

**PGS.MD4**

Copyright © 1995 Soft-Logik Publishing Corporation

---

**COLLABORATORS**

	<i>TITLE :</i> PGS.MD4		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		February 12, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>PGS.MD4</b>	<b>1</b>
1.1	insert	1
1.2	insertbreak	1
1.3	insertchar	2
1.4	insertcontrol	2
1.5	insertdash	3
1.6	insertdate	3
1.7	insertindexmark