)
[WP_SELECT_WHOLE_LINE_DOWN@](#)

WP_SELECT_WHOLE_PARAS@

Selects paragraphs

Format WP_SELECT_WHOLE_PARAS@()

Method [this.select_whole_paras@](#)

Description Adjusts the current selection to ensure that no paragraph is partially selected. This function does nothing (and is harmless) in an intra-table ("multiple") selection since paragraphs do not span cells and thus there can be no partially selected paragraphs.

WP_SELECT_WORDS@

Selects words

Format WP_SELECT_WORDS@(trailFlag)

Method [this.select_words@\(trailFlag\)](#)

Arguments trailFlag If set to TRUE, trailing spaces are included.

Description Expands a selection so that no words are partially selected. If a cursor is selected, this macro selects the word (if one exists) at the cursor.

WP_SERIES_ASTERISKS@

Demonstration series using asterisks for footnote series

Format WP_SERIES_ASTERISKS@(value, fmtCode)

Method [this.series_asterisks@](#)(value, fmtCode)

Arguments value A number indicating the number of asterisks to be printed.
fmtCode Reserved for Applixware use.

Description A demonstration macro showing how you might define a series containing the asterisks that represent a footnote number. The text of this macro (slightly edited) is as follows:

```
function WP_SERIES_ASTERISKS@(value)
    var ix, str

    for ix = 1 to value ' values are 1-based
        str = str ++ "*"
    next ix
    return (str)
endfunction
```

See also [WP_DEFINE_SERIES@](#)
[WP_SERIES_DAYS@](#)
[WP_SERIES_MONTHS@](#)

WP_SERIES_DAYS@

Demonstration series showing a day of the week series

Format day = WP_SERIES_DAYS@(value[, fmtCode])

Method [day = this.series_days@](#)(value[, fmtCode])

Arguments value A number ranging from 1 to 7 representing the days of the week.
fmtCode Reserved for Applixware use.

Description A demonstration macro showing how you might define a series containing the days of the week. The text of this macro (slightly edited) is as follows:

```
function WP_SERIES_DAYS@(value)
    var days, v
```

```

    days = "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"
    v = (value - 1) MOD 7      ' Values are 1-based
    return (days[v])
endfunction

```

See also [WP_DEFINE_SERIES@](#)
[WP_SERIES_ASTERISKS@](#)
[WP_SERIES_MONTHS@](#)

WP_SERIES_MONTHS@

Demonstration series showing a months of the year series

Format month = WP_SERIES_MONTHS@(value[, fmtCode])

Method month = this.series_months@(value[, fmtCode])

Arguments value A number ranging from 1 to 12 representing the months of the year.
 fmtCode Reserved for Applixware use.

Description A demonstration macro showing how you might define a series containing the months of the year. The text of this macro (slightly edited) is as follows:

```

function WP_SERIES_MONTHS@(value)
    var months, v

    months = "January", "February", "March", "April", "May", "June", "July",
            "August", "September", "October", "November", "December"
    v = (value - 1) MOD 12      ' Values are 1-based
    return (months[v])
endfunction

```

See also [WP_DEFINE_SERIES@](#)
[WP_SERIES_ASTERISKS@](#)
[WP_SERIES_DAYS@](#)

WP_SET_ALL_BOUNDS_DISPLAY@

Turns boundary display on or off

Format WP_SET_ALL_BOUNDS_DISPLAY@(flag)

Method this.set_all_bounds_display@(flag)

Arguments flag A Boolean value which if set to TRUE indicates that all boundaries should be displayed.

Description Tells Words if borders should be displayed or hidden. This macro controls the display of cell, header/footer, and page boundaries.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing these boundaries.

See also [WP SET CELL BOUNDS DISPLAY@](#)
[WP SET FRAME BOUNDS DISPLAY@](#)
[WP SET HDRFTR BOUNDS DISPLAY@](#)
[WP SET PAGE BOUNDS DISPLAY@](#)
[WP TOGGLE ALL BOUNDS DISPLAY@](#)

WP_SET_ASCII_DOC@

Sets the document's save format

Format WP_SET_ASCII_DOC@(isAsciiFlag)

Method [this.set_ascii_doc@\(isAsciiFlag\)](#)

Arguments isAsciiFlag A Boolean value which if set to TRUE indicates that the document will be saved as plain text rather than in Applixware Words format.

Description Sets a document's ASCII mode indicating if the document is saved in Applixware Words format or as plain text.

WP_SET_BREAKS_DISPLAY@

Turns breaks display on or off

Format WP_SET_BREAKS_DISPLAY@(flag)

Method [this.set_breaks_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE indicates that breaks should be displayed.

Description Tells Words if it should display break indicators. This macro controls the display of all break types used by Words.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing breaks.

See also [WP TOGGLE BREAKS DISPLAY@](#)

WP_SET_CELL_ATTRS@

Sets a cell's attributes

Format WP_SET_CELL_ATTRS@(beadNum, format wp_cell_attrs@ cellAttrs)

Method [this.set_cell_attrs@\(beadNum, format wp_cell_attrs@ cellAttrs\)](#)

Arguments beadNum The Cell End bead.
cellAttrs The cell's new attributes.

Description Sets the cells attributes to those in the wp_cell_attrs@ formatted cellAttrs variable. A NULL entry indicates that the existing attribute should not be changed.

See also [WP_GET_CELL@](#)

WP_SET_CELL_BOUNDS_DISPLAY@

Turns cell boundary visibility on or off

Format WP_SET_CELL_BOUNDS_DISPLAY@ (flag)

Method [this.set_cell_bounds_display@ \(flag\)](#)

Arguments flag A Boolean value which if set to TRUE indicates that cell boundaries should be displayed.

Description Tells Words if it should display or hide cell boundaries. (Cell boundary borders are *not* the borders you explicitly set using Attributes ® Borders & Shading. They are the boundaries displayed by the View ® Cell Boundaries command.)

This macro changes the display state for the current file; it does not change the Applixware global display state for showing cell boundaries.

See also [WP_SET_ALL_BOUNDS_DISPLAY@](#)
[WP_TOGGLE_CELL_BOUNDS_DISPLAY@](#)

WP_SET_CELL_MARGINS@

Sets a cell's margins

Format WP_SET_CELL_MARGINS@(beadNum, left, right, top, bottom)

Method [this.set_cell_margins@\(beadNum, left, right, top, bottom\)](#)

Arguments beadNum The Cell End bead.

left	The cell's left margin in mils. (1000 mils = 1 inch)
right	The cell's right margin in mils.
top	The cell's top margin in mils.
bottom	The cell's bottom margin in mils.

Description Sets a cells top, bottom, left and right margin.

WP_SET_CELL_VERT_ALIGN@

Sets a cell's vertical alignment

Format WP_SET_CELL_VERT_ALIGN@(cell, verticalAlignment)

Method [this.set_cell_vert_align@](#)(cell, verticalAlignment)

Arguments cell The Cell End bead of the cell whose alignment is being set.
verticalAlignment
 One of the following values:
 WP#VERT#ALIGN#TOP
 WP#VERT#ALIGN#CENTER
 WP#VERT#ALIGN#BOTTOM

Description Sets the alignment of the information with a cell to one of the following:

- Top aligned
- Center aligned
- Bottom aligned

WP_SET_CELL_WIDTH@

Sets a cell's width

Format WP_SET_CELL_WIDTH@(beadnum, width)

Method [this.set_cell_width@](#)(beadnum, width)

Arguments beadnum A Cell End bead.
width The cell's width in mils. (1000 mils = 1 inch)

Description Sets a cell's width.

WP_SET_COLOR_TABLE@

Replaces the Words document's color table

Format WP_SET_COLOR_TABLE@(format arrayof [wp_color@](#) colorTable)

Method [this.set_color_table@](#)(format arrayof wp_color@ colorTable)

Arguments colorTable An array of wp_color@ values.

Description Replaces (or initially sets) the color table contained within the Words document. Each element in the array consists of the name of the color and the color's description in the CMYK color system.

The color table is document-specific. This table determines the colors that will show up on the Character Settings and Border and Shadings dialog boxes.

See also [WP_GET_COLOR_TABLE@](#)

WP_SET_COLUMN_INFO@

Defines a section's columns

Format WP_SET_COLUMN_INFO@(sectBeadNum, numCols, gutterwidth)

Method [this.set_column_info@](#)(sectBeadNum, numCols, gutterwidth)

Arguments sectBeadNum A section beads bead number.
numCols The number of columns that will exist in the section.
gutterwidth The width (in mils) that separates columns. (1000 mils = 1 inch)

Description Sets the number of columns and the space separating these columns in the indicated section.

See also [WP_GET_COLUMN_INFO@](#)

WP_SET_CONTROLS_DISPLAY@

Indicates if controls (paragraph and tab marks) are displayed

Format WP_SET_CONTROLS_DISPLAY@(value)

Method [this.set_controls_display@](#)(value)

Arguments value A Boolean value where TRUE indicates that controls are displayed.

Description Displays (or removes from the display) the special symbols that indicate paragraph markers and tabs.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing controls.

See also [WP TOGGLE CONTROLS DISPLAY@](#)

WP_SET_CURRENT_BORDER_ATTRS@

Sets border attributes

Format WP_SET_CURRENT_BORDER_ATTRS@(format [wp_border_attrs@](#) borderAttrs)

Method [this.set_current_border_attrs@](#)(format wp_border_attrs@ borderAttrs)

Arguments borderAttrs A set of border attributes formatted into a wp_border_attrs@ format.

Description Sets all of the cell or paragraph's borders. Because this macro changes all attributes, you should use [WP_GET_CURRENT_BORDER_ATTRS@](#) to initialize borderAttrs, then change the attributes that need changing.

WP_SET_CURRENT_COLUMN_INFO@

Defines current section's columns

Format WP_SET_CURRENT_COLUMN_INFO@(numCols, gutterwidth)

Method [this.set_current_column_info@](#)(numCols, gutterwidth)

Arguments numCols The number of columns that will exist in the current section.
gutterwidth The width (in mils) that separates columns. (1000 mils = 1 inch)

Description Sets the number of columns and the space separating these columns in the current section.

See also [WP_GET_CURRENT_COLUMN_INFO@](#)

WP_SET_CURRENT_GUTTER@

Sets the inter-column spacing

Format WP_SET_CURRENT_GUTTER@(widthInMils)

Method `this.set_current_gutter@(widthInMils)`

Arguments `widthInMils` The space between this section's columns.

Description Sets the space between all columns in the current section. The argument is the number of mils between the columns. (1000 mils = 1 inch)

WP_SET_CURRENT_HDRFTR_INFO@

Sets header/footer types and names

Format `WP_SET_CURRENT_HDRFTR_INFO@(format wp_hdrftr@ hdrftrInfo)`

Method `this.set_current_hdrftr_info@(format wp_hdrftr@ hdrftrInfo)`

Arguments `hdrftrInfo` The new header/footer information.

Description Sets the types and names of headers and footers for the current section. For more information, see [WP_GET_CURRENT_HDRFTR_INFO@](#).

WP_SET_CURRENT_LEFT_MARGIN@

Sets the section's left margin

Format `WP_SET_CURRENT_LEFT_MARGIN@(widthInMils)`

Method `this.set_current_left_margin@(widthInMils)`

Arguments `widthInMils` The new left margin in mils. (1000 mils = 1 inch)

Description Sets the left margin for the current section.

WP_SET_CURRENT_PAGE_SETUP@

Sets the page's size and orientation

Format `WP_SET_CURRENT_PAGE_SETUP@(format wp_page_setup@ pageSetup)`

Method `this.set_current_page_setup@(format wp_page_setup@ pageSetup)`

Arguments `pageSetup` The new page setup information formatted as a `wp_page_setup@` variable.

Description Sets the current section's page orientation, width, height, and left, right, top, bottom, and binding margins.

If any element in pageSetup is set to NULL, this macro will not change the current value.

WP_SET_CURRENT_RIGHT_MARGIN@

Sets the section's right margin

Format WP_SET_CURRENT_RIGHT_MARGIN@(widthInMils)

Method [this.set_current_right_margin@\(widthInMils\)](#)

Arguments widthInMils The new right margin in mils. (1000 mils = 1 inch)

Description Sets the right margin for the current section.

WP_SET_CURSOR@

Moves the cursor's location

Format WP_SET_CURSOR@(format [wp_location@](#) loc)

Method [this.set_cursor@\(format wp_location@ loc\)](#)

Arguments loc A location.

Description Sets the cursor so that it is at location loc.

See also [WP_LOCATION_OF_CURSOR@](#)

WP_SET_DICTS@

Sets the dictionaries to be used for spell checking

Format WP_SET_DICTS@(dictArray)

Method [this.set_dicts@\(dictArray\)](#)

Arguments dictArray An array containing the file names of dictionaries to be used for spell checking.

Description Sets (or replaces) the names of the dictionaries that are used for spell checking. The maximum number of dictionaries is eight.

See also [WP_GET_DICTS@](#)

WP_SET_DEFAULT_POINTER@

Sets the default pointer shape

Format WP_SET_DEFAULT_POINTER@(pointerName)

Method [this.set_default_pointer@\(pointerName\)](#)

Arguments pointerName The name of the new default pointer.

Description Selects the mouse pointer used by default. This is the pointer that is used within the Words's Window excluding its margin. After using this macro, the default pointer is changed for all Words windows.

If you wish to always use this pointer, it can be made permanent by setting a preference.

To obtain a list of available shape names, use [GET CURSOR LIST@](#).

See also [WP SET TEXT POINTER@](#)

WP_SET_DOC_ATTRS@

Sets the document's attributes

Format WP_SET_DOC_ATTRS@(levelIndent, printIndexFlag, printTOCFlag, widowOrphanFlag, hyphMethod)

Method [this.set_doc_attrs@\(levelIndent, printIndexFlag, printTOCFlag, widowOrphanFlag, hyphMethod\)](#)

Arguments levelIndent The amount of space in mils (1000 mils = 1 inch) paragraphs are indented using Promote/Demote.

printIndexFlag

A Boolean value, which if TRUE indicates that the index should be printed with the document.

printTOCFlag

A Boolean value, which if TRUE indicates that the table of contents should be printed with the document.

widowOrphanFlag

A Boolean value, which if TRUE indicates that widows and orphans are allowed.

hyphMethod The method used for hyphenation. This can be one of the following values:

WP#HYPH#CALC#ONLY
 Rule-based only
 WP#HYPH#DICTI#CALC
 Dictionary with rule-based backup
 WP#HYPH#NONE
 Don't hyphenate
 WP#HYPH#DICT#ONLY
 Dictionary only
 WP#HYPH#PROFILE
 Use Profile setting

Description Sets information about five document attributes.

See also [WP_GET_DOC_ATTRS@](#)

WP_SET_DOC_INFO@

Sets file properties

Format WP_SET_DOC_INFO@(format doc_format_ info)

Method [this.set_doc_info@](#)(format doc_format_ info)

Arguments info The document information being set. The definition for this format is found in fileinf_.am.

Description Sets operating system-level information for a file. This information is as follows:

```

format doc_format_,
  name,           ' The full path name of the current document.
  docid,          ' Not important: The document's unique ID number.
  on_disk,        ' Indicates if the document has been saved or
                  ' read using the file name specified by name.
                  ' Returns TRUE if the file has been saved or
                  ' read. Returns FALSE if it hasn't.
  save_mode,      ' The mode in which the document has been
                  ' saved. save_mode can be one of the following:
                  '0 Not supported: binary format
                  '1 Machine-independent (portable) format
  grp_access,     ' The permission setting for the file for members
                  ' of the same group. grp_access can be one
                  ' of the following:
                  ' - No read or write permissions for the file
                  '1 Read permission for the file.
                  '2 Read and write permission for the file
  all_access,     ' One of the following file permissions for the users:
                  '0 No read or write permissions for the file
                  '1 Read permission for the file.
  
```

writeable, '2 Read and write permission for the file.
' Indicates whether this user has write access.
' TRUE indicates that this user can write
' this file; the value FALSE indicates that
' the user cannot write this file.

See also [WP_GET_DOC_INFO@](#)
[WP_SAVE_RENAME@](#)

WP_SET_DOC_LINKS_INFO@

Changes file's linked field attributes

Format WP_SET_DOC_LINKS_INFO@(filename, format arrayof wp_link_info@
docLinksInfo)

Method [this.set_doc_links_info@](#)(filename, format arrayof wp_link_info@ docLinksInfo)

Arguments filename The name of the file whose object's attributes are being changed.
docLinksInfo An array containing information about all links.

Description Changes attributes of all links within a file. The definition of wp_link_info@ is as follows:

format wp_link_info@
name, 'string:filename
docType, 'int: doc type returned by
' WP_RECOGNIZE_FILE@
appType, 'int: app type returned by
' WP_RECOGNIZE_FILE@
filterMacro, 'string:import filter macro
launchMacro,
'string:macro to launch application
viewname 'string:viewname (for spreadsheet)

See also [WP_GET_DOC_LINKS_INFO@](#)

WP_SET_DOC_WINDOW_ATTR_TYPE@

Sets how the document's window attributes are saved

Format WP_SET_DOC_WINDOW_ATTR_TYPE@(type)

Method [this.set_doc_window_attr_type@](#)(type)

Arguments type One of the following values:

WP#DOC#WIN#ATTR#DEFAULT

Uses the default rules to determine the window size. Size information is not saved within the file.

WP#DOC#WIN#ATTR#LAST

The window size is saved every time the file is saved. Thus, if the window is resized between saves, the last size is recorded. When the file is re-opened, the window's size will be the same as it was when it was last saved.

WP#DOC#WIN#ATTR#FIXED

The window size is saved in a document variable. However, subsequent saves of the file do not alter the saved size. When the file is re-opened, the window's size will be the same as it was when it was first saved.

Description Indicates how the document window's size will be saved to a file.

You can retrieve this value using [WP_GET_DOC_WINDOW_ATTR_TYPE@](#).

WP_SET_DOC_VAR@

Sets a document variable

Format WP_SET_DOC_VAR@(name, value)

Method [this.set_doc_var@\(name, value\)](#)

Arguments

name	The name of the document variable.
value	The value of the document variable.

Description Sets a document's variable. Document variables (and their values) are stored with the document. These values are retained between Applixware sessions.

For additional information, see [WP_GET_DOC_VAR@](#) and [WP_GET_DOC_VAR_NAMES@](#).

WP_SET_FACING_PAGES@

Sets if facing pages are used

Format WP_SET_FACING_PAGES@(facingFlag)

Method [this.set_facing_pages@\(facingFlag\)](#)

Arguments

facingFlag	A Boolean value which if set to TRUE indicates that facing pages are to be used.
------------	--

Description Tells Words if the current document should be formatted using *facing pages*; that is, should the left and right pages be the same (FALSE) or different (TRUE).

See also [WP_GET_FACING_PAGES@](#)

WP_SET_FIELDS_DISPLAY@

Turns field visibility on or off

Format WP_SET_FIELDS_DISPLAY@(flag)

Method [this.set_fields_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE indicates that field methods are displayed. FALSE indicates that field methods are not displayed.

Description Tells Words if it should display fields or the field's method.
This macro changes the display state for the current file; it does not change the Applixware global display state for showing these fields.

See also [WP_TOGGLE_FIELDS_DISPLAY@](#)

WP_SET_FLAGGED_PARA_ATTRS@

Sets paragraph attributes based on flags

Format WP_SET_FLAGGED_PARA_ATTRS@(format wp_flagged_para_attrs@ fatts)

Method [this.set_flagged_para_attrs@\(format wp_flagged_para_attrs@ fatts\)](#)

Arguments fatts The flagged attributes.

Description Sets paragraph attributes based on flags.

See [WP_GET_FLAGGED_PARA_ATTRS@](#) for more information.

WP_SET_FLAGGED_TEXT_ATTRS@

Sets text attributes based on flags

Format WP_SET_FLAGGED_TEXT_ATTRS@(format wp_flagged_text_attrs@ fatts)

Method [this.set_flagged_text_attrs@\(format wp_flagged_text_attrs@ fatts\)](#)

Arguments fatts The flagged attributes.

Description This function sets text attributes based on flags. You can set individual attributes to inherit.

See also [WP_GET_FLAGGED_TEXT_ATTRS@](#)

WP_SET_FORM_MODE@

Enter form mode if flag is TRUE

Format WP_SET_FORM_MODE@(flag)

Method [this.set_form_mode@\(flag\)](#)

Arguments flag A Boolean value, where TRUE means enter form mode.

Description This functions sets the form mode of the Words document.

See also [WP_GET_FORM_MODE@](#)
[WP_TOGGLE_FORM_MODE@](#)

WP_SET_FRAME_BOUNDS_DISPLAY@

Turns frame boundaries on or off

Format WP_SET_FRAME_BOUNDS_DISPLAY@(flag)

Method [this.set_frame_bounds_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE, tells Words that it should display frame boundaries.

Description Controls the display of frame borders.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing frame boundaries.

See also [WP_SET_ALL_BOUNDS_DISPLAY@](#)
[WP_TOGGLE_FRAME_BOUNDS_DISPLAY@](#)

WP_SET_FRAME_LEFT_EDGE@

Sets a frame's left position

Format WP_SET_FRAME_LEFT_EDGE@(milpos)

Method `this.set_frame_left_edge@(milpos)`

Arguments `milpos` The frame's left position in mils. (1000 mils = 1 inch)

Description Sets the position of the left edge of the current frame. This position is an absolute setting based on the page's dimension. That is, it is relative to the 0 (zero) setting on the ruler. (The zero position on the ruler can be at the left indent based on the Word's preference [Display Ruler Zero at Page Edge Rather than Margin.](#))

WP_SET_FRAME_WIDTH@

Sets a frame's width

Format `WP_SET_FRAME_WIDTH@(milwidth)`

Method `this.set_frame_width@(milwidth)`

Arguments `milwidth` The frame's width, in mils. (1000 mils = 1 inch)

Description Sets the width of the current frame.

See also [WP_GET_FRAME_ATTRS@](#)

WP_SET_GLOSSARY_INFO@

Changes glossary information

Format `flag = WP_SET_GLOSSARY_INFO@(name, newName, liveFlag)`

Method `flag = this.set_glossary_info@(name, newName, liveFlag)`

Arguments

<code>name</code>	The glossary entry's existing name.
<code>newName</code>	The glossary's new name. (This can be the same as the old name.)
<code>liveFlag</code>	A Boolean value which if set to TRUE indicates that the entry should be live; that is, glossary references update when glossary definitions change.

Description Changes an existing glossary's name or link status. This macro returns TRUE if the glossary exists.

See also [WP_GET_GLOSSARY_INFO@](#)

WP_SET_GRAPHICS_DISPLAY@

Toggles graphics display

Format WP_SET_GRAPHICS_DISPLAY@(flag)

Method [this.set_graphics_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE indicates that graphics are displayed.

Description Turns the display of graphics on or off.

Graphics will always print even if the display is off. Also, space is reserved for the graphic when the display is off.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing graphics.

See also [WP TOGGLE GRAPHICS DISPLAY@](#)

WP_SET_HDRFTR_BOUNDS_DISPLAY@

Turns header/footer boundary display on or off

Format WP_SET_HDRFTR_BOUNDS_DISPLAY@(flag)

Method [this.set_hdrftr_bounds_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE tells Words that it should display the boundaries surrounding headers and footers.

Description Controls the display of header/footer boundaries.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing these boundaries.

See also [WP SET ALL BOUNDS DISPLAY@](#)
[WP TOGGLE HF BOUNDS DISPLAY@](#)

WP_SET_HDRFTR_INFO@

Updates section information

Format WP_SET_HDRFTR_INFO@(section, format wp_hdrftr@ hdrftrInfo)

Method [this.set_hdrftr_info@\(section, format wp_hdrftr@ hdrftrInfo\)](#)

Arguments section The section's bead number.
hdrftrInfo The new header/footer information.

Description Sets the section's header and footer information.

See also [WP_GET_CURRENT_HDRFTR_INFO@](#)

WP_SET_HDRFTR_MARGINS@

Sets header and footer margins

Format WP_SET_HDRFTR_MARGINS@(headerMargin, footerMargin)

Method [this.set_hdrftr_margins@\(headerMargin, footerMargin\)](#)

Arguments headerMargin The new header margin in mils. (1000 mils = 1 inch)
footerMargin The new footer margin in mils.

Description Sets new header and footer margins for the document.

See also [WP_GET_HDRFTR_MARGINS@](#)

WP_SET_HOOK@

Causes a macro to execute when the document is opened

Format WP_SET_HOOK@(macroName)

Method [this.set_hook@\(macroName\)](#)

Arguments macroName The name of the macro. If macroName is not loaded when this macro executes, ELF will load, compile, and execute a file with the same name (after a .am extension is appended) as macroName that exists in your default path.

Description Embeds within a Words document an instruction to run a given macro every time a Words document is opened. To unset the WP_SET_HOOK@ ``hook" in a Words document, run WP_SET_HOOK@ without any arguments.

If a document that does not have a hook macro is opened, the macro specified by the [Macro to Run when Starting Words](#) preference will be run.

See also [WP_GET_HOOK@](#)

WP_SET_HTML_MODE@

Set the mode of the HTML editor

Format WP_SET_HTML_MODE@(mode)

Arguments mode A number indicating the mode in which to place the editor.

0	Reserved for use by Applixware Words and the Macro editor
1	Place the editor in HTML Edit Mode
2	Place the editor in HTML View Mode

Description The mode argument determine the mode in which to place the editor. Typically this macro is used within Applixware HTML Author to toggle between View Mode and Edit Mode.

WP_SET_HTML_TAGS_DISPLAY@

Turns HTML Tags display on and off in the HTML Author

Format WP_SET_HTML_TAGS_DISPLAY@(flag)

Arguments flag A Boolean. If TRUE, the HTML Author displays HTML Tags. If false, HTML tags are displayed as formatted text.

Description Allows you to display HTML tags either as editable tags (flag = TRUE) or formatted text (flag = FALSE). The following shows two lines of HTML text. The top line is unformatted. The bottom line is formatted.

<H3>Soft Cheeses</H3>

Soft Cheeses

WP_SET_INHERITS@

Sets text inheritance attribute

Format WP_SET_INHERITS@(typeface, size, bold, italic, underline, strikethru, shiftcase, color)

Method [this.set_inherits@](#)(typeface, size, bold, italic, underline, strikethru, shiftcase, color)

Arguments typeface A Boolean value which if set to TRUE indicates that typeface is inherited.

size	A Boolean value which if set to TRUE indicates that size is inherited.
bold	A Boolean value which if set to TRUE indicates that bold is inherited.
italic	A Boolean value which if set to TRUE indicates that italic is inherited.
underline	A Boolean value which if set to TRUE indicates that underline is inherited.
strikethru	A Boolean value which if set to TRUE indicates that strikethru is inherited.
shiftcase	Not used.
color	A Boolean value which if set to TRUE indicates that color is inherited.

Description Sets the text attributes to an inherit/non-inherited stated. (More precisely it is setting the inheritFlags array within the wp_flagged_text_attrs@ format.

See also [WP_GET_FLAGGED_TEXT_ATTRS@](#)

WP_SET_LANGUAGE@

Sets the language for spell-checking, hyphenation, and thesaurus

Format WP_SET_LANGUAGE@(code[, createFlag])

Method [this.set_language@](#)(code[, createFlag])

Arguments	code	0 - not set
		1 - English
		2 - German
		3 - French
		4 - Spanish
		5 - Italian
		6 - British English
		7 - Swedish
		8 - Danish
		9 - Norwegian
		10 - Dutch
		11 - Portuguese
		12 - Brazilian Portuguese
		13 - Canadian French
		14 - Swiss German
	15 - Nynorsk (Norwegian dialect)	

createFlag A Boolean value which if set to TRUE tells Words that it should create the dictionary if it does not exist. If this argument is omitted, the default is FALSE.

Description Sets the language used by the dictionary and thesaurus in a given document. The language setting is set in the user's ax_prof4 [Language Used by Dictionary and Thesaurus](#) preference. To read the language setting in a document, see:

[WP_GET_CURRENT_LANGUAGE@](#)

To set the default language for this user (in the user's ax_prof4 file), see:

- [EDITPREFS@](#)
- [CHANGE_PREFS@](#)

WP_SET_LANGUAGE@ is called by Utilities ® Spelling ® Change Language.

See also [WP_GET_LANGUAGE@](#)

WP_SET_LINKS_INFO@

Changes information for all links

Format WP_SET_LINKS_INFO@(format arrayof [wp link info@](#) infoList)

Method this.set_links_info@(format arrayof wp_link_info@ infoList)

Arguments infoList A list of all links and the data associated with them.

Description Changes information for all links.

See also [WP_SET_LINKS_INFO@](#)

WP_SET_MARKER_BEAD_FLAG@

Allows/prohibits deleting a marker bead

Format WP_SET_MARKER_BEAD_FLAG@(name, flagVal)

Method this.set_marker_bead_flag@(name, flagVal)

Arguments name The name of an existing marker.
flagVal If set to TRUE, the bead can be deleted.

Description Indicates if an existing marker bead can be deleted.

WP_SET_MODIFIED@

Sets the document's *modified* flag

Format WP_SET_MODIFIED@(flag)

Method [this.set_modified@](#)(flag)

Arguments flag A Boolean value which if set to TRUE indicates that the document needs saving.

Description Sets a document's modified flag. If this flag is set to TRUE, Words will ask if the document needs saving when you exit. If it is set to FALSE, Words will assume that the document is saved and will simply exit.

See also [WP_MODIFIED@](#)

WP_SET_MULTIPLE_SELECTION@

Creates an intra-table selection

Format WP_SET_MULTIPLE_SELECTION@(format [wp_range@](#) range, nCols)

Method [this.set_multiple_selection@](#)(format wp_range@ range, nCols)

Arguments range The bounding range of the selection
nCols The number of columns.

Description Creates an intra-table ("multiple") selection. The bounding range *and* the number of columns selected must be specified.

WP_SET_OVERTYPE_MODE@

Turns on or off overtype mode

Format WP_SET_OVERTYPE_MODE@(switch)

Method [this.set_overtypemode@](#)(switch)

Arguments switch A Boolean value which if set to TRUE places Words into its overtype mode.

Description Turns on or off overtype mode. If WP_SET_OVERTYPE_MODE@ is turned off insert mode is used.

WP_SET_PAGE_BOUNDS_DISPLAY@

Turns page boundary display on or off

Format WP_SET_PAGE_BOUNDS_DISPLAY@(flag)

Method [this.set_page_bounds_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE indicates that page boundaries are displayed.

Description Controls the display of page boundaries. If the following macro is set to TRUE, this macro will have no effect on page boundary display:
This macro changes the display state for the current file; it does not change the Applixware global display state for showing these boundaries.

See also [WP SET ALL BOUNDS DISPLAY@](#)
[WP TOGGLE PAGE BOUNDS DISPLAY@](#)

WP_SET_PAGE_NUMBERING_FORMAT@

Sets a section's page numbering style

Format WP_SET_PAGE_NUMBERING_FORMAT@(pageNumStyle, restartPageNum, restartPageNumVal, separator, sectNumStyle, sectNumPolicy, restartSectNumVal)

Method [this.set_page_numbering_format@\(pageNumStyle, restartPageNum, restartPageNumVal, separator, sectNumStyle, sectNumPolicy, restartSectNumVal\)](#)

Arguments pageNumStyle
One of the following predefined constants:

WP#NUMBERING#NONE
-1
WP#NUMBERING#ARABIC
0 /* 1, 2, 3... */
WP#NUMBERING#LETTER#UPPER
1 /* A, B, C... */
WP#NUMBERING#LETTER#LOWER
2 /* a, b, c... */

WP#NUMBERING#ROMAN#UPPER
 3 /* I, II, III... */
 WP#NUMBERING#ROMAN#LOWER
 4 /* i, ii, iii... */

restartPageNum

A Boolean value, which if TRUE means that page numbers are restarted in this section; FALSE means that the page numbers are incremented from the last page of the previous section.

restartPageNumVal

If restartPageNum is set, the page numbers begin with this value.

separator The separator used between a section value and a page value.

sectNumStyle

The numbering style used for section numbers. This is one of the same constants as are used for pageNumStyle.

sectNumPolicy

A constant defining the relationship of section numbers in this section with section numbers in the previous section. This can be one of the following values:

WP#SECT#NUM#SAME	0
WP#SECT#NUM#INCREM	1
WP#SECT#NUM#RESTART	2

restartSectNumVal

If sectNumPolicy is set to WP#SECT#NUM#RESTART, the section number begins with this value.

Description Sets the page numbering format for all pages in the current section. If any parameter is set to NULL, that parameter is not changed.

See also [WP_GET_PAGE_NUMBERING_FORMAT@](#)

WP_SET_PARA_BULLET@

Sets the current bullet

Format flag = WP_SET_PARA_BULLET@(format [wp_range@](#) range, format arrayof wp_bead@ newBeads, ifBulletPolicy, ifNoBulletPolicy, latinBullets, spacePolicy)

Method [this.set_para_bullet@](#)(format wp_range@ range, format arrayof wp_bead@ newBeads, ifBulletPolicy, ifNoBulletPolicy, latinBullets, spacePolicy)

Arguments range The range in which to look for the bullet.

newBeads The beads to add or replace.

ifBulletPolicy Indicates the action performed if the selected paragraphs already have bullets. This is one of the following values:

- 0 WP_BULLET_NO_CHANGE
- 2 WP_BULLET_REPLACE
- 3 WP_BULLET_DELETE

ifNoBulletPolicy

Indicates the action performed if the selected paragraphs do not have bullets. This is one of the following values:

- 0 WP_BULLET_NO_CHANGE
- 1 WP_BULLET_ADD

latinBullets Used to determine which ASCII characters are bullets.

spacePolicy Indicates how trailing and leading whitespace is treated. This is one of the following values:

- 0 WP_BULLET_NO_TRAILING
- 1 WP_BULLET_SPACES
- 2 WP_BULLET_SPACES_OR_TAB

Description Examines the passed range (which should represent an entire paragraph). If it has a bullet, it is deleted or replaced with the passed bullet. If it does not have a bullet, one is added. This function returns TRUE if a bead is added or replaced; otherwise FALSE is returned. Ideally, the caller will use the [WP_GET_RANGE_OF_PARA@](#) function to generate the passed range for the paragraph.

NOTE: Any bead that falls partially within the selection is completely examined. Do not try to restrict the examined region to partial beads.

NOTE: This function alters the current selection and does not restore it.

See also [WP_GET_PARA_BULLET@](#)

WP_SET_PARA_FIRST_INDENT@

Sets the paragraph's first line indent

Format WP_SET_PARA_FIRST_INDENT@(milwidth)

Method [this.set_para_first_indent@\(milwidth\)](#)

Arguments milwidth The new first line indentation in mils. (1000 mils = 1 inch)

Description Sets the amount of additional space the selected or current paragraph is indented (or outdented) from the normal paragraph margin.

WP_SET_PARA_LEFT_INDENT@

Sets the paragraph's left margin

Format WP_SET_PARA_LEFT_INDENT@(milwidth)

Method [this.set_para_left_indent@\(milwidth\)](#)

Arguments milwidth The paragraph's left offset in mils. (1000 mils = 1 inch)

Description Sets the amount of space the selected or current paragraph is indented from the page's left margin.

WP_SET_PARA_NUMBERING@

Inserts paragraph numbers

Format WP_SET_PARA_NUMBERING@(format [wp_range@](#) range, format wp_bead@ numberBead, prefix, suffix, trailingWS, ifNumPolicy, ifNoNumPolicy)

Method [this.set_para_numbering@\(format wp_range@ range, format wp_bead@ numberBead, prefix, suffix, trailingWS, ifNumPolicy, ifNoNumPolicy\)](#)

Arguments range The range in which to look for the number.
numberBead The bead with a new number to insert or add.
prefix The prefix to look for in paragraphs with existing numbers.
suffix The suffix to look for in paragraphs with existing numbers.
trailingWS If set to TRUE, the whitespace following an existing number is replaced/deleted along with number.
ifNumPolicy Indicates the action performed if the selected paragraphs already have numbers. This is one of the following values:
 0 WP_NUMBERING_NO_CHANGE
 2 WP_NUMBERING_REPLACE
 3 WP_NUMBERING_DELETE
ifNoNumPolicy Indicates the action performed if the selected paragraphs do not have numbers. This is one of the following values:
 0 WP_NUMBERING_NO_CHANGE
 1 WP_NUMBERING_ADD

Description Examines the passed range (which should represent an entire paragraph). If it has a number, it is deleted or replaced with the passed number. If it does not

have a number, one may be added. This function returns TRUE if a bead is added or replaced; otherwise FALSE is returned.

NOTE: Any bead that falls partially within the selection is completely examined. Do not try to restrict the examined region to partial beads.

NOTE: This function alters the current selection and does not restore it.

See also [WP_GET_PARA_NUMBERING@](#)

WP_SET_PARA_RIGHT_INDENT@

Sets the paragraph's right margin

Format WP_SET_PARA_RIGHT_INDENT@(milwidth)

Method [this.set_para_right_indent@\(milwidth\)](#)

Arguments milwidth The paragraph's right offset in mils. (1000 mils = 1 inch)

Description Sets the amount of space the current or selected paragraph is indented from the page's right margin.

WP_SET_PARA_SETTINGS@

Sets the paragraph attributes

Format WP_SET_PARA_SETTINGS@(format [wp_para_attrs@](#) attrs)

Method [this.set_para_settings@\(format wp_para_attrs@ attrs\)](#)

Arguments attrs The paragraph attributes you want to set.

Description This function sets the paragraph attributes for all selected paragraphs. If an element is NULL, the value is left unchanged.

See also [WP_GET_PARA_SETTINGS@](#)

WP_SET_PRINT_CLASS@

Sets the printer class: PostScript or PCL5

Format WP_SET_PRINT_CLASS@(name)

Method [this.set_print_class@\(name\)](#)

Arguments name One of the following string values:

PCL5
PostScript

Description Tells Applixware Words which class of printing device will be used to print the file. The value sent to this macro determines the way in which Applixware will format the Words document to be printed.

WP_SET_READ_ONLY@

Sets the "read-only" flag

Format WP_SET_READ_ONLY@(flag)

Method [this.set_read_only@\(flag\)](#)

Arguments flag Set to TRUE for read only mode, FALSE for normal mode.

Description Sets the "read-only" flag. When in *read only* mode, you cannot save the current file to the same name. This means that the File ® Save command is disabled. This command will not violate operating system file permissions.

See also [WP_GET_READ_ONLY@](#)

WP_SET_ROW_ALIGNMENT@

Sets row alignment

Format WP_SET_ROW_ALIGNMENT@(row, align)

Method [this.set_row_alignment@\(row, align\)](#)

Arguments row A Row Start bead.
align The type of alignment. Specify one of the following values:
WP#ROW#ALIGN#LEFT
WP#ROW#ALIGN#CENTER
WP#ROW#ALIGN#RIGHT

Description Sets the row alignment for the passed rows. If an element is NULL, it is not changed.

WP_SET_ROW_ATTRS@

Sets row attributes

Format WP_SET_ROW_ATTRS@(row, format [wp_row_attrs@](#) rowAttrs)

Method [this.set_row_attrs@](#)(row, format wp_row_attrs@ rowAttrs)

Arguments row A Row Start bead.
rowAttrs The row attributes.

Description Sets the row attributes. If an item is NULL in rowAttrs, this macro does not change its current value.

WP_SET_ROW_HEADING@

Sets a heading row for a table

Format WP_SET_ROW_HEADING@(rowStartBeadNum, tableHeadingFlag)

Method [this.set_row_heading@](#)(rowStartBeadNum, tableHeadingFlag)

Arguments rowStartBeadNum
 The bead number of the "row start" bead preceding the row that is being named as a heading.
tableHeadingFlag
 A Boolean value indicating that row is a table's heading row.

Description Indicates that the designated row will become a table heading for the table. This macro simply sets the heading element of a WP_ROW_ATTRS@ format to the value of tableHeadingFlag. For more information, see [WP SET ROW ATTRS@](#).

You can use the [WP FIND VISIBLE BEAD@](#) macro to locate the row start bead. For example:

```
WP_FIND_VISIBLE_BEAD@(range.start.bead, WP_ROW_START_CLASS,  
                          true, range.last.bead)
```

WP_SET_ROW_HEIGHT@

Sets the row's height

Format WP_SET_ROW_HEIGHT@(row, height)

Method [this.set_row_height@](#)(row, height)

Arguments row A Row Start bead.
 height The row height in mils. (1000 mils = 1 inch)

Description Sets the row height for the passed row.

WP_SET_ROW_HEIGHT_TYPE@

Sets the row's height type

Format WP_SET_ROW_HEIGHT_TYPE@(row, heightType)

Method [this.set_row_height_type@](#)(row, heightType)

Arguments row A Row Start bead.
 height The row height type. Enter one of the following values:
 WP#ROW#HEIGHT#AUTO
 WP#ROW#HEIGHT#EXACT
 WP#ROW#HEIGHT#MIN
 WP#ROW#HEIGHT#MAX

Description This function sets the manner in which a row's height is adjusted.

WP_SET_ROW_INDENT@

Sets the row's indentation

Format WP_SET_ROW_INDENT@(row, indent)

Method [this.set_row_indent@](#)(row, indent)

Arguments row A Row Start bead.
 indent The indentation in mils. (1000 mils = 1 inch)

Description Sets the row indentation.

WP_SET_ROW_MARGINS@

Sets the row's margins

Format WP_SET_ROW_MARGINS@(row, top, bottom)

Method [this.set_row_margins@](#)(row, top, bottom)

Arguments row A Row Start bead.

top The top margin in mils. (1000 mils = 1 inch)
bottom The bottom margin in mils.

Description Sets the row's top and bottom margins.

WP_SET_RULER_DECIMAL_TAB_CHAR@

Sets the decimal tab character

Format WP_SET_RULER_DECIMAL_TAB_CHAR@(char)

Method [this.set_ruler_decimal_tab_char@\(char\)](#)

Arguments char The decimal character.

Description Set the decimal character for decimal tabs that are inserted using the ruler.

WP_SET_RULER_DISPLAY@

Turns ruler display on or off

Format WP_SET_RULER_DISPLAY@(flag)

Method [this.set_ruler_display@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE tells this macro to display the ruler.

Description Sets the ruler display to be on or off.

This macro changes the display state for the current file; it does not change the Applixware global display state for showing the ruler.

See also [WP TOGGLE RULER DISPLAY@](#)

WP_SET_SECTION_GUTTER@

Sets the inter-column spacing

Format WP_SET_SECTION_GUTTER@(milwidth)

Method [this.set_section_gutter@\(milwidth\)](#)

Arguments milwidth The gutter width in mils. (1000 mils = 1 inch)

Description Sets the amount of space separating columns for the current selection.

WP_SET_SECTION_LEFT_MARGIN@

Sets the section's left margin

Format WP_SET_SECTION_LEFT_MARGIN@(milpos)

Method [this.set_section_left_margin@\(milpos\)](#)

Arguments milpos The new left margin in mils. (1000 mils = 1 inch)

Description Changes the left margin in the current selection to milpos.

WP_SET_SECTION_PAGE_SETUP@

Sets section's page setup attributes

Format WP_SET_SECTION_PAGE_SETUP@(section, format [wp_page_setup@](#)
pageSetup)

Method [this.set_section_page_setup@\(section, format wp_page_setup@ pageSetup\)](#)

Arguments section The section's section bead.

pageSetup The page setup attributes being changed.

Description Changes the following section attributes: page orientation, width, height, and top, bottom, right, left, and binding margins. If an element in the pageSetup variable is NULL, this macro will not change the existing value.

See also [WP_GET_SECTION_PAGE_SETUP@](#)

WP_SET_SECTION_RIGHT_MARGIN@

Sets the right margin

Format WP_SET_SECTION_RIGHT_MARGIN@(milpos)

Method [this.set_section_right_margin@\(milpos\)](#)

Arguments milpos The new right margin in mils. (1000 mils = 1 inch)

Description Changes the right margin of all sections in the current selection to milpos.

WP_SET_SELECTED_CELL_ATTRS@

Sets cell attributes

Format WP_SET_SELECTED_CELL_ATTRS@(format [wp_cell_attrs@](#) cellAttrs)

Method [this.set_selected_cell_attrs@](#)(format wp_cell_attrs@ cellAttrs)

Arguments cellAttrs The new cell attributes.

Description Sets the cell attributes in the current selection.

See also [WP_GET_SELECTED_CELL_ATTRS@](#)

WP_SET_SELECTED_CELL_MARGINS@

Sets margins of selected cells

Format WP_SET_SELECTED_CELL_MARGINS@(left, right, top, bottom)

Method [this.set_selected_cell_margins@](#)(left, right, top, bottom)

Arguments left The left cell margin in mils. (1000 mils = 1 inch)
right The right cell margin in mils.
top The top cell margin in mils.
bottom The bottom cell margin in mils.

Description Sets the left, right, top, and bottom cell margins of all selected cells. See [WP_SET_SELECTED_CELL_ATTRS@](#) for more information.

WP_SET_SELECTED_CELL_V_ALIGN@

Sets the selected cell's vertical alignment

Format WP_SET_SELECTED_CELL_V_ALIGN@(verticalAlignment)

Method [this.set_selected_cell_v_align@](#)(verticalAlignment)

Arguments verticalAlignment
One of the following values:
WP#VERT#ALIGN#TOP
WP#VERT#ALIGN#CENTER
WP#VERT#ALIGN#BOTTOM
WP#VERT#ALIGN#JUSTIFIED

Description Sets the alignment of the information with the selected cells to one of the following:

- Top aligned
- Center aligned
- Bottom aligned
- Justified

WP_SET_SELECTED_CELL_WIDTH@

Sets cell widths

Format WP_SET_SELECTED_CELL_WIDTH@(width)

Method [this.set_selected_cell_width@\(width\)](#)

Arguments width The cell's width in mils. (1000 mils = 1 inch)

Description Sets the width of all selected cells to width. For more information, see, [WP_SET_SELECTED_CELL_ATTRS@](#).

WP_SET_SELECTED_FORM_ATTRS@

Sets cell attributes in current selection

Format WP_SET_SELECTED_FORM_ATTRS@(format [wp_cell_attrs@](#) cellAttrs, locked, unlocked)

Method [this.set_selected_form_attrs@\(format wp_cell_attrs@ cellAttrs, locked, unlocked\)](#)

Arguments cellAttrs The new cell attributes.
locked A Boolean value, where TRUE means set the cell values for locked cells.
unlocked A Boolean value, where TRUE means set the cell values for unlocked cells.

Description Sets the cell attributes of selected cells in a form document.

See also [WP_GET_SELECTED_FORM_ATTRS@](#)

WP_SET_SELECTED_ROW_ALIGNMENT@

Sets row alignment

Format WP_SET_SELECTED_ROW_ALIGNMENT@(align)

Method [this.set_selected_row_alignment@\(align\)](#)

Arguments align The cell's alignment, as follows:
WP#ROW#ALIGN#LEFT
WP#ROW#ALIGN#CENTER
WP#ROW#ALIGN#RIGHT

Description Sets the alignment of all selected rows to align.

See also [WP_SET_SELECTED_ROW_ATTRS@](#)

WP_SET_SELECTED_ROW_ATTRS@

Sets row attributes in current selection

Format WP_SET_SELECTED_ROW_ATTRS@(format [wp_row_attrs@](#) rowAttrs)

Method [this.set_selected_row_attrs@\(format wp_row_attrs@ rowAttrs\)](#)

Arguments rowAttrs The row attributes.

Description Sets the row attributes in a current selection. If an attribute in rowAttrs is NULL, it is not changed.

See also [WP_GET_SELECTED_ROW_ATTRS@](#)

WP_SET_SELECTED_ROW_HEADING@

Sets a table heading

Format WP_SET_SELECTED_ROW_HEADING@(tableHeadingFlag)

Method [this.set_selected_row_heading@\(tableHeadingFlag\)](#)

Arguments tableHeadingFlag
A Boolean value indicating that the selected row is a table's heading row.

Description Indicates that the selected row is a table heading for the selected rows. This macro simply sets the heading element of a WP_ROW_ATTRS@ format to the

value of tableHeadingFlag. See [WP SET SELECTED ROW ATTRS@](#) for more information.

WP_SET_SELECTED_ROW_HEIGHT@

Sets row height

Format WP_SET_SELECTED_ROW_HEIGHT@(height)

Method [this.set_selected_row_height@\(height\)](#)

Arguments height The cell's height in mils. (1000 mils = 1 inch)

Description Sets the height of all selected rows to height. For more information, see, [WP SET SELECTED ROW ATTRS@](#).

WP_SET_SELECTED_ROW_HT_TYPE@

Sets row height type

Format WP_SET_SELECTED_ROW_HT_TYPE@ (heightType)

Method [this.set_selected_row_ht_type@ \(heightType\)](#)

Arguments heightType The cell's height type, as follows:

WP#ROW#HEIGHT#AUTO
WP#ROW#HEIGHT#EXACT
WP#ROW#HEIGHT#MIN
WP#ROW#HEIGHT#MAX

Description Sets the height type of all selected rows to heightType.

See also [WP SET SELECTED ROW ATTRS@](#)

WP_SET_SELECTED_ROW_INDENT@

Sets row indent

Format WP_SET_SELECTED_ROW_INDENT@(indent)

Method [this.set_selected_row_indent@\(indent\)](#)

Arguments indent The cell's indent in mils. (1000 mils = 1 inch)

Description Sets the indentation of all selected rows to indent.

See also [WP SET SELECTED ROW ATTRS@](#)

WP_SET_SELECTED_ROW_MARGINS@

Sets row margins

Format WP_SET_SELECTED_ROW_MARGINS@(top, bottom)

Method [this.set_selected_row_margins@\(top, bottom\)](#)

Arguments top The cell's top margin in mils. (1000 mils = 1 inch)
bottom The cell's bottom margin in mils.

Description Sets the top and bottom margins of all selected rows to top and bottom.

See also [WP SET SELECTED ROW ATTRS@](#)

WP_SET_SELECTION_MARKS@

Surrounds current selection with marker beads

Format WP_SET_SELECTION_MARKS@(name, rowsFlag, includeCellEndFlag, markFootnotesFlag, markHdrFtrsFlag)

Method [this.set_selection_marks@\(name, rowsFlag, includeCellEndFlag, markFootnotesFlag, markHdrFtrsFlag\)](#)

Arguments name A unique name for the markers.
rowsFlag Set to TRUE if you want selection in chunks of rows. Set to FALSE if you want individual cells.
includeCellEndFlag
 Set to TRUE to include the Cell End marker for individual cells in the range.
markFootnotesFlag
 Set to TRUE if you want footnote references in the selection to have marks also.
markHdrFtrsFlag
 Set to TRUE if you want headers and footers within the selection to have marks also. That is, a header or footer with section beads in the selection should also get marks.

Description Surrounds the current selection with marker beads. WP_SET_SELECTION_MARKS@ handles the "multiple" (intra-table) case properly and adjusts the

Words cursor appropriately. This macro assumes that the previous selection markers have been deleted.

See also [WP_GET_SELECTION_MARKS@](#)

WP_SET_SESSION_VAR@

Sets a session variable

Format WP_SET_SESSION_VAR@(name, value)

Method [this.set_session_var@\(name, value\)](#)

Arguments name The name of the document variable.
value The value of the document variable.

Description Sets the session variable name to value.

See also [WP_GET_SESSION_VAR@](#)

WP_SET_SKIP_DICT@

Names the skip dictionary

Format WP_SET_SKIP_DICT@(filename)

Method [this.set_skip_dict@\(filename\)](#)

Arguments filename The skip dictionary's file name.

Description Sets the skip dictionary. (This dictionary contains a list of words that the spelling checker should ignore.)

See also [WP_GET_SKIP_DICT@](#)

WP_SET_STATUS_LINE@

Writes a message into the status line

Format WP_SET_STATUS_LINE@(message)

Method [this.set_status_line@\(message\)](#)

Arguments message A string containing the message to be displayed.

Description Writes a message to the status line. After this message is displayed, Applixware Words will no longer write its "normal" status information to the status line. To resume normal status messages, use the following macro:

WP_CLEAR_STATUS_LINE@

See also **WP_SET_STATUS_LINE_TEMP@**

WP_SET_STATUS_LINE_TEMP@

Writes status line message that is overwritten by next normal message

Format WP_SET_STATUS_LINE_TEMP@(message)

Method [this.set_status_line_temp@\(message\)](#)

Arguments message The text written to the status line.

Description Writes status line message that is overwritten by the next normal message. Contrast the function of this macro with that of the **WP_SET_STATUS_LINE@** which sets the status line in a way that normal messages do not overwrite message.

WP_SET_STYLE@

Sets the style

Format WP_SET_STYLE@(format wp_style@ style, noReformatFlag)

Method [this.set_style@\(format wp_style@ style, noReformatFlag\)](#)

Arguments style The new or edited style.

noReformatFlag

If set to TRUE, it indicates that there are more styles to come, so do not reformat the document as a consequence of the style change.

Description Sets the attributes of a style.

In most cases, you will retrieve the current style's attributes using the **WP_GET_STYLE@** macro, alter a few attributes, then define the style using this macro.

WP_SET_TABLE_CELLS_WIDTH@

Sets cell widths

Format WP_SET_TABLE_CELLS_WIDTH@(colNumber, milwidth)

Method this.set_table_cells_width@(colNumber, milwidth)

Arguments colNumber zero-based cell number
milwidth The new width in mils. (1000 mils = 1 inch)

Description Changes the width of cells in a column. The current selection determines the cells rows.

For example, assume you have a table with four row and three columns and have selected three of the four cells in columns 2, leaving the last cell in the column unselected.

You now execute WP_SET_TABLE_CELLS_WIDTH@(0, 1500). The three cells in the first column are changed to 1 1/2 inch. The fourth cell in the first column is unchanged

WP_SET_TABLE_ROWS_INDENT@

Sets row indentation

Format WP_SET_TABLE_ROWS_INDENT@(milpos)

Method this.set_table_rows_indent@(milpos)

Arguments milpos The new row indentation in mils. (1000 mils = 1 inch)

Description Sets the left indentation of each selected row.

WP_SET_TABSTOPS@

Sets tab stops

Format WP_SET_TABSTOPS@(format arrayof [wp_tab@](#) tabs)

Method this.set_tabstops@(format arrayof wp_tab@ tabs)

Arguments tabs An array containing information about each of the paragraph's tab stops.

Description Sets the tabs for all the paragraphs in the selection.

WP_SET_TEXT_ATTRS@

Sets the text attributes

Format WP_SET_TEXT_ATTRS@(format wp_text_attrs@ att)

Method [this.set_text_attrs@](#)(format wp_text_attrs@ att)

Arguments att The text attributes.

Description Sets the text attributes of the selected material. If an attribute within the att structure is set to NULL, it is not changed.

See also [WP_GET_TEXT_ATTRS@](#)

WP_SET_TEXT_POINTER@

Sets text pointer shape

Format WP_SET_TEXT_POINTER@(pointerName)

Method [this.set_text_pointer@](#)(pointerName)

Arguments pointerName The name of the new text pointer.

Description Sets the new text pointer shape. This is the pointer that is used within the Words window's margins. After using this macro, the text pointer is changed for all Words windows.

If you wish to always use this pointer, it can be made permanent by setting a preference.

To obtain a list of available shape names, use [GET_CURSOR_LIST@](#).

See also [WP_SET_DEFAULT_POINTER@](#)

WP_SET_USER_DICT@

Sets the user dictionary

Format WP_SET_USER_DICT@(filename)

Method [this.set_user_dict@](#)(filename)

Arguments filename The file name of the new user dictionary.

Description Sets the name of the user's dictionary.

See also [WP_GET_USER_DICT@](#)

WP_SET_VIEW_CHAR_ACCURATE@

Sets Applixware Words into a WYSIWYG state

Format WP_SET_VIEW_CHAR_ACCURATE@()

Method [this.set_view_char_accurate@\(\)](#)

Description Places Applixware into a WYSIWYG (What You See Is What You Get) state where, given the difference between screen display and the output resolution, the information displayed on the screen very closely approximates the printed result. On some systems, this state will display slightly slower than that into which your system is placed when [WP_SET_VIEW_LINE_ACCURATE@](#) is chosen.

WP_SET_VIEW_LINE_ACCURATE@

All information is on the correct line

Format WP_SET_VIEW_LINE_ACCURATE@()

Method [this.set_view_line_accurate@\(\)](#)

Description Places Applixware into a near-WYSIWYG (What You See Is What You Get) state where information is placed on the correct line but character widths may differ slightly from what they would actually be when printed. On some systems, this state will display slightly faster than that into which your system is placed when [WP_SET_VIEW_CHAR_ACCURATE@](#) is chosen.

WP_SET_VIEW_ONLY@

Changes the view-only state

Format WP_SET_VIEW_ONLY@(flag)

Method [this.set_view_only@\(flag\)](#)

Arguments flag A Boolean value which if set to TRUE indicates that the current Applixware window is in a *view-only* state.

Description Changes the Applixware Words *view-only* state. When this state is turned on, no changes can be made to the information displayed on the screen.

See also [WP_GET_VIEW_ONLY@](#)

WP_SET_VIEW_STATE@

Sets viewing state

Format WP_SET_VIEW_STATE@(format wp_view_state@ info)

Method [this.set_view_state@](#)(format wp_view_state@ info)

Arguments info A variable containing the new viewing state.

Description Sets the visibility/invisibility of most Words optional viewing parameters. The info parameter to this macro has the following format:

```
format wp_view_state@
    pilcros,
    ruler,
    breaks,
    fieldMethods,
    allBounds,
    cellBounds,
    frameBounds,
    hdrFtrBounds,
    pageBounds,
    zoomFactor,      'percentage zoom
    wysiwyg,         '0-line positioning
                    '1-character positioning
    graphics         'if false, use placeholder for graphics
```

See also [WP_GET_VIEW_STATE@](#)

WP_SET_WINDOW_TITLE@

Sets window and icon title

Format WP_SET_WINDOW_TITLE@(windowTitle, iconTitle)

Method [this.set_window_title@](#)(windowTitle, iconTitle)

Arguments windowTitle The text to which the window's title will be set.

iconTitle The text to be displayed when the window is made into an icon.

Description Defines the text that is displayed as the Words Window's title and the text that is displayed when this window is minimized into an icon.

WP_SHADING@

Apply shading to paragraphs and cells

Format WP_SHADING@(shadingStyle, fgColor, bgColor)

Method [this.shading@](#)(shadingStyle, fgColor, bgColor)

Arguments shadingStyle The border's shading style. This is the border shadingStyle component of a wp_border_attrs@ format. This is a number from 0 to 18, representing one of the nineteen predefined shading styles.

fgColor The string name of the foreground color; e.g., *black*.

bgColor The string name of the background color; e.g., *white*.

Description Apply shading to paragraphs or cells depending on the current selection or cursor.

WP_SHIFT@

Shifts selection in indicated direction

Format flag = WP_SHIFT@(dir)

Method [flag = this.shift@](#)(dir)

Arguments dir One of the following directions:

0	WP_SHIFT_UP
1	WP_SHIFT_LEFT
2	WP_SHIFT_RIGHT
3	WP_SHIFT_DOWN

Description Shifts selection in indicated direction. Paragraphs and rows can go up and down. Columns and cells can go left and right.

This macro returns TRUE if the request is legal (and was accomplished); otherwise, it returns FALSE.

WP_SHIFT_CASE@

Switches the case of the selected text

Format WP_SHIFT_CASE@()

Method [this.shift_case@](#)

Description Switches the case of the selected text (aaa ® AAA ® Aaa). Only ISO-LATIN-1 text is altered. The first two ISO-LATIN-1 characters in the range determine the "current state" of the text. That is, Symbol and Dingbats characters are not altered.

WP_SHIFT_CASE_IN_RANGE@

Shifts the case in a range

Format WP_SHIFT_CASE_IN_RANGE@(format [wp_range@](#) range)

[Method](#) this.shift_case_in_range@(format wp_range@ range)

Arguments range A range within the document.

Description Switches the case of the selected text (aaa ® AAA ® Aaa). Only ISO-LATIN-1 text is altered. The first two ISO-LATIN-1 characters in the range determine the "current state" of the text. That is, Symbol and Dingbats characters are not altered.

If range is NULL, this macro shifts the case of the currently selected text.

WP_SHIFT_CELLS@

Shifts a set of cells

Format WP_SHIFT_CELLS@(cellBead1, nCells1, cellBead2, beforeFlag)

[Method](#) this.shift_cells@(cellBead1, nCells1, cellBead2, beforeFlag)

Arguments cellBead1 Bead number of Cell End bead in the first set.
nCells1 The number of beads in the first set.
cellBead2 The bead number of the target Cell End bead.
beforeFlag If set to TRUE, moves cells before the indicated cell. If set to FALSE the cells are moved after the indicated cell.

Description Shifts a set of cells. WP_SHIFT_CELLS@ moves one or more adjacent cells before or after some other cell (in the same or different row). The target cell should not be in the set of moving cells. If the operation leaves a row without cells, the row is deleted.

WP_SHIFT_DOWN@

Shifts paragraphs and rows down

Format WP_SHIFT_DOWN@()

Method [this.shift_down@](#)

Description Moves paragraphs and rows down one row or paragraph.

See also [WP_SHIFT@](#)

WP_SHIFT_INDENTS@

Moves selected paragraphs' first and left indents

Format WP_SHIFT_INDENTS@(shift)

Method [this.shift_indents@\(shift\)](#)

Arguments shift The number of mils to shift the selected paragraphs. (1000 mils = 1 inch)

Description Adds shift mils to the first and left indents of all paragraph's in the current selection.

WP_SHIFT_LEFT@

Shifts columns and cells left

Format WP_SHIFT_LEFT@()

Method [this.shift_left@](#)

Description Moves columns and cells left.

See also [WP_SHIFT@](#)

WP_SHIFT_NAME@

Renames a file

Format WP_SHIFT_NAME@(newName)

Method [this.shift_name@](#)(newName)

Arguments newName The current file's new name.

Description Renames a file in the simplest way. The details of file renaming will be handled internally by [WP_RENAME@](#).

See also [WP_SAVE_RENAME@](#)

WP_SHIFT_RIGHT@

Shifts columns and cells right

Format WP_SHIFT_RIGHT@()

Method [this.shift_right@](#)

Description Moves columns and cells right.

See also [WP_SHIFT@](#)

WP_SHIFT_SELECTED@

Moves the currently selected range

Format WP_SHIFT_SELECTED@(target, beforeFlag)

Method [this.shift_selected@](#)(target, beforeFlag)

Arguments target Cell, row, or paragraph bead number.
beforeFlag If set to TRUE, moves item before the indicated target. If set to FALSE, the item is moved after the indicated target.

Description Moves the currently selected range before or after the indicated item (cell, row, or paragraph). Rows and paragraphs may be freely shifted, but cells can only be shifted before or after other cells. After you exit from this function, the shifted items are still selected.

WP_SHIFT_UP@

Shifts paragraphs and rows up

Format WP_SHIFT_UP@()

Method [this.shift_up@](#)

Description Moves paragraphs and rows up.

See also [WP_SHIFT@](#)

WP_SORT@

Sorts paragraphs within a range

Format WP_SORT@(sortOrder, sortCaseFlag, sortType, sortUpon, sortKey1, sortKey2, sortSep)

Method [this.sort@](#)(sortOrder, sortCaseFlag, sortType, sortUpon, sortKey1, sortKey2, sortSep)

Arguments

sortOrder	Indicates ascending or descending order. Use one of the following values: 0 WP_SORT_ASCENDING 1 WP_SORT_DESCENDING
sortCaseFlag	If set to TRUE, it is case sensitive. If it is set to FALSE, it is not case sensitive.
sortType	Indicates whether is it an alphanumeric or numeric sort. Choose one of the following values: 0 WP_SORT_ALPHANUM 1 WP_SORT_NUMERIC
sortUpon	Specifies the sort criteria. Choose one of the following values: 0 WP_SORT_ROWS 1 WP_SORT_PARTIAL_ROWS 2 WP_SORT_CELLS_IN_ROW 3 WP_SORT_COLS 4 WP_SORT_CELLS_IN_COL
sortKey1	Primary sort field.
sortKey2	Secondary sort field. Set this to zero if there is no secondary key.

sortSep The text that separates the fields.

Description Sorts paragraphs within the selected range. If a numeric sort is being performed and if the paragraph does not contain a proper number, then Words sorts the paragraph as if it contained an extremely small number.

WP_SPECIAL_CHARS@

Invokes the Special Characters dialog box

Format WP_SPECIAL_CHARS@()

Method [this.special_chars@](#)

WP_SPELLCHECK_RANGE@

Spellchecks the passed range

Format WP_SPELLCHECK_RANGE@(format [wp_range@](#) range, mask, dicts, format [wp_location@](#) loc)

Method [this.spellcheck_range@](#)(format [wp_range@](#) range, mask, dicts, format [wp_location@](#) loc)

Arguments

range	The bead range that you want spell checked.
mask	A string containing a bit pattern. This bit pattern is used to determine the types of errors that are detected by the spell checker. If you set this argument to NULL, Words uses your Spell Checker preferences to determine which errors to detect.
dicts	An array containing a list of the dictionaries that are inspected.
loc	The location at which spell checking will resume. This position is after the located error.

Description Returns the error code of the first error found. For example, if the first error found is a mis-spelled word, the error returned would be 11.

The text containing the error is highlighted in the Words window. The mask argument contains a string with a bit pattern. For example, the following string would return the first repeated word:

mask = "000001" ' Bit 5 turns on repeated word detection

If the mask argument is set to NULL, WP_SPELLCHECK_RANGE@ returns the errors indicated in your Spellchecker preferences.

The following list shows the type of errors that can be returned, and the position of the bit that controls whether or not that error is detected:

0	Unused
1	Unused
2	Unused
3	Unused
4	Unused
5	Repeated Word
6	Unused
7	A number
9	A date
10	Not alphabetic (contains symbols)
11	Misspelled
12	Bad Start of sentence (needs cap)
13	A Roman numeral
14	Invalid punctuation prefix
15	Invalid punctuation suffix
16	An outline marker
17	Word contains invalid prefix/suffix
18	Invalid prefix
19	Words attached to invalid prefix/suffix
20	Invalid suffix
21	Text contains parentheses
22	Word with parentheses text is invalid
23	Word without parentheses text is invalid
24	Valid part of a compound word
25	Compound word (German only)
26	Invalid compound word
27	A valid compound word
28	A slash that separates words
29	Word does not need a hyphen
30	Could contract word with the previous word (French and Italian only)
31	Word contains an apostrophe prefix (French and Italian only)
32	Invalid word contains an apostrophe prefix (French and Italian only)
33	A hyphen
34	Invalid hyphen separator
35	Word contains hyphens
36	Invalid word contains hyphens
37	Word is in possessive form
38	Invalid word is in possessive form
39	Apostrophe prefix of a prefixed word (French and Italian only)
40	Invalid apostrophe prefix (French and Italian only)
41	Possessive part of word
42	Invalid possessive part of word
43	Linkage symbol (German and Dutch only)
44	Hyphenated number (e.g., phone number)

WP_SPELL_WORD_GUESS@

Replaces a word with best-guess alternative

Format WP_SPELL_WORD_GUESS@()

Method [this.spell_word_guess@](#)

Description Replaces a word with a best-guess alternative.

WP_SPLIT_BEAD@

Splits a bead

Format WP_SPLIT_BEAD@(format [wp location@](#) loc)

Method [this.split_bead@\(format wp_location@ loc\)](#)

Arguments loc The location of the bead.

Description Splits a bead at a location. An error occurs if the bead is not a text bead, or the location is beyond the last character or at the first character.

WP_SPLIT_CELL@

Splits a table cell

Format WP_SPLIT_CELL@(format [wp location@](#) loc)

Method [this.split_cell@\(format wp_location@ loc\)](#)

Arguments loc The location within the cell at which the cell will be split.

Description Split the indicated table cell at the indicated location. The new cell will have the same attributes as the indicated cell. The indicated location must fall within a cell. This macro reserves the right to move the location a bit if it is after one or more markers. WP_SPLIT_CELL@ *assumes* a "legal" location (one that could represent the selection cursor) is passed to it.

WP_SPLIT_CELL_AT_CURSOR@

Splits the current cell at the cursor location

Format WP_SPLIT_CELL_AT_CURSOR@(flag)

Arguments flag a Boolean. If TRUE, the current cell is split in half. If FALSE, two cells are created from the current cell. Both of the resulting cells will be the same width as the original.

Method [this.split_cell_at_cursor@](#)

Description Splits the cell containing the cursor at the cursor's location. If flag is set to TRUE, the current cell is split in half. If FALSE, two cells are created from the current cell. Both of the resulting cells will be the same width as the original.

WP_SPLIT_CELL_HALF_AT_CURSOR@

Splits the current cell at the cursor location in half

Format WP_SPLIT_CELL_HALF_AT_CURSOR@()

Description Splits the current cell in half.

WP_SPLIT_PARAGRAPH@

Splits a paragraph at the current position

Format WP_SPLIT_PARAGRAPH@()

Method [this.split_paragraph@](#)

Description Divides a paragraph into two separate paragraphs.

WP_SPLIT_ROW@

Splits a row

Format WP_SPLIT_ROW@(bead, beforeFlag)

Method [this.split_row@\(bead, beforeFlag\)](#)

Arguments bead A bead within the row being split.

beforeFlag If set to TRUE, the row is split before the cell indicated by the bead parameter. If set to FALSE, the row is split after the cell.

Description Splits the row containing the passed bead, before or after the passed bead's cell.

WP_SPLIT_ROW_AFTER_CELL@

Splits a row after the current location

Format WP_SPLIT_ROW_AFTER_CELL@()

Method [this.split_row_after_cell@](#)

Description Splits a table after the current cell. This macro calls [WP_SPLIT_SELECTED_ROWS@](#) with its beforeFlag set to FALSE.

WP_SPLIT_ROW_BEFORE_CELL@

Splits a row before the current location

Format WP_SPLIT_ROW_BEFORE_CELL@()

Method [this.split_row_before_cell@](#)

Description Splits a table before the current cell. This macro calls [WP_SPLIT_SELECTED_ROWS@](#) with its beforeFlag set to TRUE.

WP_SPLIT_SELECTED_ROWS@

Splits rows before or after cells

Format WP_SPLIT_SELECTED_ROWS@(beforeFlag)

Method [this.split_selected_rows@\(beforeFlag\)](#)

Arguments beforeFlag If set to TRUE, it is split before the row. If set to FALSE, it is split after the row.

Description Splits rows before or after the current cell. This macro calls [WP_SPLIT_ROW@](#).

WP_SPLIT_SELECTED_TABLE@

Splits selected table

Format WP_SPLIT_SELECTED_TABLE@(beforeFlag)

Method `this.split_selected_table@(beforeFlag)`

Arguments `beforeFlag` If set to TRUE, it is split before the row. If set to FALSE, it is split after the row.

Description Splits the selected table. This macro calls `WP_SPLIT_TABLE@`.

See also [**WP_SPLIT_TABLE@**](#)

WP_SPLIT_TABLE@

Places a paragraph before or after a row

Format `WP_SPLIT_TABLE@(bead, beforeFlag)`

Method `this.split_table@(bead, beforeFlag)`

Arguments `bead` A bead in the row.
`beforeFlag` If set to TRUE, it is split before the row. If set to FALSE, it is split after the row.

Description Places a paragraph before or after the indicated row, thus splitting the table.

WP_SPLIT_TABLE_AFTER_ROW@

Places a paragraph after the current row

Format `WP_SPLIT_TABLE_AFTER_ROW@()`

Method `this.split_table_after_row@`

Description Splits the current table by placing a paragraph after the current row. This macro calls [**WP_SPLIT_SELECTED_TABLE@**](#) with its `beforeFlag` argument set to FALSE.

WP_SPLIT_TABLE_BEFORE_ROW@

Places a paragraph before the current row

Format `WP_SPLIT_TABLE_BEFORE_ROW@()`

Method `this.split_table_before_row@`

Description Splits the current table by placing a paragraph before the current row. This macro calls **WP_SPLIT_SELECTED_TABLE@** with its beforeFlag argument set to TRUE.

WP_STAMP_NEW_FILE@

Adds new file information to document variables

Format WP_STAMP_NEW_FILE@(app, version)

Method [this.stamp_new_file@](#)(app, version)

Arguments app Name of application generating the file. For generic Words applications, pass NULL.
version Version of creating application. This is ignored if app is NULL.

Description Sets the creation date, creation user, and so on, document variables. It should be called for new documents or (perhaps) by user scripts that create documents. In the latter case, the script writer might want to pass a non-NULL "Creation application" variable so these documents can be differentiated from those created by Words.

WP_STATUS_LINE@

Returns status line text

Format statusBuf = WP_STATUS_LINE@()

Method [statusBuf =this.status_line@](#)

Description Returns the text being displayed in the status line.

WP_STYLE_EFFECTIVE_ATTS@

Returns the effective attributes of a style

Format newStyle = WP_STYLE_EFFECTIVE_ATTS@(format wp_style@ style, format wp_style@ parent)

Method [newStyle = this.style_effective_atts@](#)(format wp_style@ style, format wp_style@ parent)

Arguments style The style you are examining.
parent The parent style of the style you are examining.

Description Returns the effective attributes of a style; that is, the attributes that a paragraph of this style would default to. `WP_STYLE_EFFECTIVE_ATT@` takes as input the style in question and its parent style. The parent style should already contain its effective attributes. A new style-formatted variable is returned.

WP_STYLE_INHERIT_ATT@

Compares the attributes of a style and its parent

Format `newStyle = WP_STYLE_INHERIT_ATT@(format wp_style@ style, format wp_style@ parent)`

Method `newStyle = this.style_inherit_atts@(format wp_style@ style, format wp_style@ pare`

Arguments `style` The style you are examining.
`parent` The parent style of the style you are examining.

Description Compares the attributes of a style and its parent; where the attributes are the same, the child's attribute is changed to "inherit". The modified child style is returned. The parent style should already contain its effective attributes.

WP_SWAP_BEADS@

Swaps a pair of ranges

Format `WP_SWAP_BEADS@(format wp_range@ range1, format wp_range@ range2)`

Method `this.swap_beads@(format wp_range@ range1, format wp_range@ range2)`

Arguments `range1` The range you want to swap.
`range2` The range that is being swapped in.

Description Swaps a pair of ranges within the document.
NOTE: Moving the wrong beads to the wrong location will cause severe problems.

WP_SWAP_CELLS@

Swaps a pair or set of cells

Format `WP_SWAP_CELLS@(cellBead1, nCells1, cellBead2, nCells2)`

Method `this.swap_cells@(cellBead1, nCells1, cellBead2, nCells2)`

Arguments cellBead1 The bead number of the Cell End bead in the first set.
nCells1 The number of beads in the first set.
cellBead2 The bead number of the Cell End bead in the second set.
nCells2 The number of beads in the second set.

Description Swaps a pair or set of cells. The sets should not overlap. However, they can be in the same row, or in different rows.

WP_SWAP_TAGGED_ITEMS@

Swaps paragraphs, rows, or cells

Format WP_SWAP_TAGGED_ITEMS@(tag1, tag2, sort, orig_cols)

Method [this.swap_tagged_items@](#)(tag1, tag2, sort, orig_cols)

Arguments tag1 The tag name of one item.
tag2 The tag name of the other item.
sort The scope of the swap.
orig_cols If it is a multi-selection, this argument indicates the number of columns selected.

Description Swaps the paragraphs, rows, or cells indicated by the passed tag names. It is assumed that both tags exist and each points to a (different) Paragraph, Row Start, or Cell End bead (both or neither tags should point to the Cell End beads). The tag locations are also updated during this operation. It is the caller's responsibility to remove the tags after this operation, if desired.

WP_SWAP_TAGGED_ITEMS@ *assumes* that tags always point to the offset zero of the given beads at all times. The tags are indicating bead numbers and not offsets.

WP_TABLE_TO_TEXT@

Converts table text to paragraph text

Format WP_TABLE_TO_TEXT@(sepChar)

Method [this.table_to_text@](#)(sepChar)

Arguments sepChar A one-character string representing the cell-separator character such as a tab or comma. If NULL is used, this macro converts the table to text without separation characters between cell contents.

Description Converts the table to text with or without separation characters between cell contents. It is called by the Table ® Convert Table to Text option.

WP_TABLE_TO_TEXT_QUICK@

Repeats last 'Table-to-Text' operation

Format WP_TABLE_TO_TEXT_QUICK@ ()

Method [this.table_to_text_quick@](#)

Description Converts the table to text using the separation character settings of the pervious 'Table-to-Text' operation. No separation characters are used in the conversion if there is no previous 'Table-to-Text' operation.

WP_TAB_KEY@

Inserts a tab

Format WP_TAB_KEY@ ()

Method [this.tab_key@](#)

Description Inserts a tab at the current location.

WP_TAG_RANGE@

Tags a range

Format WP_TAG_RANGE@ (startTag, endTag, format [wp_range@](#) range)

Method [this.tag_range@](#) (startTag, endTag, format wp_range@ range)

Arguments startTag A unique name for the tag at the start of range.
endTag A unique name for the tag at the end of range.
range The range to tag.

Description Tags (or names) a range by creating a pair of tags that will be associated with locations in the document. (Unlike markers, these do not alter the document.) These tag names must be passed to the following macro when you are retrieving the range:

WP_RETRIEVE_TAGGED_RANGE@

WP_TAG_SELECTION@

Tags the current selection

Format ncols = WP_TAG_SELECTION@(startTag, endTag)

Method ncols = this.tag_selection@(startTag, endTag)

Arguments startTag A unique name for the tag at the start of the current selection.
endTag A unique name for the tag at the end of the current selection.

Description Tags (or names) the current selection by creating a pair of tags that will be associated with locations in the document. (Unlike markers, these do not alter the document.) These tag names must be passed to **WP_RESTORE_TAGGED_SELECTION@** to restore the selection. This macro returns the number of columns currently selected; this also must be passed to the restored function. After calling this function, the selection *must* be restored before this function is called again.

WP_TEMP_SAVE@

Saves current state of the document into a temporary file

Format WP_TEMP_SAVE@(filename)

Method this.temp_save@(filename)

Arguments filename The name of the file into which the temporary save will be made.

Description Saves the current state of the document to filename. This macro is used internally to generate a document for printing, mailing, and the like.

Using this function resets the error handler. If an error occurs while the file is being saved, filename is deleted.

WP_TEXT_*_POINT@

Sets the size of selected text

Format WP_TEXT_*_POINT@()

Method this.text_*_point@

Description Sets the size of selected text in a Words document to 6, 8, 10, 12, 14, 18, 24, or 36 point. If no text is selected, the specified type size is applied to text that is typed at the current location. The following point size macros exist:

WP_TEXT_6_POINT@
WP_TEXT_8_POINT@
WP_TEXT_10_POINT@
WP_TEXT_12_POINT@
WP_TEXT_14_POINT@
WP_TEXT_18_POINT@
WP_TEXT_24_POINT@
WP_TEXT_36_POINT@

WP_TEXT_DICT_ADD_WORD@

Adds a word to the dictionary

Format WP_TEXT_DICT_ADD_WORD@(word, correct, language, dict)

Method [this.text_dict_add_word@](#)(word, correct, language, dict)

Arguments

word	The word you are adding to the dictionary.
correct	The correct spelling of word.
language	The integer dictionary language .
dict	The name of the dictionary you are adding word to.

Description Adds a word to the Proximity dictionary. If word is illegal, an error displays. This macro calls [TEXT_DICT_ADD_WORD@](#).

When the spell checker suggests a replacement word, the replacement word must be phonetically similar to the misspelled word for it to be one of the suggested words.

WP_TEXT_FACE@

Applies a typeface to text

Format WP_TEXT_FACE@(fontName)

Method [this.text_face@](#)(fontName)

Arguments

fontName	The typeface name. Valid typeface strings are: Avant Garde Bookman Courier
----------	---

Gothic
Helvetica
Helvetica Narrow
Minchou
New Century Schoolbook
Palatino
Times
Zapf Chancery
Zapf Dingbats

Description Applies fontName to selected text in a Words document. If no text is selected, this macro specifies the typeface of the text that will next be typed at the cursor's location. The specified typeface becomes the default. If fontName is not a valid typeface, the typeface remains unchanged

WP_TEXT_SIZE@

Sets text size

Format WP_TEXT_SIZE@(size)

Method [this.text_size@](#)(size)

Arguments size The new point size.

Description Sets the selected text to size. If no text is selected, this macro makes size the default point size of the text that you type at the current cursor location.

WP_TEXT_TO_TABLE@

Converts paragraph text to table text

Format WP_TEXT_TO_TABLE@(sepChar)

Method [this.text_to_table@](#)(sepChar)

Arguments sepChar A one-character string representing the cell-separator character such as a tab or comma. If NULL is used, it converts the text to table making each paragraph a one-cell row.

Description Converts paragraph text to table text. It is called by the Table Ý Convert Text to Table option.

WP_TEXT_TO_TABLE_QUICK@

Repeats the last 'Text-to-Table' operation

Format WP_TEXT_TO_TABLE_QUICK@ ()

Method [this.text_to_table_quick@](#)

Description Converts the text to table using the separation character settings of the previous 'Text-to-Table' operation. If there is no previous 'Text-to-Table' operation, the macro converts the text to table by making each paragraph a one-cell row.

WP_TEXT_typeface@

Applies a typeface to text

Format WP_TEXT_typeface@ ()

Method [this.text_typeface@](#)

Description Applies the specified typeface to selected text in a Words document. If no text is selected, this macro specifies the typeface of the text that will next be typed at the cursor's location. The specified typeface becomes the default. The following *typeface* macros are supported:

WP_TEXT_AVANT_GARDE@
WP_TEXT_BOOKMAN@
WP_TEXT_CHANCERY@
WP_TEXT_COURIER@
WP_TEXT_DINGBATS@
WP_TEXT_GOTHIC@
WP_TEXT_HELV_NARROW@
WP_TEXT_HELVETICA@
WP_TEXT_MINCHOU@
WP_TEXT_PALATINO@
WP_TEXT_SCHOOLBOOK@
WP_TEXT_SYMBOL@
WP_TEXT_TIMES@

WP_TOGGLE_ALL_BOUNDS_DISPLAY@

Turns on or off all borders

Format WP_TOGGLE_ALL_BOUNDS_DISPLAY@ ()

Method [this.toggle_all_bounds_display@](#)

Description Displays or suppresses the viewing of all boundaries in a document. Boundaries include all header and footer boundaries, table and cell boundaries, frame boundaries and margins.

Boundaries do not print.

This command sets how boundaries are displayed when the user next opens an Applixware Words window.

WP_TOGGLE_ALL_BOUNDS_DISPLAY@ is called by View ® All Boundaries. For more information, see the *Words* manual.

See also [WP SET ALL BOUNDS DISPLAY@](#)

WP_TOGGLE_BREAKS_DISPLAY@

Turns on or off breaks display

Format WP_TOGGLE_BREAKS_DISPLAY@()

Method [this.toggle_breaks_display@](#)

Description Displays or suppresses the display of page, section, column, and line breaks in a document. Break display characters do not print.

This command sets how breaks are displayed when the user next opens an Applixware Words window.

WP_TOGGLE_BREAKS_DISPLAY@ is called by View ® Breaks. For more information, see the *Words* manual.

See also [WP SET BREAKS DISPLAY@](#)

WP_TOGGLE_CELL_BOUNDS_DISPLAY@

Turns on and off cell boundary viewing

Format WP_TOGGLE_CELL_BOUNDS_DISPLAY@()

Method [this.toggle_cell_bounds_display@](#)

Description Displays or suppresses the display of cell boundaries in a table. Boundaries do not print.

This command sets how cell boundaries are displayed when the user next opens an Applixware Words window.

WP_TOGGLE_CELL_BOUNDS_DISPLAY@ is called by View ® Cell Boundaries. For more information, see the *Words* manual.

See also [WP_SET_CELL_BOUNDS_DISPLAY@](#)

WP_TOGGLE_CONTROLS_DISPLAY@

Turns off or on the viewing of format characters

Format WP_TOGGLE_CONTROLS_DISPLAY@(flag)

Method [this.toggle_controls_display@](#)(flag)

Arguments flag A Boolean value which if set to TRUE indicates that format characters should be displayed.

Description Displays or suppresses the display of format characters. Format characters include:

- Paragraph markers (¶). A paragraph marker marks the beginning of every paragraph in a document. Paragraph markers appear to the right of the given column of text.
- Tab markers (->). Tab markers appear wherever you have pressed the TAB key.

You can turn the viewing of format characters on or off using View ® Format Characters. Format characters do not print. For more information, see the *Words* manual.

This command sets how controls are displayed when the user next opens an Applixware Words window.

See also [WP_SET_CONTROLS_DISPLAY@](#)

WP_TOGGLE_FIELDS_DISPLAY@

Turns off or on the viewing of fields

Format WP_TOGGLE_FIELDS_DISPLAY@()

Method [this.toggle_fields_display@](#)

Description Displays or suppresses the viewing of field methods. Types of fields include index fields, glossary fields, date and time fields, include fields, cross-reference fields, merge fields, hypertext, conditional variables, page numbers, and so on.

WP_TOGGLE_FIELDS_DISPLAY@ is called by View ® Field Methods. Field methods will print. For more information, see the *Words* manual.

This command sets how fields are displayed when the user next opens an Applixware Words window.

See also [WP SET FIELDS DISPLAY@](#)

WP_TOGGLE_FORM_MODE@

Toggles Words into or out of Forms mode

Format WP_TOGGLE_FORM_MODE@ ()

Method [this.toggle_form_mode@](#)

Description Toggles the Words document into or out of forms mode.

See also [WP DOC IS FORM@](#)
[WP GET FORM MODE@](#)
[WP SET FORM MODE@](#)

WP_TOGGLE_FRAME_BOUNDS_DISPLAY@

Displays or suppresses the display of frame boundaries

Format WP_TOGGLE_FRAME_BOUNDS_DISPLAY@ ()

Method [this.toggle_frame_bounds_display@](#)

Description Displays or suppresses the viewing of frame boundaries.

This command sets how frame boundaries are displayed when the user next opens an Applixware Words window.

See also [WP SET FRAME BOUNDS DISPLAY@](#)

WP_TOGGLE_GRAPHICS_DISPLAY@

Displays or hides graphics

Format WP_TOGGLE_GRAPHICS_DISPLAY@ ()

Method [this.toggle_graphics_display@](#)

Description Toggles the display of graphics. Graphics always print even if their display is turned off. This macro does not affect page layout. That is, you will see blank space the same size as the graphic.

This macro is most often used on slower systems or if you have a large document with many graphics and you want to improve performance.

This command sets how graphics are displayed when the user next opens an Applixware Words window.

See also [WP SET GRAPHICS DISPLAY@](#)

WP_TOGGLE_HF_BOUNDS_DISPLAY@

Turns on or off the display of header/footer boundaries

Format WP_TOGGLE_HF_BOUNDS_DISPLAY@()

Method [this.toggle_hf_bounds_display@](#)

Description Displays or suppresses the viewing of header and footer boundaries. Header and footer boundaries will display only if you have turned them on in your document using Format ® Header and Footer, Insert ® Quick Header, or Insert ® Quick Footer.

Header and Footer boundaries do not print.

This command sets how these boundaries are displayed when the user next opens an Applixware Words window.

See also [WP SET HDRFTR BOUNDS DISPLAY@](#)

WP_TOGGLE_PAGE_BOUNDS_DISPLAY@

Displays or suppresses the display of page boundaries

Format WP_TOGGLE_PAGE_BOUNDS_DISPLAY@()

Method [this.toggle_page_bounds_display@](#)

Description Displays or suppresses the display of page boundaries.

This command sets how page boundaries are displayed when the user next opens an Applixware Words window.

See also [WP SET PAGE BOUNDS DISPLAY@](#)

WP_TOGGLE_ROW_HEADING@

Turns on and off the use of row headings

Format WP_TOGGLE_ROW_HEADING@()

Method [this.toggle_row_heading@](#)

Description Changes the state of the current row so that it becomes a table header if it is not currently one. If it is a table header, it becomes a normal row.

WP_TOGGLE_RULER_DISPLAY@

Turns off or on the viewing of the ruler

Format WP_TOGGLE_RULER_DISPLAY@()

Method [this.toggle_ruler_display@](#)

Description Displays or suppresses the viewing of the document ruler. The ruler displays includes all tab stops, style selection pull-down, table ruler toggle, column ruler toggle, and justification icons.

WP_TOGGLE_RULER_DISPLAY@ is called by View ® Ruler, or by clicking on the *ExpressLine* ruler icon. For more information, see the *Words* manual.

This command sets how the ruler is displayed when the user next opens an Applixware Words window.

See also [WP_SET_RULER_DISPLAY@](#)

WP_TO_ASCII@

Creates an ASCII version of a Words document

Format WP_TO_ASCII@(filename)

Method [this.to_ascii@\(filename\)](#)

Arguments filename The full path name of the Words document to be converted. filename must include the .aw file name extension. The path name's directory defaults to the current directory.

Description Places the ASCII file in the same directory as the Words file. Line breaks and indentations are placed exactly where the original ones were. The ASCII file is given the same name as the Words file, minus the .aw extension. For example, WP_TO_ASCII@("book.aw") creates an ASCII book file.

Note This macro will become obsolete in a future release. Use FILTER_WP_TO_ASCII@ instead.

See also [WP_COPY_TO_ASCII@](#)
[FILTER_WP_TO_ASCII@](#)

WP_TRANSPOSE_CHARS@

Switches the position of the character on either side of the cursor

Format WP_TRANSPOSE_CHARS@()

Method [this.transpose_chars@](#)

Description Inverts the position of the characters on either side of the cursor. For example, suppose you spelled *the* as *teh*. If you move the cursor so that it is between the *e* and the *h* and invoke this function, it will be correctly spelled. WP_TRANSPOSE_CHARS@ is called by Keys® Transpose.

WP_TYPE@

Inserts text at the current location

Format WP_TYPE@(string)

Method [this.type@\(string\)](#)

Arguments string The text to be inserted.

Description Inserts text into the document beginning at the current location.

WP_TYPE_EURO_SYMBOL@

Inserts a Euro symbol at the current location

Format WP_TYPE_EURO_SYMBOL@()

Method [this.type_euro_symbol@\(\)](#)

Description Inserts a euro currency symbol into the document at the current location. The euro symbol is supported only in TRUETYPE fonts, If you are typing in a bitmap font, Applixware will insert the euro symbol in the TRUETYPE font closest to the currently-used bitmap font.

WP_UNDERLINE@

Places underlines beneath selected text

Format WP_UNDERLINE@()

Method [this.underline@](#)

Description Underlines the selected text. If no text is selected, the underline attribute is applied to whatever you type after invoking the macro.

See also [WP_BOLD@](#)
[WP ITALICS@](#)

WP_UNDO@

Undoes the last edit in the current Words document

Format WP_UNDO@()

Method [this.undo@](#)

Description Repeated use of WP_UNDO@ will undo previous edits in the reverse order in which they were made. WP_UNDO@ is called by Edit Ÿ Undo.

For more information, see [Undo Description](#).

See also [WP REDO@](#)

WP_UNDO_ACTIVATE@

Turns Undo on or off

Format WP_UNDO_ACTIVATE@(activateFlag)

Method [this.undo_activate@\(activateFlag\)](#)

Arguments activateFlag A Boolean value where TRUE means that *undo* operations can occur and *undo* information will be saved.

Description Allows you to turn *undo* on or off. Normally, undo is on. However, many macros can significantly improve their performance if they turn undo off.

Applixware Words assumes that *undo* is turned on. This means that you will have to turn it off every time you load a new file.

See also [WP_UNDO@](#)

WP_UNDO_CLEAR@

Deletes the *undo* file

Format WP_UNDO_CLEAR@()

Method [this.undo_clear@](#)

Description Deletes the *undo* file.

For more information, see [Undo Description](#).

WP_UNDO_END@

Marks the end of an undoable action

Format WP_UNDO_END@(undoStr)

Method [this.undo_end@\(undoStr\)](#)

Arguments undoStr A string matching one used for [WP_UNDO_START@](#).

Description Tells Words to stop writing the transaction to the undo file. This transaction's name is identified by undoStr, which is an arbitrary name. The information recorded between the corresponding [WP_UNDO_START@](#) and this will be restored as one unit when the user invokes Edit ® Undo.

WP_UNDO_START@

Marks the beginning of an undoable action

Format WP_UNDO_START@(undoStr)

Method [this.undo_start@\(undoStr\)](#)

Arguments undoStr An arbitrary string naming the undo transaction.

Description Tells Words to begin writing change information to its undo file. undoStr names the *transaction*; that is, it names the data that will be restored. As it is typical for transactions to be nested within transactions, the name groups the data that will be restored.

For more information, see [Undo Description](#).

See also [WP_UNDO_END@](#)

WP_UNDO_STATUS@

Checks the status of the Undo/Redo system

Format infoArray = WP_UNDO_STATUS@()

Method infoArray = this.undo_status@

Description This function checks the status of the Undo/Redo system. It returns a 5-member array:

- 0: The number of Undos pending; that is, the number of times the user can press the Undo key at this point.
- 1: The text of the next Undo (as specified in original [WP_UNDO_START@](#)).
- 2: The number of Redos pending.
- 3: The text of the next Redo.
- 4: The nesting level (number of *UndoStarts* minus *UndoEnds*). It should be 0 at the start and end of a high-level operation. If it is not 0, it ignores all other information in the returned array.

For more information, see [Undo Description](#).

WP_UNESCAPE_TEXT@

Removes escape characters from text

Format outText = WP_UNESCAPE_TEXT@(inText)

Method outText = this.unescape_text@(inText)

Arguments inText The passed text string.

Description Removes escape characters from text. The macro removes escape (back slash) characters before double-quote and back slash characters, returning unescaped text.

For example, if you had the string

\"Absolute\" value \\ number

with the escape character before double-quotes and back slashes, and you wanted to remove the escape characters, you would pass the string to the macro. The macro would return the following string in outText:

"Absolute" value \ number

See also [WP_ESCAPE_TEXT@](#)

WP_UNFIELD@

Unfields all fields in a selection

Format WP_UNFIELD@()

Method [this.unfield@](#)

Description Changes the field's value into normal material. That is, this macro unfields all fields (merge, index, date, and so on) in a selection. If fields are nested within fields, only the outermost field is *unfielded*.

WP_UNFIELD_QUERY@

Determines if a selection can be *unfielded*

Format flag = WP_UNFIELD_QUERY@()

Method [flag = this.unfield_query@](#)

Description Checks if *unfielding* the current selection will have any affect. It returns TRUE if this macro will have any effect on the current selection. Otherwise, FALSE is returned.

WP_UNIQUE_MARKER_NAME@

Insures that marker name is unique

Format WP_UNIQUE_MARKER_NAME@(proposed)

Method [this.unique_marker_name@\(proposed\)](#)

Arguments proposed The proposed name of the marker.

Description Accepts a proposed name, and makes it unique so that it does not conflict with any other marker currently in the document. This is most efficient if the caller supplies a name that is likely to be unique.

WP_UP_ARROW_KEY@

Moves up one line

Format WP_UP_ARROW_KEY@()

Method [this.up_arrow_key@](#)

Description Moves the cursor up one line, retaining the current position from the left margin (if possible).

See also [WP_DOWN_ARROW_KEY@](#)

WP_USE_STYLE@

Applies a style

Format WP_USE_STYLE@(name)

Method [this.use_style@\(name\)](#)

Arguments name The name of the style.

Description Applies a style to the current selection. Inheritance rules are based on previous use. That is, it uses the last setting for inheritance set by the user when the Apply Style command was invoked.

WP_VALID_FOOTNOTE_LOCATION@

Determines if the cursor is in the main flow

Format flag = WP_VALID_FOOTNOTE_LOCATION@()

Method [flag = this.valid_footnote_location@](#)

Description Returns TRUE if a footnote reference can be inserted at the current location.

WP_VIEW_*@

Determines the scale of the displayed text

Format WP_VIEW_*@()

Method [this.view_*@](#)

Description This macro determines the scale of the displayed text. This macro calls [WP_VIEW_SCALE@](#). The following reduced and enlarged view modes are available:

WP_VIEW_40@

WP_VIEW_60@

WP_VIEW_80@

WP_VIEW_100@
WP_VIEW_120@
WP_VIEW_150@
WP_VIEW_200@
WP_VIEW_300@
WP_VIEW_400@

WP_VIEW_CHAR_ACCURATE@

Sets character accurate display state and updates profile

Format WP_VIEW_CHAR_ACCURATE@()

Method [this.view_char_accurate@](#)

Description Places Applixware Words into its WYSIWYG state. (See [WP SET VIEW CHAR ACCURATE@](#).) It also sets a profile (wpZoomCharAccurate:1) so that this state is remembered.

Use [WP VIEW LINE ACCURATE@](#) to place Applixware Words into its line-accurate state.

WP_VIEW_EXPRESSLINE@

Turns on and off the *Expressline*

Format WP_VIEW_EXPRESSLINE@()

Method [this.view_expressline@](#)

Description Toggles on or off *ExpressLine* display in Words.

WP_VIEW_LINE_ACCURATE@

Information is displayed line accurate; character spacing may be slightly inaccurate

Format WP_VIEW_LINE_ACCURATE@()

Method [this.view_line_accurate@](#)

Description Places Applixware into its line-accurate display state. (See [WP SET VIEW LINE ACCURATE@](#).) It also sets a profile (wpZoomCharAccurate:0) so that this state is remembered.

Use [WP VIEW CHAR ACCURATE@](#) to place Applixware Words into its WYSIWYG state.

WP_VIEW_SCALE@

Enlarges or reduces the display size

Format WP_VIEW_SCALE@(percentage)

Method [this.view_scale@](#)(percentage)

Arguments percentage The percentage increase or decrease of the text display size. The recommended percentages are between 25% - 1000%, depending upon your current minimum and maximum text point size.

Description Enlarges or reduces the display size of your words document. This macro offers more sizes than what is available through View ® Zoom ® nnn%.

WP_WORD_COUNT@

Count visible words in the range

Format nWords = WP_WORD_COUNT@()

Method [nWords = this.word_count@](#)

Description Counts the number of visible words in the document and returns this total. A word is a set of adjacent letters. For example, a##b is two words.
If there is no current selection, WP_WORD_COUNT@ counts the number of words in the cursor's flow whether it be the main flow, the header/footer flow, or footnote flow.

CREATE_MAIL_WP_COPY@

Writes temporary copy of current file in mail temporary directory

Format mailNamesArray = CREATE_MAIL_WP_COPY@()

Description Writes a copy of the current Words document into Applixware Mail's temporary directory.

This macro is one of the macros executed by Spreadsheets when you choose File ® Send. This temporary copy contains all of your current changes (even if

the changes have not yet been saved to disk). Ordinarily, this temporary file is deleted after the file is mailed.

The following two-element array is returned:

array[0,0] The name of the temporary file.

array[0,1] The file's original file name.

FILTER_ASCII_LINES_TO_WP@

Converts an ASCII file into a Words file

Format FILTER_ASCII_LINES_TO_WP@(infile, outfile)

Arguments infile The ASCII file to convert.
 outfile The converted Applixware Words file.

Description Converts an ASCII file into an Applixware Words file. Each line in the ASCII file is a separate paragraph in the Words file.

FILTER_ASCII_PARAS_TO_WP@

Converts an ASCII file into a Words file

Format FILTER_ASCII_PARAS_TO_WP@(infile, outfile)

Arguments infile The ASCII file to convert.
 outfile The converted Applixware Words file.

Description Converts an ASCII file into an Applixware Words file. Paragraph breaks in outfile occur when lines of text are separated by two RETURNS in infile.

FILTER_ASCL_TO_WP@

Converts an ASCII file into a Words file

Format FILTER_ASCL_TO_WP@(infile, outfile)

Arguments infile The ASCII file to convert.
 outfile The converted Applixware Words file.

Description Converts an ASCII file into an Applixware Words file, maintaining the line breaks of the original file, infile.

FILTER_ASCP_TO_WP@

Converts an ASCII file into a Words file

Format FILTER_ASCP_TO_WP@(infile, outfile)

Arguments infile The ASCII file to convert.
 outfile The converted Applixware Words file.

Description Converts an ASCII file into an Applixware Words file, constructing multi-line paragraphs in the format of the ASCII file, infile.

FILTER_HTML_TO_WP@

Converts an HTML file to an Applixware Words file

Format FILTER_HTML_TO_WP@(infile, outfile[, URL])

Arguments infile The full path name of the HTML file to be converted.
 outfile The full path name of the converted Applixware Words file, including the .aw extension.

See also [FILTER_URL_TO_GR@](#)
[FILTER_URL_TO_WP@](#)
[WP_IMPORT_HTML@](#)
[WWW_FETCH_URL@](#)
[WWW_MERGE_URL@](#)

FILTER_MSWORD_TO_WP@

Converts a MS Word into a Words file

Format FILTER_MSWORD_TO_WP@(infile, outfile)

Arguments infile The full path name of the MS Word document to be converted.
 outfile The full path name of the converted Words file, including the .aw extension.

Description Converts a copy of the MS Word file into a Words document. If infile is specified without the whole path, the path defaults to the path of an open Words window.

See also [WP_IMPORT *@](#) and [WP_EXPORT *@](#).

For more information about filters, refer to the *Words* manual.

FILTER_MSWORD70_TO_WP@

Converts a MS Word 7.0 file into a Words file

Format FILTER_MSWORD70_TO_WP@(infile, outfile)

Arguments infile The full path name of the MS Word document to be converted.
outfile The full path name of the converted Words file, including the .aw extension.

Description Converts a copy of the MS Word file into a Words document. This macro converts Word for Windows version 7.0 files, and Word for Macintosh 6.0 files.

If infile is specified without the whole path, the path defaults to the path of an open Words window.

See also [WP_IMPORT *@](#) and [WP_EXPORT *@](#).

For more information about filters, refer to the *Words* manual.

outfile The full path name of the converted Words file, including the .aw extension.

FILTER_RTF_TO_WP@

Converts from Rich Text Format to Applixware Words

Format FILTER_RTF_TO_WP@(infile, outfile)

Arguments infile The full path name of the Rich Text Format Version 1.0 document to be converted.
outfile The full path name of the converted Applixware Words file, including the .aw extension.

FILTER_URL_TO_WP@

Filters HTML files pointed to by a URL into a Words file

Format FILTER_URL_TO_WP@(URLin, wpOut[, baseURL])

Arguments URLin The URL of the graphic object.
wpOut The name of the file into which the Web HTML file will be written.
baseURL If URLin is a relative URL, this portion of the URL will be merged by [WWW_MERGE_URL@](#) with URLin to form the graphic object's URL.

Description Fetches an HTML file whose address is URLin and transforms it into an Ap-
plixware Words file.

See also [FILTER_HTML_TO_WP@](#)
[FILTER_URL_TO_GR@](#)
[WP_IMPORT_HTML@](#)
[WP_IMPORT_URL@](#)
[WWW_FETCH_URL@](#)
[WWW_MERGE_URL@](#)

FILTER_WINWORD_TO_WP@

Converts a Microsoft Word® file into a Words file

Format FILTER_WINWORD_TO_WP@(infile, outfile]

Arguments infile The full path name of the Microsoft® Word file to be converted.
outfile The full path name of the converted Words file, including the .aw
extension.

Description Converts a copy of the Microsoft Word file into a Words document. If infile is
specified without the whole path, the path defaults to the path of an open Words
window.

For more information about filters, refer to the Applixware *Words* manual.

FILTER_WINWORD20_TO_WP@

Converts a Microsoft Word® 2.0 file into a Words file

Format FILTER_WINWORD20_TO_WP@(infile, outfile]

Arguments infile The full path name of the Microsoft® Word 2.0 file to be converted.
outfile The full path name of the converted Words file, including the .aw
extension.

Description Converts a copy of the Microsoft Word 2.0 file into a Words document. If infile is
specified without the whole path, the path defaults to the path of an open Words
window.

For more information about filters, refer to the Applixware *Words* manual.

FILTER_WINWORD60_TO_WP@

Converts a Microsoft Word® 6.0 file into a Words file

Format FILTER_WINWORD60_TO_WP@(infile, outfile)

Arguments infile The full path name of the Microsoft® Word 6.0 file to be converted.
outfile The full path name of the converted Words file, including the .aw extension.

Description Converts a copy of the Microsoft Word 6.0 file into a Words document. If infile is specified without the whole path, the path defaults to the path of an open Words window.

For more information about filters, refer to the Applixware *Words* manual.

FILTER_WORDPERFECT_TO_WP@

Converts a WordPerfect® file into a Words file

Format FILTER_WORDPERFECT_TO_WP@(infile, outfile)

Arguments infile The full path name of the WordPerfect® file being converted.
outfile The full path name of the converted Applixware Words file, including the .aw extension.

Description Converts a copy of the WordPerfect file into a Words document, selecting the appropriate filter for Version 5.0, 5.1 and 6.0 WordPerfect files. If infile is specified without the whole path, the path defaults to the path of an open Words window.

See also [WP_IMPORT *@](#) and [WP_EXPORT *@](#).

For more information about filters, refer to the *Words* manual.

FILTER_WP_TO_ASC@

Converts an Applixware Words document to an ASCII file

Format flag = FILTER_WP_TO_ASC@(infile, outfile)

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.
outfile The full path name of the converted ASCII file.

Description Converts an Applixware Words document into an ASCII file. Returns flag.

NOTE: In Applixware 4.0, this macro has been replaced with [FILTER_WP_TO_ASCII_PARAS@](#).

FILTER_WP_TO_ASCII_LAYOUT@

Converts an Applixware Words document to an ASCII file, retaining format where possible

Format flag = FILTER_WP_TO_ASCII_LAYOUT@(infile, outfile[, dontLocalizeFlag])

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted ASCII file.

dontLocalizeFlag
A Boolean value which if set to TRUE indicates that no links are localized. (See [LOCALIZE_LINKS@](#).)

In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an Applixware Words document into an ASCII file. To the extent possible, the ASCII file retains the format (layout) of the Applixware Words file. Returns flag.

See also [FILTER_WP_TO_ASCII_LINES@](#)
[FILTER_WP_TO_ASCII_PARAS@](#)

FILTER_WP_TO_ASCII_LINES@

Converts an Applixware Words document to an ASCII lines file

Format flag = FILTER_WP_TO_ASCII_LINES@(infile, outfile[, dontLocalizeFlag])

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted ASCII file.

dontLocalizeFlag
A Boolean value which if set to TRUE indicates that no links are localized. (See [LOCALIZE_LINKS@](#).)

In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an Applixware Words document into an ASCII lines file. That is each paragraph in the Words file is converted into one ASCII line. Returns flag.

See also [FILTER_WP_TO_ASCII_LAYOUT@](#)
[FILTER_WP_TO_ASCII_PARAS@](#)

FILTER_WP_TO_ASCII_PARAS@

Converts an Applixware Words document to an ASCII paragraphs file

Format flag = FILTER_WP_TO_ASCII_PARAS@(infile, outfile[, dontLocalizeFlag])

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted ASCII file.

dontLocalizeFlag
A Boolean value which if set to TRUE indicates that no links are localized. (See [LOCALIZE_LINKS@](#).)

In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an Applixware Words document into an ASCII paragraphs file. That is, Words paragraph are separated from one another by blank lines. However, lines within a paragraph are broken into separate lines in the output file. Returns flag.

See also the Words Preference Filter option [Maximum Characters Per Line when Exporting to ASCII Paragraphs](#).

See also [FILTER_WP_TO_ASCII_LAYOUT@](#)
[FILTER_WP_TO_ASCII_LINES@](#)

FILTER_WP_TO_HTML@

Converts an Applixware Words document to an HTML file

Format FILTER_WP_TO_HTML@(infile, outfile)

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted HTML file.

FILTER_WP_TO_MSWORD@

Converts an Applixware Words document to a MicroSoft Word file

Format FILTER_WP_TO_MSWORD@(infile, outfile[, dontLocalizeFlag])

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted MS Word file.

dontLocalizeFlag
A Boolean value which if set to TRUE indicates that no links are localized. (See [LOCALIZE LINKS@](#).)

In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an Applixware Words document into a DOS MicroSoft Word 5.0 file.

FILTER_WP_TO_RTF@

Converts an Applixware Words document to a Rich Text Format file

Format FILTER_WP_TO_RTF@(infile, outfile[, dontLocalizeFlag])

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted Rich Text Format file.

dontLocalizeFlag
A Boolean value which if set to TRUE indicates that no links are localized. (See [LOCALIZE LINKS@](#).)

In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an Applixware Words document into a Rich Text Format file.

FILTER_WP_TO_WP311@

Converts current Words file into 3.11 format

Format FILTER_WP_TO_WP311@(infile, outfile)

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted Applixware Words 3.11 file.

Description Converts an Applixware Words file of the current release to an Applixware Words file in version 3.11 format.

This release was actually shipped as release 3.2.

See also [FILTER_WP_TO_WP400@](#)

FILTER_WP_TO_WP400@

Converts an older Words file into 4.0 format

Format FILTER_WP_TO_WP400@(infile, outfile[, dontLocalizeFlag])

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted Applixware Words 4.0 file.

dontLocalizeFlag

A Boolean value which if set to TRUE indicates that no links are localized. (See [LOCALIZE_LINKS@](#).)

In addition, if set to FALSE, the current file is written to the most current Words version before exporting begins.

Description Converts an older Applixware Words file to the current release format.

See also [FILTER_WP_TO_WP311@](#) and [FILTER_WP_TO_WP420@](#).

FILTER_WP_TO_WP420@

Converts an older Words file into 4.2 format

Format FILTER_WP_TO_WP420@(infile, outfile)

Arguments infile The full path name of the Applixware Words file being converted, including the .aw extension.

outfile The full path name of the converted Applixware Words 4.0 file.

Description Converts an older Applixware Words file to the current release format.

See also [FILTER_WP_TO_WP311@](#) and [FILTER_WP_TO_WP400@](#).

FILTER_WP400_TO_WP311@

Converts an Applixware Words 4.0 file into Words 3.11 format

Format FILTER_WP400_TO_WP311@(rev4name, rev3name)

Arguments rev4name The full path name of the Applixware Words 4 file being converted, including the .aw extension.
rev3name The full path name of the converted Applixware Words 3.11 file, including the .aw extension.

Description Converts an Applixware Words version 4.0 file into an Applixware Words version 3.11 file. Information is lost in the transformation. For example, saved markers are not supported in release 3.11.

FILTER_WP420_TO_WP311@

Converts an Applixware Words 4.2 file into Words 3.11 format

Format FILTER_WP420_TO_WP311@(rev4name, rev3name)

Arguments rev4name The full path name of the Applixware Words 4.2 file being converted, including the .aw extension.
rev3name The full path name of the converted Applixware Words 3.11 file, including the .aw extension.

Description Converts an Applixware Words version 4.2 file into an Applixware Words version 3.11 file. Information is lost in the transformation. For example, saved markers are not supported in release 3.11.

wp_.am

' Basics

format wp_location@
 bead,
 offset

format wp_range@
 format wp_location@ start,
 format wp_location@ last

' Generic bead

format WP_BEAD@

' Common elements:

 class, ' int: Class (e.g. ED_TEXT, ED_PARAGRAPH...)
 status, ' int: bits for in-table, etc.

' Text (or any symbol) bead:

 text, ' string: Raw 8-bit ASCII text
 format wp_text_attrs@ text_attrs, ' array: Effective attributes
 format wp_text_attrs_flags@ text_attrs_flags, ' array: local/inherit attrs

' Field bead:

 fieldType, ' int: Field type (e.g. ED_AUTONUM_FIELD)

' Paragraph and style bead:

 format wp_para_attrs@ para_attrs, ' array: Effective paragraph attributes
 format wp_para_attrs_flags@ para_attrs_flags, ' array: local/inherit attrs
 parentStyleName, ' string: Parent style name
 parentStyleNum, ' int: Parent style bead sequence number
 frameFlag, ' int: WP#FRAME#DEF#INHERIT, #NONE, #LOCAL
 format wp_frame@ frame, ' array: frame attributes
 tabFlag, ' int: WP#ATTRS#INHERIT, #NONE, #LOCAL
 tabs, ' array: each element is a format wp_tab@ tab
 myStyleName, ' string: Style name (for style bead only)
 styleGlossary, ' glossary prefix for styles
 nextStyleName, ' string: for style beads.

' Marker bead:

,
markerName, ' string: Marker name
deletable, ' int: Can marker be deleted?
saveToFile, ' int: Should marker be saved in .aw file?
,

' Table cell bead:

,
width, ' int: cell width, in mils (1000 mils = 1 inch)
leftCellMargin, ' int: margin from left of cell
rightCellMargin, ' int: margin from right of cell
topCellMargin, ' int: margin from top of cell
bottomCellMargin, ' int: margin from bottom of cell
verticalAlignment, ' int: WP#VERT#ALIGN#TOP, CENTER, ...
locked, ' int: TRUE -> cell contents locked (form mode)
cellId, ' str: unique cell ID (form mode)
dblClickMacro, ' str: macro to call on dbl click (form mode)
entryMacro, ' str: macro to call when cell entered (form mode)
exitMacro, ' str: macro to call when cell exited (form mode)
maxChars, ' int: Max # chars (enforced by C
code)
caseness, ' int: Force this case upon contents
underline, ' int: TRUE -> underline entire cell
width
promptText, ' str: Display this text on status line
when in cell
nextIfFull, ' int: TRUE -> type in next cell when
this fills
nextCellId, ' str: Where Select Next Cell goes
(form mode)
lastCellInRow, ' int: TRUE -> this is last cell in row
,

' Border attributes & flags (para, style and cell_end beads)

,
format wp_border_attrs@ border, ' array: Effective attrs
format wp_border_attrs_flags@ border_flags, ' array: local/inherit attrs
,

' Table row bead:

,
heading, ' int: TRUE -> this is a table heading row
alignment, ' int: WP#ROW#ALIGN#LEFT, #CENTER, #RIGHT
indent, ' int: indent from left margin, in mils (1000 mils = 1 inch)
topRowMargin, ' int: in mils
bottomRowMargin, ' int: in mils
heightType, ' int: WP#ROW#HEIGHT#AUTO, #EXACT, #MIN, #MAX
height, ' int: row height, in mils
,

' Section bead:

```

sectType,          ' int: WP#SECTION#CONTINUOUS, ...
nCols,            ' int: # of columns in section
gutterWidth,     ' int: space between snaking columns
pageWidth,       ' int: width of page, in mils (1000 mils = 1 inch)
pageHeight,     ' int: height of page, in mils
pageOrientation, ' int: WP_PAGE_PORTRAIT or _LANDSCAPE
paperSize,      ' int: code for letter, A4, etc.
printSourceTray, ' int: specific tray; overrides
                 paperSize
printDestTray,  ' int: code for specific tray
duplex,         ' int: WP#PRINT#SIMPLEX,
                 #DUPLEX#LONG, #SHORT
leftMargin,     ' int: left page margin, in mils (1000 mils = 1 inch)
rightMargin,   ' int: right page margin, in mils
topMargin,     ' int: top page margin, in mils
bottomMargin,  ' int: bottom page margin, in mils
bindingMargin, ' int: addl margin on "binding" side (2-sided)
oddHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
evenHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
firstHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
lastHeaderType, ' int: WP_HDRFTR_NORMAL, etc.
oddFooterType, ' int: WP_HDRFTR_NORMAL, etc.
evenFooterType, ' int: WP_HDRFTR_NORMAL, etc.
firstFooterType, ' int: WP_HDRFTR_NORMAL, etc.
lastFooterType, ' int: WP_HDRFTR_NORMAL, etc.
oddHeaderName,  ' string: name of hdrftr/glossary
evenHeaderName, ' string: name of hdrftr/glossary
firstHeaderName, ' string: name of hdrftr/glossary
lastHeaderName, ' string: name of hdrftr/glossary
oddFooterName,  ' string: name of hdrftr/glossary
evenFooterName, ' string: name of hdrftr/glossary
firstFooterName, ' string: name of hdrftr/glossary
lastFooterName, ' string: name of hdrftr/glossary
sectNumStyle,   ' int: section numbering style (arabic, ...)
controlSectNum, ' int: section # control (same, restart, ...)
restartSectNumVal, ' int: number to restart section numbering at
separator,      ' string: section/page number separator
pageNumStyle,   ' int: page numbering style (arabic, ...)
restartPageNum, ' int: restart page numbering if TRUE
restartPageNumVal, ' int: number to restart page numbering at

```

```
' Series definition bead
```

```
format wp_series@ series,
```

```
' Bitmap beads:
```

```

        bitmapName,                ' string: Name (for bitmaps only)
,
' Document Start bead:
,
        levelIndent,              ' int: Extra left/first indent per para level
        language,                 ' int: Extra left/first indent per para level
        hyphMethod,               ' int: WP_HYPH_NONE,WP_HYPH_DICT,WP_HYPH_CALC
        printIndex,               ' int: TRUE -> print index with doc
        printTOC,                 ' int: TRUE -> print TOC with doc
        allowWidowOrphan,        ' int: TRUE -> disable widow/orphan control
        facingPages,              ' int: TRUE -> hdr/fters are even and odd
        headerMargin,             ' int: top of page to top of header, in mils (1000 mils = 1 inch)
        footerMargin,             ' int: bottom of page to bottom of footer, in mils
,
' Clipboard beads:
,
        clipType,                 ' int: Clipboard number (e.g. CLIP_PRIMARY)
,
' Header/Footer bead
,
        hdrftrName,              ' string: header/footer name
,
' tag (character style) bead
,
'''    tagName,                  ' string: tag name
,
        theEnd
,
.....
' Text attributes
,
format wp_text_attrs@
        bold,                     ' int: TRUE or FALSE
        italic,                   ' int: TRUE or FALSE
        strikethru,               ' int: TRUE or FALSE
        hidden,                   ' int: TRUE or FALSE
        capitals,                 ' int: WP_ATTRS_CAPS_XXXX
        underline,                ' int: WP_ATTRS_UNDER_XXXX
        position,                 ' int: WP_ATTRS_POS_XXXX
        hyphenate,                ' int: TRUE or FALSE
        color,                    ' string: color name
        face,                     ' string: Font face name
        size,                     ' int: in points
        tag,                      ' string: tag name
,
.....
' Text attribute flags
,

```

```

format wp_text_attrs_flags@
    bold,                ' int: TRUE->local, FALSE->inherited
    italic,              ' int: TRUE->local, FALSE->inherited
    strikethru,         ' int: TRUE->local, FALSE->inherited
    hidden,             ' int: TRUE->local, FALSE->inherited
    capitals,          ' int: TRUE->local, FALSE->inherited
    underline,         ' int: TRUE->local, FALSE->inherited
    position,          ' int: TRUE->local, FALSE->inherited
    hyphenate,
    color,              ' int: TRUE->local, FALSE->inherited
    face,               ' int: TRUE->local, FALSE->inherited
    size,               ' int: TRUE->local, FALSE->inherited
    tag                 ' int: TRUE->local, FALSE->inherited

```

.....

' Text attributes with flags
,

```

format wp_flagged_text_attrs@
    format wp_text_attrs_flags@ setFlags,
        'set if attribute value is used
    format wp_text_attrs_flags@ inheritFlags,
        'set if attribute value is inherited
    format wp_text_attrs@ attrs 'attribute values where setFlags is set

```

.....

' Paragraph attributes
,

```

format wp_para_attrs@
    pageBreak,          ' int: TRUE (pagebreak before para) or FALSE
    keepWith,          ' int: TRUE (keep w/next para) or FALSE
    block,              ' int: TRUE (keep para on single page) or FALSE
    justify,            ' int: horiz alignment (WP_ATTRS_HAL_XXXX)
    inTOC,              ' int: TRUE (put para in Tbl of Cont) or FALSE
    indentToLevel,     ' int: TRUE (indent para to level) or FALSE
    noSpell,            ' int: TRUE (don't spellcheck para) or FALSE
    firstIndent,       ' int: left indent of 1st line in picas
    leftIndent,        ' int: left indent of other lines in picas
    rightIndent,       ' int: right indent from right edge in picas
    preParaSpacing,    ' mils before para (1000 mils = 1 inch)
    postParaSpacing,   ' mils after para
    lineSpacing,       ' mils after each line
    level,              ' int: para level number
    hyphZone,          ' int: percent whitespace
    hyphMinFrag        ' int: min chars before and/or after hyphen

```

.....

' Paragraph attribute flags
,

```

format wp_para_attrs_flags@
    pageBreak,          ' int: TRUE->local, FALSE->inherited
    keepWith,           ' int: TRUE->local, FALSE->inherited
    block,              ' int: TRUE->local, FALSE->inherited
    justify,            ' int: TRUE->local, FALSE->inherited
    inTOC,              ' int: TRUE->local, FALSE->inherited
    indentToLevel,     ' int: TRUE->local, FALSE->inherited
    noSpell,           ' int: TRUE->local, FALSE->inherited
    firstIndent,       ' int: TRUE->local, FALSE->inherited
    leftIndent,        ' int: TRUE->local, FALSE->inherited
    rightIndent,       ' int: TRUE->local, FALSE->inherited
    preParaSpacing,    ' int: TRUE->local, FALSE->inherited
    postParaSpacing,   ' int: TRUE->local, FALSE->inherited
    lineSpacing,       ' int: TRUE->local, FALSE->inherited
    level,              ' int: TRUE->local, FALSE->inherited
    hyphZone,          ' int: TRUE->local, FALSE->inherited
    hyphMinFrag        ' int: TRUE->local, FALSE->inherited

```

' Paragraph attributes with flags

```

format wp_flagged_para_attrs@
    format wp_para_attrs_flags@ setFlags,
    'set if attribute value is used or inherited
    format wp_para_attrs_flags@ inheritFlags,
    'set if attribute value is inherited
    format wp_para_attrs@ attrs 'attribute values where set and not inherited

```

' Frame

```

format wp_frame@
    type,
    xposType,
    xposRelative,
    xpos,
    yposType,
    yposRelative,
    ypos,
    widthType,
    width,
    heightType,
    height,
    leftMargin,
    rightMargin,
    topMargin,
    bottomMargin

```

' Series definition bead

```

format wp_series@
    name,                ' name of series
    style,               ' Code number for 1,2,3... or a,b,c.. etc.
    multiLevel,         ' if true, show multiple levels
    showFrom,           ' array, but 1st element not used.
    seperator,          ' array, but 1st element not used.
    leaderText,         ' Global leader
    trailerText,        ' Global trailer
    leaders,             ' array of per-level leaders
    trailers             ' array of per-level trailers

```

```
' Tab
,
```

```

format wp_tab@
    type,                ' int: WP_TAB_XXXX
    deleted,             ' int: TRUE -> tab deleted, not added
    xpos,                ' int: indent of tab in picas
    leaderStr,           ' string: leadering for tab (may incl spaces)
    decimalChar          ' string: decimal tab control char (one char)

```

```

.....
' Style, as returned by WP_GET_STYLE@(name)
' !!1 should include effective text & para atts.
,
```

```

format wp_style@
    name,                ' Latest name for style
    orig_name,           ' Original name of style
    textFlag,            ' int: TRUE->locals, FALSE->all inherits
    format wp_text_attrs@ text_attrs, ' array: Effective attrs
    format wp_text_attrs_flags@ text_attrs_flags, ' array: local/inherit attrs
    paraFlag,           ' int: TRUE->locals, FALSE->all inherits
    format wp_para_attrs@ para_attrs,      ' array: Effective paragraph attributes
    format wp_para_attrs_flags@ para_attrs_flags, ' array: local/inherit attrs
    frameFlag,          ' int: TRUE->local, FALSE->inherit
    format wp_frame@ frame, ' array: frame attributes if local
    tabFlag,            ' int: TRUE->local, FALSE->inherit
    format wp_tab@ tabs, ' array: add and delete list for tabs
    format wp_tab@ inheritTabs, ' array: inherited tabs (obs?)
    borderFlag,         ' int: TRUE->local, FALSE->inherit
    format wp_border_attrs@ border, ' array: Effective border attributes
    format wp_border_attrs_flags@ border_flags, ' arr: local/inherit atts (obs?)
    parent,             ' name of parent style
    glossary,           ' prefix glossary
    nextStyle,          ' next style name
    changed             ' TRUE -> values changed

```

```

.....
' View state, as returned by WP_GET_VIEW_STATE@

```

```

format wp_view_state@
    pilcros,           ' view pilcros
    ruler,            ' view ruler
    breaks,          ' view page/column/section/line breaks
    fieldMethods,    ' view field methods (rather than field codes)
    allBounds,       ' view all boundaries
    cellBounds,      ' view cell boundaries
    frameBounds,     ' view frame boundaries
    hdrFtrBounds,    ' view hdr/ftr boundaries
    pageBounds,      ' view page boundaries
    zoomFactor,      ' percentage zoom (100 -> 100%, 200 -> 2X)
    wysiwyg,         ' 0-line positioning, 1-char positioning
    graphics,        ' if false, use placeholders for graphics
    htmlTags         ' view unknown HTML tags

```

```

.....
' Header/Footer info

```

```

format wp_hdrftr@
    oddHeaderType,
    evenHeaderType,
    firstHeaderType,
    lastHeaderType,
    oddFooterType,
    evenFooterType,
    firstFooterType,
    lastFooterType,
    oddHeaderName,
    evenHeaderName,
    firstHeaderName,
    lastHeaderName,
    oddFooterName,
    evenFooterName,
    firstFooterName,
    lastFooterName

```

```

.....
' Page Setup

```

```

format wp_page_setup@
    pageOrientation, ' portrait, landscape
    pageWidth,       ' in mils (1000 mils = 1 inch)
    pageHeight,      ' in mils
    leftMargin,      ' in mils
    rightMargin,     ' in mils
    topMargin,       ' in mils
    bottomMargin,    ' in mils

```

bindingMargin,	' in mils
paperSize,	' int: code for letter, A4, etc.
printSourceTray,	' int: specific tray; overrides paperSize
printDestTray,	' int: code for specific tray
duplex	' int: WP#PRINT#SIMPLEX, #DUPLEX#LONG, #DUPLEX#SHORT

.....

' Border attributes

format wp_line_attrs@

thickness,	' line thickness in mils, 0 if no line
doubleLine,	' if true, two lines of same thickness
color	' color name (1000 mils = 1 inch)

format wp_shading_attrs@

shadingStyle,	
fillName,	' user defined fill pattern (Aster*x image object name)
fgColor,	' foreground color name
bgColor	' background color name

format wp_border_attrs@

format wp_line_attrs@ left,	' left border
format wp_line_attrs@ right,	' right border
format wp_line_attrs@ top,	' top border
format wp_line_attrs@ bottom,	' bottom border
format wp_line_attrs@ horizontal,	' horz separator (cells, between paras)
format wp_line_attrs@ vertical,	' vertical separator (cells)
format wp_shading_attrs@ shading,	' background shading
horizontalMargin,	' in mils (paras) (1000 mils = 1 inch)
verticalMargin,	' in mils (paras)
dropShadow	' drop shadow style

format wp_border_attrs_flags@

left,	' int: TRUE->local, FALSE->inherited
right,	' int: TRUE->local, FALSE->inherited
top,	' int: TRUE->local, FALSE->inherited
bottom,	' int: TRUE->local, FALSE->inherited
horizontal,	' int: TRUE->local, FALSE->inherited
vertical,	' int: TRUE->local, FALSE->inherited
shading,	' int: TRUE->local, FALSE->inherited
horizontalMargin,	' int: TRUE->local, FALSE->inherited
verticalMargin,	' int: TRUE->local, FALSE->inherited
dropShadow	' int: TRUE->local, FALSE->inherited

.....

.....

' Row Attributes (same as row_start bead)

,

format wp_row_attrs@

alignment,	' int: WP#ROW#ALIGN#LEFT, #CENTER, #RIGHT
indent,	' int: indent from left margin, in mils
topRowMargin,	' int: in mils (1000 mils = 1 inch)
bottomRowMargin,	' int: in mils
heightType,	' int: WP#ROW#HEIGHT#AUTO, #EXACT, #MIN, #MAX
height,	' int: row height, in mils
heading	' int: TRUE -> this is a table heading row

.....

' Cell Attributes (same as cell_end bead)

' Also used for Entry Fields

,

format wp_cell_attrs@

width,	' int: cell width, in mils (min width of Entry fld) (1000 mils = 1 inch)
leftCellMargin,	' int: margin from left of cell (ignore for E fld)
rightCellMargin,	' int: margin from right of cell (ignore for E fld)
topCellMargin,	' int: margin from top of cell (ignore for E fld)
bottomCellMargin,	' int: margin from bottom of cell (ignore for E fld)
verticalAlignment,	' int: WP#VERT#ALIGN#TOP, CENTER, ... (not for E f)
locked,	' int: TRUE -> cell contents locked (in form mode)
cellId,	' str: unique cell ID (form mode)
dblClickMacro,	' str: macro to call on dbl click (form mode)
entryMacro,	' str: macro to call when cell entered (form mode)
exitMacro,	' str: macro to call when cell exited (form mode)
maxChars,	' int: Max # chars (enforced by C code)
caseness,	' int: Force this case upon contents
underline,	' int: TRUE -> underline entire cell width
promptText,	' str: Display this text on status line when in cell
nextIfFull,	' int: TRUE -> type in next cell when this fills

nextCellId, ' str: Where Select Next Cell goes
(form mode)
lastCellInRow ' int: TRUE-> this is last cell in row
(cell only)

.....

' Equation editor: matrix info
,

format eqn_matrix_info@
rows, 'int: number of rows
cols, 'int: number of columns
rowAlign, 'int: row align type
colAlign, 'int: column align type
equalRowHts, 'int: TRUE if row heights are set to equal
equalColWds 'int: TRUE if column widths are set to equal

.....

' WP_PRINT_DEVICE_SPECIFIC@ parameter block
,

format wp_print_info@
printername, ' printer name or "" if to file
color, ' print in color?
copies, ' number of copies
banner, ' print banner page too?
allPages, ' print all pages?
range,
merge, ' merge print?
file, ' merge file
class, ' printer class (eg PostScript)
type, ' printer type (eg HP4SI6_1.PPD)
page1, ' start page num
pageN, ' last page num
draft, ' OBSOLETE
contins, ' OBSOLETE
changed, ' OBSOLETE
pageExt1, ' OBSOLETE
pageExtN, ' OBSOLETE
background ' print in background?

.....

' Status
,

format wp_status@
pageNum, /* page number at cursor */
pageCount, /* total number of pages */
colCount, /* num columns selected (0 for non-table selection) */
verticalPosition, /* position on page (from top) of cursor */
pageHeight, /* height of page in mils (1000 mils = 1 inch) */

```

inFootNote,      /* TRUE iff cursor is in a footnote */
inHdrFtr,        /* TRUE iff cursor is in a header or footer */
inFrame,         /* TRUE iff cursor is in a frame */
inTable,         /* TRUE iff cursor is in a table */
inHeading,       /* TRUE if cursor is in a table heading */
inField,         /* TRUE iff cursor is in a field */
fieldMethod,     /* level 1 field method string if cursor in field */
overtimeMode,   /* TRUE iff typing mode is overtime. */
macroEditing,    /* TRUE iff this is the macro editor */
equationEditing, /* TRUE iff this is the equation editor */
helpMode,        /* TRUE iff we are in help mode */
formMode,        /* TRUE iff we are in form mode */
paraNum,         /* Paragraph number (line number in macro editor) */
paraCount,       /* num paras in current flow (line count in ME) */
readOnly,        /* TRUE iff WP in read-only mode */
viewOnly,        /* TRUE iff WP in view-only mode */
zoomFactor,      /* zoom percentage */
modified,        /* TRUE if document modified */
htmlEditing,     /* HTML_MODE_NONE, _EDIT,
                 _VIEW */
windowless       /* TRUE if opened in background (by
                 ELF script) */

```

.....

```

' CodePg
' To be used to determine the codepage to use for file conversion

```

```

format codepage@
    force_use,
    other_choices

```

.....

```

/* Generic defines for ELF */

```

```

define AX#FORBIDDEN#SYMBOL "$"
define AX#RESERVED#SYMBOL "@"

```

```

/* Common characters */

```

```

define BLANK$ " "
define SQUOTE$ "'"
define SBACKQUOTE$ "`"
define DQUOTE$ "\""
define BANG$ "!"

```

```

/*** Document variable names ***/

```

```

define WP#EDIT#HISTORY#VAR      "EditHistory$"
define WP#LANGUAGE#VAR          "Language@"
define WP#USER#NAME#VAR         "UserName$"
define WP#FILE#CREATOR#VAR      "Creator$"
define WP#FILE#CREATED#VAR      "Created$"
define WP#CREATION#DATE#VAR     "CreationDate$"
define WP#CREATION#USER#VAR     "CreationUser$"
define WP#FILE#UPDATED#VAR      "Updated$"
define WP#UPDATE#DATE#VAR       "UpdateDate$"
define WP#UPDATE#USER#VAR       "UpdateUser$"
define WP#READ#ONLY#VAR        "ReadOnly@"
define WP#TERSE#FILE#VAR        "TerseFile@"
define WP#BUILD#STAMP#VAR       "BuildStamp$"
define WP#DOC#NAME#VAR          "DocName$"
define WP#PATH#NAME#VAR         "PathName$"
define WP#PATH#NAME#REAL#VAR    "PathNameReal$"
define WP#BACKGROUND#SAVE#VAR   "BackgroundSave$"
define ME#SAVE#COMPILED#VAR     "SaveCompiled@"
define WP#LEVEL#INDENT#VAR      "LevelIndent@"
define WP#WIDOW#ORPHAN#VAR     "AllowWidowOrphan@"
define WP#FACING#PAGES#VAR      "FacingPages@"
define WP#HEADER#MARGIN#VAR     "HeaderMargin@"
define WP#FOOTER#MARGIN#VAR     "FooterMargin@"
define WP#IGNORE#LAST#PARA#VAR  "IgnoreLastPara@"
define WP#IGNORE#LAST#SECT#VAR  "IgnoreLastSect@"
define WP#CHANGE#BAR#POS#VAR    "ChangeBarPos@"
define WP#HOOK#MACRO#NAME#VAR   "HookMacroName@"
define WP#HOOKS#VAR             "HookArray$"
define WP#DOC#WIN#ATTR#TYPE#VAR "DocWindowAttrType@"
define WP#DOC#WIN#SIZE#VAR      "DocWindowSize$"
define WP#DOC#IS#FORM#VAR       "DocsIsForm@"
define WP#MACRO#SRC#PREFIX#VAR  "ElfSrc$"
define WP#MACRO#OBJ#PREFIX#VAR  "ElfObj$"
define WP#EMULATE#21#VAR        "Emulate21@"
define WP#DOC#ZOOM#VAR          "DocZoom@"

```

/** Session variable names **/

```

define WP#USER#DICT#VAR         "DefaultDict@"
define WP#DICTS#VAR             "CustomDicts@"
define WP#SKIP#DICT#VAR         "SkipDict@"
define WP#DOC#DICT#VAR          "DocDict@"
define WP#NEW#DICTS#VAR         "NewDicts@"
define WP#PAGE#COUNT#VAR      "PageCount$"
define WP#VIEW#ONLY#VAR         "ViewOnly$"
define ME#CURRENT#COMPILE#VAR   "CurrentCompile$"
define ME#COMPILED#OBJECT#VAR   "CompiledObject$"

```

```
define ME#SAVE#ASCII#VAR "SaveASCII@"
define WP#IS#REV2#DOC#VAR "IsRev2Doc@"
```

```
/** Document-specific Window Creation Options **/
```

```
define WP#DOC#WIN#ATTR#DEFAULT 0
define WP#DOC#WIN#ATTR#LAST 1
define WP#DOC#WIN#ATTR#FIXED 2
```

```
/** Style names **/
```

```
/*
```

Note there are a pair of defines for each style:

- The default name of the style (e.g. 'Cell')
- The name of an associated document variable (e.g. 'CellStyle@')

When the code needs to find a default style, it looks up the name in the appropriate document var. These will be preloaded with the default names.

```
*/
```

```
define WP#CELL#STYLE "Style name","Cell"
define WP#CELL#STYLE#VAR "CellStyle@"
define WP#HDR#FTR#STYLE "Style name","HdrFtr"
define WP#HDR#FTR#STYLE#VAR "HdrFtrStyle@"
define WP#FOOTNOTE#STYLE "Style name","Footnote"
define WP#FOOTNOTE#STYLE#VAR "FootnoteStyle@"
define WP#FIELD#ERROR#STYLE "Style name","FieldError"
define WP#FIELD#ERROR#STYLE#VAR "FieldErrorStyle@"
define WP#FRAME#STYLE "Style name","Frame"
define WP#FRAME#STYLE_VAR "FrameStyle@"
define WP#TOC#STYLE "Style name","TOC"
define WP#TOC#STYLE#VAR "TOCStyle@"
define WP#TOC#STYLE#PREFIX "Style name prefix","TOC-
```

```
/** Tags **/
```

```
define WP#CELL#MACRO#TAG "$$PrevSelectedCell$$"
define WP#MAIN#FLOW#CURSOR#TAG "$$MainFlowCursor$$"
```

```
/** Glossaries **/
```

```
define WP#MERGE#GLOSSARY "Merge$"
define WP#CELL#RADIO#ON#GLOSSARY "CellRadioOn@"
define WP#CELL#RADIO#OFF#GLOSSARY "CellRadioOff@"
define WP#CELL#TOGGLE#ON#GLOSSARY "CellToggleOn@"
define WP#CELL#TOGGLE#OFF#GLOSSARY "CellToggleOff@"
define WP#FOOTNOTE#BAR#GLOSSARY "FootnoteBar@"
define WP#FOOTNOTE#CONT#BAR#GLOSSARY "FootnoteContBar@"
define WP#ENDNOTE#BAR#GLOSSARY "EndnoteBar@"
```

```

define WP#TABLE#CONT#GLOSSARY "TableContinuation@"

/*** Constants ***/

define FORCE#ON          1
define OPTIONAL#ON      2
define WP#TEMPLATE#FILE "empty.aw" /* Dflt WP style */
define WP#ASCII#TEMPLATE#FILE "ascii.aw" /* Dflt WP ASCII style*/
define ME#TEMPLATE#FILE  "empty.am" /* Dflt ME style */
define ME#ASCII#TEMPLATE#FILE "ascii.am" /* Dflt ME ASCII style*/
define HTML#TEMPLATE#FILE XLT("html.aw")/* Dflt HTML style */
define WP#MARKER#PADDED#CELL NXLT("PaddedCell")
                                /* Extra Copy cell */
define WP#MARKER#ENTRY#START NXLT("$$EntryStart$$")
                                /* Before E fld */
define WP#MARKER#ENTRY#END   NXLT("$$EntryEnd$$")
                                /* After E fld */

/*** WP attributes ***/

/* 'Generic' attribute flags */

define WP#ATTRS#NONE      -2
define WP#ATTRS#LOCALS   -1
define WP#ATTRS#INHERIT   0
define WP#ATTRS#UNKNOWN  -999

/* General values */

define WP#NONE#STRING "None"
define WP#OFF#STRING  "Off"
define WP#ON#STRING   "On"
define WP#LOCAL#STRING "Local"
define WP#NO#STRING   "No"
define WP#YES#STRING  "Yes"

/*** Character attributes ***/

/* Font names */

define WP#FONT#TIMES_ "Times"
define WP#FONT#TIMES#STRING "Times Roman"
define WP#FONT#BOOKMAN_ "Bookman"
define WP#FONT#BOOKMAN#STRING "ITC Bookman"
define WP#FONT#SCHOOL_ "New Century Schoolbook"
define WP#FONT#SCHOOL#STRING "New Century Schoolbook"
define WP#FONT#CHANCERY_ "Zapf Chancery"
define WP#FONT#CHANCERY#STRING "ITC Zapf Chancery"

```

```

define WP#FONT#PALATINO_ "Palatino"
define WP#FONT#PALATINO#STRING "Palatino"
define WP#FONT#COURIER_ "Courier"
define WP#FONT#COURIER#STRING "Courier"
define WP#FONT#HELVETICA_ "Helvetica"
define WP#FONT#HELVETICA#STRING "Helvetica"
define WP#FONT#HELVNARROW_ "Helvetica Narrow"
define WP#FONT#HELVNARROW#STRING "Helvetica Narrow"
define WP#FONT#AVANT_ "Avant Garde"
define WP#FONT#AVANT#STRING "ITC Avant Garde Gothic"
define WP#FONT#SYMBOL_ "Symbol"
define WP#FONT#SYMBOL#STRING "Symbol"
define WP#FONT#DINGBATS_ "Zapf Dingbats"
define WP#FONT#DINGBATS#STRING "Zapf Dingbats"
define WP#FONT#MINCHOU_ "Minchou"
define WP#FONT#MINCHOU#STRING "Minchou"
define WP#FONT#GOTHIC "Gothic"
define WP#FONT#GOTHIC#STRING "Gothic"

/* "Raw text face array" contents:
 * Note we don't use zero, as it is convenient in the C source to treat
 * this array as a string */

define WP#FONT#TYPE#LATIN      1
define WP#FONT#TYPE#SYMBOL    2
define WP#FONT#TYPE#DINGBATS  3

/* Pointsize */

define WP#SIZE#6#STRING "6 point"
define WP#SIZE#8#STRING "8 point"
define WP#SIZE#10#STRING "10 point"
define WP#SIZE#12#STRING "12 point"
define WP#SIZE#14#STRING "14 point"
define WP#SIZE#18#STRING "18 point"
define WP#SIZE#24#STRING "24 point"
define WP#SIZE#36#STRING "36 point"
define WP#SIZE#OTHER#STRING "Other"

/* Position */

define WP#POS#NORMAL#POINTS_ 0
define WP#POS#SUB#POINTS_ -3
define WP#POS#SUPER#POINTS_ 3
define WP#POS#NORMAL#STRING "Normal"
define WP#POS#SUB#STRING "Subscript"
define WP#POS#SUPER#STRING "Superscript"

```

```

/* Underline */

define WP#UNDER#NONE_0
define WP#UNDER#SINGLE_1
define WP#UNDER#SINGLE#STRING "Single"
define WP#UNDER#DOUBLE_2 ' Double underlining
define WP#UNDER#DOUBLE#STRING "Double"
define WP#UNDER#WORDS_3 ' By words
define WP#UNDER#WORDS#STRING "Single Words"
define WP#UNDER#DOUBLE#WORDS_4
define WP#UNDER#DOUBLE#WORDS#STRING "Double Words"
define WP#UNDER#NOTABS 5
define WP#UNDER#NOTABS#STRING "Single Not Tabs"
define WP#UNDER#DOUBLE#NOTABS 6
define WP#UNDER#DOUBLE#NOTABS#STRING "Double Not Tabs"

/* Caps */

define WP#CAPS#NORMAL_0
define WP#CAPS#UPPER_1
define WP#CAPS#LOWER_2

/* Hyphenation Methods */

6.                0 */
define WP#HYPHENATE#ALGORITHM 0
define WP#HYPHENATE#DICTIONARY 1      ' Really means Dict w/Rule-based backup
define WP#HYPHENATE#NONE_2

7.                0 */
/* Note we preserve the order of the previous defines (see above)
   so that old scripts will still run properly */
define WP#HYPH#CALC#ONLY 0      /* Rule-based only */
define WP#HYPH#DICT#CALC 1      /* Dictionary with rule-based backup */
define WP#HYPH#NONE 2          /* Don't hyphenate */
define WP#HYPH#DICT#ONLY 3      /* Dictionary only */
define WP#HYPH#PROFILE 4        /* Use profile setting */

/* Proximity hyphenation defines - as copied from aspell.h for use in
   direct calls to generic (non-WP) hyphenation functions (e.g. in the
   TEXT_HYPHENATE@ macro) */
define HYPH_NONE 0          /* Don't hyphenate */
define HYPH_DICT 1         /* Dictionary only */
define HYPH_CALC 2         /* Rule-based only */
define HYPH_DICT_CALC 3    /* Dictionary with rule-based backup */

/** Paragraph attributes */

```

```

/* Justification */

define WP#JUSTIFY#LEFT_0    ' Left aligned
define WP#JUSTIFY#LEFT#STRING "Left"
define WP#JUSTIFY#RIGHT_1  ' Right aligned
define WP#JUSTIFY#RIGHT#STRING "Right"
define WP#JUSTIFY#CENTER_2    ' Center aligned
define WP#JUSTIFY#CENTER#STRING "Center"
define WP#JUSTIFY#FULL_3    ' Padded to both sides
define WP#JUSTIFY#FULL#STRING "Justified"

/* Tab types */

define WP#MAX#TABS          50
define WP#TAB#LEFT_0
define WP#TAB#RIGHT_1
define WP#TAB#CENTER_2
define WP#TAB#DECIMAL_3
define WP#TAB#INDENT_4

/* Pagebreak */

define WP#PBREAK#BEFORE#STRING "Before"

/* Keep with next */

define WP#KEEPWITH#NEXT#STRING "Next"

/* Indent to level */

define WP#INDENTTOLEV#NO#STRING "No affect"
define WP#INDENTTOLEV#YES#STRING "Has affect"

/** Frames **/

/* Frame identity */

define WP#FRAME#DEF#INHERIT    0
define WP#FRAME#DEF#NONE      1
define WP#FRAME#DEF#LOCAL      2

/* Frame types */

define WP#FRAME#FLOW#AROUND    0
define WP#FRAME#JUMP#OVER      1
define WP#FRAME#BACKGROUND     2

/* Frame horizontal position */

```

```

define WP#FRAME#XREL#PAGE      0
define WP#FRAME#XREL#MARGIN    1
define WP#FRAME#XREL#COLUMN    2

define WP#FRAME#XPOS#LEFT      0
define WP#FRAME#XPOS#CENTER    1
define WP#FRAME#XPOS#RIGHT     2
define WP#FRAME#XPOS#OUTSIDE   3
define WP#FRAME#XPOS#INSIDE    4

/* Frame vertical position */

define WP#FRAME#YREL#PAGE      0
define WP#FRAME#YREL#MARGIN    1
define WP#FRAME#YREL#PARA      2

define WP#FRAME#YPOS#TOP       0
define WP#FRAME#YPOS#CENTER    1
define WP#FRAME#YPOS#BOTTOM    2

/* Frame width */

define WP#FRAME#WIDTH#AUTO     0
define WP#FRAME#WIDTH#EXACT    1

/* Frame height */

define WP#FRAME#HEIGHT#AUTO    0
define WP#FRAME#HEIGHT#EXACT   1
define WP#FRAME#HEIGHT#MIN     2
define WP#FRAME#HEIGHT#MAX     3

/*** Section ***/

/* Section types */

define WP#SECTION#CONTINUOUS   0
define WP#SECTION#NEXT#COLUMN  1
define WP#SECTION#NEXT#PAGE    2
define WP#SECTION#ODD#PAGE     3
define WP#SECTION#EVEN#PAGE    4

/* Header/Footer types */

define WP#ODD#HDRFTR#INHERIT   0
define WP#ODD#HDRFTR#NORMAL    1
define WP#ODD#HDRFTR#NONE      2

```

```

define WP#ODD#HDRFTR#USE#GLOSSARY 3

define WP#EVEN#HDRFTR#USE#ODD0
define WP#EVEN#HDRFTR#INHERIT 1
define WP#EVEN#HDRFTR#NORMAL 2
define WP#EVEN#HDRFTR#NONE 3
define WP#EVEN#HDRFTR#USE#GLOSSARY 4

define WP#FIRST#HDRFTR#NOT#SPECIAL 0
define WP#FIRST#HDRFTR#INHERIT 1
define WP#FIRST#HDRFTR#NORMAL 2
define WP#FIRST#HDRFTR#NONE 3
define WP#FIRST#HDRFTR#USE#GLOSSARY 4

define WP#LAST#HDRFTR#NOT#SPECIAL 0
define WP#LAST#HDRFTR#INHERIT 1
define WP#LAST#HDRFTR#NORMAL 2
define WP#LAST#HDRFTR#NONE 3
define WP#LAST#HDRFTR#USE#GLOSSARY 4

/* Page orientations */

define WP#PAGE#PORTRAIT_ 0
define WP#PAGE#LANDSCAPE_ 1

/* Printing options */

define WP#PRINT#SIMPLEX 0
define WP#PRINT#DUPLEX#LONG 1
define WP#PRINT#DUPLEX#SHORT 2

/** Borders & Shading */

/* Shading */

define WP#SHADING#FILL#A 0
define WP#SHADING#FILL#M 1
define WP#SHADING#FILL#B 2
define WP#SHADING#FILL#C 3
define WP#SHADING#FILL#D 4
define WP#SHADING#FILL#N 5
define WP#SHADING#FILL#E 6
define WP#SHADING#FILL#F 7
define WP#SHADING#FILL#G 8
define WP#SHADING#FILL#H 9
define WP#SHADING#FILL#I 10
define WP#SHADING#FILL#J 11
define WP#SHADING#FILL#K 12

```

```

define WP#SHADING#FILL#L    13
define WP#SHADING#FILL#O    14
define WP#SHADING#FILL#P    15
define WP#SHADING#FILL#Q    16
define WP#SHADING#FILL#R    17
define WP#SHADING#NONE      18
define WP#SHADING#LINEAR#GRADIENT  19
define WP#SHADING#RADIAL#GRADIENT  20
define WP#SHADING#RECT#GRADIENT    21
define WP#SHADING#USER#DEFINED     22

/* Shadow */

define WP#SHADOW#NONE          0
define WP#SHADOW#BOTTOM#RIGHT  1
define WP#SHADOW#TOP#LEFT      2
define WP#SHADOW#BOTTOM#LEFT   3
define WP#SHADOW#TOP#RIGHT     4

/* Border type based on cursor/selection */

define WP#BORDER#PARAGRAPH      0
define WP#BORDER#SINGLE#ROW#SINGLE#COL  1
define WP#BORDER#SINGLE#ROW#MULTI#COL  2
define WP#BORDER#MULTI#ROW#SINGLE#COL  3
define WP#BORDER#MULTI#ROW#MULTI#COL  4

/* Range types based on section */

define WP#RANGE#CURSOR          0
define WP#RANGE#SINGLE#SECTION  1
define WP#RANGE#MULTI#SECTION  2

/** Numbering **/

/* Generic numbering styles - for series, page and section numbering*/

define WP#NUMBERING#NONE -1
define WP#NUMBERING#ARABIC      0    /* 1, 2, 3... */
define WP#NUMBERING#LETTER#UPPER  1    /* A, B, C... */
define WP#NUMBERING#LETTER#LOWER  2    /* a, b, c... */
define WP#NUMBERING#ROMAN#UPPER   3    /* I, II, III... */
define WP#NUMBERING#ROMAN#LOWER  4    /* i, ii, iii... */

/* Page numbering styles */
/* !!0 To be phased out in favor of the above */

define WP#PAGE#NUM#ARABIC        0

```

```

define WP#PAGE#NUM#ALPHABETIC#UPPER    1
define WP#PAGE#NUM#ALPHABETIC#LOWER    2
define WP#PAGE#NUM#ROMAN#UPPER         3
define WP#PAGE#NUM#ROMAN#LOWER         4

```

```
/* Section numbering controls */
```

```

define WP#SECT#NUM#SAME      0
define WP#SECT#NUM#INCREM    1
define WP#SECT#NUM#RESTART   2

```

```
/* Special for standard and user-defined numbered series formats */
```

```

define WP#NUM#SERIES#1      0
define WP#NUM#SERIES#CAP#A  1
define WP#NUM#SERIES#SMALL#A 2
define WP#NUM#SERIES#CAP#I  3
define WP#NUM#SERIES#SMALL#I 4
define WP#NUM#SERIES#PREFIX  "UserSeries"
define WP#NUM#SERIES#BASE 1000

```

```
/** Change bars */
```

```

define WP#CB#NOCHANGE      0
define WP#CB#ADDED         1
define WP#CB#DELETED       2
define WP#CB#CHANGED       3

```

```

define WP#CB#POS#LEFT      0
define WP#CB#POS#RIGHT     1
define WP#CB#POS#OUTER     2

```

```

define WP#CB#NONE          0
define WP#CB#STANDARD      1
define WP#CB#NO#ATTRS     2

```

```
/** Fields */
```

```
/* Field defines */
```

```

define WP#DATE#FIELD 1
define WP#DATE#FIELD#STRING "date"
define WP#DATE#FIELD#NAME "Date"
define WP#TIME#FIELD 2
define WP#TIME#FIELD#STRING "time"
define WP#TIME#FIELD#NAME "Time"
define WP#NUMBER#FIELD 3
define WP#NUMBER#FIELD#STRING "series"

```

```

define WP#NUMBER#FIELD#NAME "Numbered Series"
define WP#FOOTNOTE#REF#FIELD 4
define WP#FOOTNOTE#FIELD#STRING "footnote"
define WP#FOOTNOTE#FIELD#NAME "Footnote"
define WP#FOOTNOTE#BODY#FIELD 5
define WP#FOOTNOTE#BODY#FIELD#STRING "footnote_body"
define WP#FOOTNOTE#BODY#FIELD#NAME "Footnote Body"
define WP#LINK#FILE#FIELD 6
define WP#LINK#FIELD#STRING "link"
define WP#LINK#FIELD#NAME "Link"
define WP#LOCAL#INSET#FIELD 7
define WP#LOCAL#INSET#FIELD#STRING "local_inset"
define WP#LOCAL#INSET#FIELD#NAME "Local Inset"
define WP#INCLUDE#FILE#FIELD 8
define WP#INCLUDE#FIELD#STRING "include"
define WP#INCLUDE#FIELD#NAME "Include"
define WP#GLOSSARY#FIELD 9
define WP#GLOSSARY#FIELD#STRING "glossary"
define WP#GLOSSARY#FIELD#NAME "Glossary"
define WP#XREF#FIELD 10
define WP#XREF#FIELD#STRING "xref"
define WP#XREF#FIELD#NAME "Cross-Reference"
define WP#XREF#SOURCE#FIELD 11
define WP#XREF#SOURCE#FIELD#STRING "xref_source"
define WP#XREF#SOURCE#FIELD#NAME "Cross-Reference Source"
define WP#INDEX#FIELD 12
define WP#INDEX#FIELD#STRING "index"
define WP#INDEX#FIELD#NAME "Index Entry"
define WP#MAKE#INDEX#FIELD 13
define WP#MAKE#INDEX#FIELD#STRING "make_index"
define WP#MAKE#INDEX#FIELD#NAME "Index"
define WP#MAKE#TOC#FIELD 14
define WP#MAKE#TOC#FIELD#STRING "make_TOC"
define WP#MAKE#TOC#FIELD#NAME "Table of Contents"
define WP#MERGE#FIELD 15
define WP#MERGE#FIELD#STRING "merge"
define WP#MERGE#FIELD#NAME "Merge"
define WP#MACRO#FIELD 16
define WP#MACRO#FIELD#STRING "macro"
define WP#MACRO#FIELD#NAME "Macro"
define WP#HYPER#TARGET#FIELD 17
define WP#HYPER#TARGET#FIELD#STRING "hyper_target"
define WP#HYPER#TARGET#FIELD#NAME "Hypertarget"
define WP#SOFT#HYPHEN#FIELD 18
define WP#SOFT#HYPHEN#FIELD#STRING "-"
define WP#SOFT#HYPHEN#FIELD#NAME "Soft Hyphen"
define WP#PAGE#NUMBER#FIELD 19
define WP#PAGE#NUMBER#FIELD#STRING "page_number"

```

```

define WP#PAGE#NUMBER#FIELD#NAME "Page Number"
define WP#PAGE#COUNT#FIELD 20
define WP#PAGE#COUNT#FIELD#STRING "page_count"
define WP#PAGE#COUNT#FIELD#NAME "Page Count"
define WP#IF#VAR#FIELD 21
define WP#IF#VAR#FIELD#STRING "if_var"
define WP#IF#VAR#FIELD#NAME "Conditional Variable"
define WP#IF#MACRO#FIELD 22
define WP#IF#MACRO#FIELD#STRING "if_macro"
define WP#IF#MACRO#FIELD#NAME "Conditional Macro"
define WP#ELF#STR#FIELD 23
define WP#ELF#STR#FIELD#STRING "macro_string"
define WP#ELF#STR#FIELD#NAME "ELF Macro String"
define WP#ELF#BEAD#FIELD 24
define WP#ELF#BEAD#FIELD#STRING NXLT("macro_beads")
define WP#ELF#BEAD#FIELD#NAME XLT("ELF Macro Beads")
define WP#VAR#STR#FIELD 25
define WP#VAR#STR#FIELD#STRING NXLT("var_string")
define WP#VAR#STR#FIELD#NAME XLT("Document Variable
                                String")

define WP#ENTRY#FIELD 26
define WP#ENTRY#FIELD#STRING NXLT("entry")
define WP#ENTRY#FIELD#NAME XLT("Entry")
define WP#HTML#TAG#FIELD 27
define WP#HTML#TAG#FIELD#STRING NXLT("html_tag")
define WP#HTML#TAG#FIELD#NAME NXLT("HTML Tag")
define WP#PRINTER#CODE#FIELD 28
define WP#PRINTER#CODE#FIELD#STRING NXLT("printer_code")
define WP#PRINTER#CODE#FIELD#NAME NXLT("Printer Code")

/* Field parameters */
/* Note:
    General pattern is WP_FIELD_PARM_<fieldtype>_<identifier>

    However, if a parameter is shared between multiple field types,
    use the pattern: WP_FIELD_PARM_<identifier>
*/
/* !!0 To do:
    - Make Index and TOC share _TITLE parm
*/

define WP#FIELD#PARM#TEXT           NXLT("-text")
define WP#FIELD#PARM#DTEXT          NXLT("-dtext")
define WP#FIELD#PARM#STEXT          NXLT("-stext")
define WP#FIELD#PARM#TEXT#STYLE     NXLT("-textStyle")
define WP#FIELD#PARM#OUT#OF#DATE    NXLT
    ("-saveOutOfDate")
define WP#FIELD#PARM#READ#EVAL      NXLT("-readEval")

```

```

define WP#FIELD#PARM#PRINT#EVAL      NXLT("-printEval")
define WP#FIELD#PARM#SAVE#EVAL       NXLT("-saveEval")
define WP#FIELD#PARM#NO#EVAL         NXLT("-noEval")
define WP#FIELD#PARM#EMPTY#VALUE     NXLT("-emptyValue")
define WP#FIELD#PARM#GLOSSARY#VALUE  NXLT("-glossary")
define WP#FIELD#PARM#2CLICK          NXLT("-doubleClick")
define WP#FIELD#PARM#2CLICK#MACRO    NXLT
    ("-dblClickMacro")
define WP#FIELD#PARM#ENTER#MACRO     NXLT("-enterMacro")
define WP#FIELD#PARM#EXIT#MACRO      NXLT("-exitMacro")
define WP#FIELD#PARM#TYPING#MACRO    NXLT("-typingMacro")
define WP#FIELD#PARM#MACRO#TEXTBUTTON NXLT
    ("-textButton")
define WP#FIELD#PARM#MACRO#TB#WIDTH  NXLT
    ("-buttonWidth")
define WP#FIELD#PARM#MACRO#ICONBUTTON NXLT("-iconButton")
define WP#FIELD#PARM#MACRO#CMD        NXLT("-pass")
define WP#FIELD#PARM#INSET#DOC#TYPE   NXLT("-docType")
define WP#FIELD#PARM#INSET#APP#TYPE   NXLT("-appType")
define WP#FIELD#PARM#INSET#TYPE       NXLT("-type")
/* OLD (use -appType) */
define WP#FIELD#PARM#INSET#CVT        NXLT("-cvt")
define WP#FIELD#PARM#INSET#CVTCMD    NXLT("-cvtcmd")
define WP#FIELD#PARM#INSET#APP        NXLT("-app")
define WP#FIELD#PARM#INSET#VIEW       NXLT("-view")
define WP#FIELD#PARM#INSET#CLIPMODE   NXLT("-clipMode")
define WP#FIELD#PARM#INSET#XSCALE     NXLT("-xscale")
define WP#FIELD#PARM#INSET#YSCALE     NXLT("-yscale")
define WP#FIELD#PARM#INSET#XOFFSET    NXLT("-xoffset")
define WP#FIELD#PARM#INSET#YOFFSET    NXLT("-yoffset")
define WP#FIELD#PARM#INSET#SRCWIDTH   NXLT("-srcWidth")
define WP#FIELD#PARM#INSET#SRCHEIGHT  NXLT("-srcHeight")
define WP#FIELD#PARM#INSET#PROPORTION NXLT
    ("-proportional")
define WP#FIELD#PARM#INSET#LASTSECT   NXLT("-lastSection")
define WP#FIELD#PARM#INSET#SHRINK#TBL NXLT
    ("-shrinkToFit")
define WP#FIELD#PARM#INDEX#SPAN       NXLT("-span")
define WP#FIELD#PARM#INDEX#SEEALSO    NXLT("-see")
define WP#FIELD#PARM#INDEX#LETTERHEAD  NXLT
    ("-letterHeader")
define WP#FIELD#PARM#INDEX#BLANKHEAD  NXLT
    ("-blankHeader")
define WP#FIELD#PARM#INDEX#NOHEAD     NXLT("-noHeader")
define WP#FIELD#PARM#INDEX#TITLE      NXLT("-title")
define WP#FIELD#PARM#NEWSTYLES        NXLT("-newStyles")
define WP#FIELD#PARM#TOC#STYLES       NXLT("-styles")
define WP#FIELD#PARM#TOC#TITLE        NXLT("-contents")

```

```

define WP#FIELD#PARM#TOC#MAXLEVEL NXLT("-maxLevel")
define WP#FIELD#PARM#TOC#NOTAB NXLT("-noTab")
define WP#FIELD#PARM#TOC#PLAINTAB NXLT("-plainTab")
define WP#FIELD#PARM#TOC#DOTTAB NXLT("-dotTab")
define WP#FIELD#PARM#TOC#NONUMS NXLT("-noPageNums")
define WP#FIELD#PARM#TOC#STRIP#TABS NXLT("-stripTabs")
define WP#FIELD#PARM#SERIES#NOINCREM NXLT
(-noIncrement")
define WP#FIELD#PARM#SERIES#NODISPLAY NXLT
(-noDisplay")
define WP#FIELD#PARM#SERIES#SET NXLT("-set")
define WP#FIELD#PARM#SERIES#LEVEL NXLT("-level")
define WP#FIELD#PARM#FOOTNOTE#SERIES NXLT("-series")
define WP#FIELD#PARM#FOOTNOTE#ENDDOC NXLT
(-endDoc")
define WP#FIELD#PARM#XREFSRC#LIVE NXLT("-live")
define WP#FIELD#PARM#XREF#NEXT NXLT("-next")
define WP#FIELD#PARM#XREF#PREV NXLT("-prev")
define WP#FIELD#PARM#XREF#MAINFLOW NXLT("-mainFlow")
define WP#FIELD#PARM#XREF#PAGE#TOP NXLT("-pageTop")
define WP#FIELD#PARM#XREF#PAGE#BOTTOM NXLT
(-pageBottom")
define WP#FIELD#PARM#MERGE#OFFSET NXLT("-rowOffset")
define WP#FIELD#PARM#MERGE#ADVANCE NXLT("-rowAdvance")
define WP#FIELD#PARM#MERGE#LAST#PARA NXLT("-usePara")
define WP#FIELD#PARM#NO#ICON NXLT("-noIcon")
define WP#FIELD#PARM#ICON NXLT("-icon")
define WP#FIELD#PARM#FORMAT NXLT("-format")
define WP#FIELD#PARM#COMMENT NXLT("-comment")
define WP#FIELD#PARM#HTML NXLT("-html")
define WP#FIELD#PARM#HTML#TAG NXLT("-htmlTag")
define WP#FIELD#PARM#HTML#PARM NXLT("-htmlParm")
define WP#FIELD#PARM#HTML#ANCHOR NXLT("-htmlAnchor")
define WP#FIELD#PARM#HTML#IMG NXLT("-htmlImg")
define WP#FIELD#PARM#HTML#ALIGN NXLT("-htmlAlign")
define WP#FIELD#PARM#HTML#ALT NXLT("-htmlAlt")
define WP#FIELD#PARM#HTML#ISMAP NXLT("-htmlIsmap")
define WP#FIELD#PARM#ENTRY#CHARS NXLT("-maxChars")
define WP#FIELD#PARM#ENTRY#START#VAL NXLT("-startValue")
define WP#FIELD#PARM#ENTRY#END#VAL NXLT("-endValue")
define WP#FIELD#PARM#ENTRY#NEXT#ID NXLT("-nextId")
define WP#FIELD#PARM#ENTRY#LOCKED NXLT("-locked")
define WP#FIELD#PARM#ENTRY#NEXT#FULL NXLT("-nextIfFull")
define WP#FIELD#PARM#ENTRY#CASE NXLT("-case")
define WP#FIELD#PARM#ENTRY#UNDERLINE NXLT("-underline")
define WP#FIELD#PARM#ENTRY#PROMPT NXLT("-promptText")
define WP#FIELD#PARM#ENTRY#WIDTH NXLT("-minWidth")
define WP#FIELD#PARM#DOMAIN NXLT("-domain")

```

```

define WP#FIELD#PARM#PRINT#ASCII    NXLT("-ascii")
define WP#FIELD#PARM#PRINT#PCL      NXLT("-pcl")
define WP#FIELD#PARM#PRINT#PS       NXLT("-postscript")
define WP#FIELD#PARM#PRINT#GDI      NXLT("-gdi")

```

/** Objects **/

/* App types */

```

define WP#APP#TYPE#NONE              0
define WP#APP#TYPE#WORDS             1
define WP#APP#TYPE#GRAPHICS          2
define WP#APP#TYPE#SPREADSHEET      3
define WP#APP#TYPE#MACRO             4
define WP#APP#TYPE#AUDIO             5
define WP#APP#TYPE#EQUATION          6
define WP#APP#TYPE#QUERY            7

```

/* Graphics clip modes */

```

define WP#PICT#SCALE#TO#FIT          0
define WP#PICT#CLIP#TO#FIT           1
define WP#PICT#SCALE#IF#WOULD#CLIP  2

```

/** Color **/

```

define WP#MAX#COLOR#VALUE 255

```

/** Tables **/

/* Row alignment */

```

define WP#ROW#ALIGN#LEFT             0
define WP#ROW#ALIGN#CENTER           1
define WP#ROW#ALIGN#RIGHT            2

```

/* Row height */

```

define WP#ROW#HEIGHT#AUTO            0
define WP#ROW#HEIGHT#EXACT           1
define WP#ROW#HEIGHT#MIN             2
define WP#ROW#HEIGHT#MAX             3

```

/* Vertical alignment within cell */

```

define WP#VERT#ALIGN#TOP              0
define WP#VERT#ALIGN#CENTER           1
define WP#VERT#ALIGN#BOTTOM          2

```

```

define WP#VERT#ALIGN#JUSTIFIED 3

/*** Footnotes ***/

/* Special series for footnotes */

define WP#FOOTNOTE#SERIES#PAGE      "Footnote$" /* old define */
define WP#FOOTNOTE#SERIES#DOC "Endnote$" /* old define */
define WP#FOOTNOTE#SERIES "Footnote$"
define WP#ENDNOTE#SERIES "Endnote$"

/* Footnote types */

define WP#FOOTNOTE#BOTTOM#PAGE      0
define WP#FOOTNOTE#END#DOC          1

/*** Hypertext ***/

/* Special for Help Hypertext */

define WP#HELP#PREFIX#VAR "HelpPrefix@"
define WP#HELP#GLOSSARY#VAR "HelpGlossary@"
define WP#HELP#SECTION#VAR "HelpSection@"
define WP#HELP#STYLE#VAR "HelpStyle@"

/* Search directions */

define WP#SEARCH#FORWARDS      0
define WP#SEARCH#BACKWARDS    1
define WP#SEARCH#BOTH          2

/*** Spelling ***/

/* Useful constants */

define WP#MAX#NUM#DICTS      8

/* Maximum-sized word Proximity (speller, thesaurus) wants to deal with: */

define MAXWORD 32

/*** HTML stuff ***/
define HTML#MODE#NONE      0
define HTML#MODE#EDIT     1
define HTML#MODE#VIEW     2

/*** Debugging tools ***/

```

```
/* Flags determining what tests are carried out (on a continual basis).
   In C, bit flags. In ELF, array indices. */
```

```
define WP#DEBUG#CHECK#MEMORY 0
define WP#DEBUG#DOC#VALID 1
```

```
/** WP bead classes **/
```

```
/*
```

```
    Document bead types.
```

```
    Note that all paired types must have the end after the start and
    have a value >= ED_FIRST_PAIRED.
```

```
*/
```

```
define WP_NO_CLASS                0
define WP_TEXT_CLASS              1
define WP_PARAGRAPH_CLASS         2
define WP_STYLE_CLASS             3
define WP_BITMAP_CLASS            4
define WP_OBJECT_CLASS            5
define WP_COLUMN_BREAK_CLASS     6
define WP_PAGE_BREAK_CLASS       7
define WP_MARKER_CLASS           8
define WP_ERROR_CLASS            9
define WP_CELL_END_CLASS         10
define WP_ROW_START_CLASS        11
define WP_SECTION_CLASS          12
define WP_SERIES_CLASS           13
define WP_FIELD_SPLIT_CLASS      14
define WP_EQUATION_CLASS         15
define WP_VARIABLE_CLASS         16
define WP_COLOR_CLASS            17
define WP_GRAPHICS_CLASS         18
define WP_TAG_CLASS              19

define WP_FIELD_START_CLASS      20
define WP_FIELD_END_CLASS        21
define WP_STYLE_START_CLASS      22
define WP_STYLE_END_CLASS        23
define WP_GROUP_START_CLASS      24
define WP_GROUP_END_CLASS        25
define WP_FOOTNOTE_START_CLASS   26
define WP_FOOTNOTE_END_CLASS     27
define WP_HDRFTR_START_CLASS     28
define WP_HDRFTR_END_CLASS       29
define WP_GLOSSARY_START_CLASS   30
define WP_GLOSSARY_END_CLASS     31
define WP_FLOW_START_CLASS       32
```

```

define WP_FLOW_END_CLASS          33
define WP_VARIABLES_START_CLASS   34
define WP_VARIABLES_END_CLASS     35
define WP_DOCUMENT_START_CLASS    36
define WP_DOCUMENT_END_CLASS      37
define WP_MAX_CLASS_NUMBER       WP_DOCUMENT_END_CLASS

```

```

/** Equation Editor */

```

```

/* Equation editor template classes */

```

```

define WP#EQN_EMPTY              0      ' empty template
define WP#EQN_EMPTY_DELETABLE    0      ' deletable slot
define WP#EQN_EMPTY_NONDELETABLE 1      ' non deletable
                                          slot

define WP#EQN_SYMBOL              1      ' symbol template
define WP#EQN_SYMBOL_CHAR         0      ' simple character from
                                          variable font

define WP#EQN_SYMBOL_EQUAL        1      ' equals character, =
define WP#EQN_SYMBOL_RELATIONAL   2      ' relational
                                          operators, >=, <=, ...

define WP#EQN_SYMBOL_EMBELLISH    3      ' embellishments like ',",`, ...
define WP#EQN_SYMBOL_SPACE        4      ' space adjustments btw templates
define WP#EQN_SYMBOL_OPERATOR     5      ' operators like +,-,*,x,...
define WP#EQN_SYMBOL_DIGIT        6      ' char is digit
define WP#EQN_SYMBOL_ARROW        7      ' different types of
                                          arrows

define WP#EQN_SYMBOL_LOGICAL      8      ' logical symbols
define WP#EQN_SYMBOL_SET          9      ' set theory symbols
define WP#EQN_SYMBOL_LOGREEK     10     ' lower case chars from
                                          greek charset

define WP#EQN_SYMBOL_UPGREEK     11     ' upper case chars from
                                          greek charset

define WP#EQN_SYMBOL_MISC         12     ' misc symbols
define WP#EQN_SYMBOL_FUNCTION     13     ' function names
define WP#EQN_SYMBOL_MAXSUBS      14     ' max sub types for
                                          symbol template

define WP#EQN_FENCE              2      ' fence template
define WP#EQN_FENCE_PAREN         0      ' ( .. )
define WP#EQN_FENCE_PAREN_LEFT   1      ' ( ..
define WP#EQN_FENCE_PAREN_RIGHT  2      ' .. )
define WP#EQN_FENCE_BRACE         3      ' { .. }
define WP#EQN_FENCE_BRACE_LEFT   4      ' { ..
define WP#EQN_FENCE_BRACE_RIGHT  5      ' .. }
define WP#EQN_FENCE_BRACKET       6      ' [ .. ]
define WP#EQN_FENCE_BRACKET_LEFT 7      ' [ ..
define WP#EQN_FENCE_BRACKET_RIGHT 8      ' .. ]
define WP#EQN_FENCE_VERT          9      ' | .. |

```

```

define WP#EQN_FENCE_VERT_LEFT      10  '| ..
define WP#EQN_FENCE_VERT_RIGHT     11  '| .. |
define WP#EQN_FENCE_DVERT          12  '| || .. ||
define WP#EQN_FENCE_DVERT_LEFT     13  '| || ..
define WP#EQN_FENCE_DVERT_RIGHT    14  '| .. ||
define WP#EQN_FENCE_OVERBAR        15  '| single overbar
define WP#EQN_FENCE_DBLOVERBAR     16  '| double overbar
define WP#EQN_FENCE_UNDERBAR       17  '| singleunderline
define WP#EQN_FENCE_DBLUNDERBAR    18  '| doubleunder
line
define WP#EQN_FENCE_FLOOR          19  '/* |_ ... _| */
define WP#EQN_FENCE_FLOOR_LEFT     20  '/* |_ ... */
define WP#EQN_FENCE_FLOOR_RIGHT    21  '/* ... _| */
define WP#EQN_FENCE_CEIL           22  '/* |~ ... ~| */
define WP#EQN_FENCE_CEIL_LEFT      23  '/* |~ ... */
define WP#EQN_FENCE_CEIL_RIGHT     24  '/* ... ~| */
define WP#EQN_FENCE_BRACKET_2LEFT  25  '/* [...[ */
define WP#EQN_FENCE_BRACKET_2RIGHT 26  '/* ]...] */
define WP#EQN_FENCE_BRACKET_OPP    27  '/* ]...[ */
define WP#EQN_FENCE_PAREN_BRACKET  28  '/* (...) */
define WP#EQN_FENCE_BRACKET_PAREN  29  '/* [...] */
define WP#EQN_FRACTION              4   ' fractions or radicals
define WP#EQN_FRACTION_OVER         0   ' up OVER down
define WP#EQN_FRACTION_DIVIDE       1   ' sub up OVER sub
down
define WP#EQN_FRACTION_SLASH        2   ' up SLASH down
define WP#EQN_FRACTION_BY           3   ' sub up SLASH sub
down
define WP#EQN_FRACTION_VERTSLASH    4   ' 80 degrees bar
up 80SLASH down
define WP#EQN_SQRT                  5
define WP#EQN_SQRT_SQRT             0   ' square root of
define WP#EQN_SQRT_NROOT            1   ' nth root of
define WP#EQN_SQRT_CURVYSQRT        2   ' curvy square root
symbol
define WP#EQN_SQRT_CURVYNROOT        3   ' curvy nth root
define WP#EQN_SQRT_SQRTOVER         4   ' body over sqrt
define WP#EQN_SQRT_MAXSUBS          5   ' max sub types for
radicals
define WP#EQN_SUBSUP                6   ' subscript and
superscript template
define WP#EQN_SUBSUP_SUB            0   ' subscript only
define WP#EQN_SUBSUP_SUP            1   ' superscript only
define WP#EQN_SUBSUP_SUBSUP         2   ' superscript and
subscript
define WP#EQN_SUBSUP_SUP_LEFT       3   ' superscript on left
side
define WP#EQN_SUBSUP_SUB_LEFT       4   ' subscript on left side

```

```

define WP#EQN_SUBSUP_SUBSUP_LEFT 5 ' sub and
superscripts on left side
define WP#EQN_SUBSUP_SUP_OVER6 ' superscript on top
define WP#EQN_SUBSUP_SUB_OVER7 ' superscript on top
define WP#EQN_SUBSUP_SUBSUP_OVER 8 ' superscript on
top
define WP#EQN_SUBSUP_MAXSUBS 5 ' max sub types for
subsup
define WP#EQN_SUM 7 ' summation template
define WP#EQN_SUM_NONE 0 ' summation with no limits
define WP#EQN_SUM_UP 1 ' up limit
define WP#EQN_SUM_DOWN 2 ' down limit
define WP#EQN_SUM_UPDOWN 3 ' up and down limits
define WP#EQN_SUM_SUP 4 ' down limit as sub
define WP#EQN_SUM_SUB 5 ' up limit as sup
define WP#EQN_SUM_SUBSUP 6 ' up and down limits as sup
and sub
define WP#EQN_INTEGRAL_NONE 7 ' no limits
define WP#EQN_INTEGRAL_UP 8 ' up limit
define WP#EQN_INTEGRAL_DOWN 9 ' down limit
define WP#EQN_INTEGRAL_UPDOWN 10 ' up and down limits
define WP#EQN_INTEGRAL_SUP 11 ' up limit as sup
define WP#EQN_INTEGRAL_SUB 12 ' down limit as sub
define WP#EQN_INTEGRAL_SUBSUP 13 ' up and down limits as
sup and sub
define WP#EQN_2INTEGRAL_NONE 14 ' no limits
define WP#EQN_2INTEGRAL_UP 15 ' up limit
define WP#EQN_2INTEGRAL_DOWN 16 ' down limit
define WP#EQN_2INTEGRAL_UPDOWN 17 ' up and down
limits
define WP#EQN_2INTEGRAL_SUP 18 ' up limit as sup
define WP#EQN_2INTEGRAL_SUB 19 ' down limit as
sub
define WP#EQN_2INTEGRAL_SUBSUP 20 ' up and down
limits as sup and sub
define WP#EQN_3INTEGRAL_NONE 21 ' no limits
define WP#EQN_3INTEGRAL_UP 22 ' up limit
define WP#EQN_3INTEGRAL_DOWN 23 ' down limit
define WP#EQN_3INTEGRAL_UPDOWN 24 ' up and down
limits
define WP#EQN_3INTEGRAL_SUP 25 ' up limit as sup
define WP#EQN_3INTEGRAL_SUB 26 ' down limit as
sub
define WP#EQN_3INTEGRAL_SUBSUP 27 ' up and down
limits as sup and sub
define WP#EQN_OINTEGRAL_NONE 28 ' no limits
define WP#EQN_OINTEGRAL_UP 29 ' up limit
define WP#EQN_OINTEGRAL_DOWN 30 ' down limit

```

```

define WP#EQN_OINTEGRAL_UPDOWN 31 ' up and down
                                limits
define WP#EQN_OINTEGRAL_SUP 32 ' up limit as sup
define WP#EQN_OINTEGRAL_SUB 33 ' down limit as
                                sub
define WP#EQN_OINTEGRAL_SUBSUP 34 ' up and down
                                limits as sup and sub
define WP#EQN_PRODUCT_NONE 35 ' no limits
define WP#EQN_PRODUCT_UP 36 ' up limit
define WP#EQN_PRODUCT_DOWN 37 ' down limit
define WP#EQN_PRODUCT_UPDOWN 38 ' up and down limits
define WP#EQN_PRODUCT_SUP 39 ' up limit as sup
define WP#EQN_PRODUCT_SUB 40 ' down limit as sub
define WP#EQN_PRODUCT_SUBSUP 41 ' up and down limits as
                                sup and sub
define WP#EQN_UNION_NONE 42 ' union no limits
define WP#EQN_UNION_UP 43 ' up limit
define WP#EQN_UNION_DOWN 44 ' down limit
define WP#EQN_UNION_UPDOWN 45 ' up and down limits
define WP#EQN_UNION_SUP 46 ' up limit as sup
define WP#EQN_UNION_SUB 47 ' down limit as sub
define WP#EQN_UNION_SUBSUP 48 ' up and down limits as
                                sup and sub
define WP#EQN_INTERS_NONE 49 ' intersection with no
                                limits
define WP#EQN_INTERS_UP 50 ' up limit
define WP#EQN_INTERS_DOWN 51 ' down limit
define WP#EQN_INTERS_UPDOWN 52 ' up and down limits
define WP#EQN_INTERS_SUP 53 ' up limit as sup
define WP#EQN_INTERS_SUB 54 ' down limit as sub
define WP#EQN_INTERS_SUBSUP 55 ' down limit as sub
define WP#EQN_SCRIPT_NONE 56 ' General function with no limits
define WP#EQN_SCRIPT_UP 57 ' up limit
define WP#EQN_SCRIPT_DOWN 58 ' down limit
define WP#EQN_SCRIPT_UPDOWN 59 ' up and down limits
define WP#EQN_SCRIPT_SUP 60 ' up limit as sup
define WP#EQN_SCRIPT_SUB 61 ' down limit as sub
define WP#EQN_SCRIPT_SUBSUP 62 ' down limit as sub
define WP#EQN_ARROW 8 ' labeled arrow
                                template
define WP#EQN_ARROW_LEFT_NONE 0 ' left arrow with
                                no limits
define WP#EQN_ARROW_LEFT_UP 1 ' left arrow with
                                up limit
define WP#EQN_ARROW_LEFT_DOWN 2 ' left arrow with
                                down limit
define WP#EQN_ARROW_RIGHT_NONE 3 ' right arrow with
                                no limits

```

```

define WP#EQN_ARROW_RIGHT_UP      4      ' right arrow with
                                         up limit
define WP#EQN_ARROW_RIGHT_DOWN    5      ' right arrow with
                                         down limit
define WP#EQN_ARROW_LTRT_NONE     6      ' left right arrow
                                         with no limits
define WP#EQN_ARROW_LTRT_UP       7      ' left right arrow
                                         with up limit
define WP#EQN_ARROW_LTRT_DOWN     8      ' left right arrow
                                         with down limit
define WP#EQN_MATRIX              9      ' matrix template
define WP#EQN_MATRIX_1ROW_2COL    0      ' matrix of size 1
                                         row 2 cols
define WP#EQN_MATRIX_2ROW_1COL    1      ' matrix of size 2
                                         rows 1 col
define WP#EQN_MATRIX_2ROW_2COL    2      ' matrix of size 2
                                         rows 2 cols
define WP#EQN_MATRIX_1ROW_3COL    3      ' matrix of size 1
                                         row 3 cols
define WP#EQN_MATRIX_3ROW_1COL    4      ' matrix of size 3
                                         rows 1 col
define WP#EQN_MATRIX_3ROW_3COL    5      ' matrix of size 3
                                         rows 3 cols
define WP#EQN_MATRIX_1ROW_NCOL    6      ' matrix of size 1
                                         row n cols
define WP#EQN_MATRIX_MROW_1COL    7      ' matrix of size m
                                         rows 1 col
define WP#EQN_MATRIX_MROW_NCOL    8      ' matrix of size m
                                         rows n cols
define WP#EQN_ATTR                10     ' template attribute
define WP#EQN_ATTR_XNUDGE         0
define WP#EQN_ATTR_YNUDGE         1
define WP#EQN_NEXT                 11     ' next equation in pile

```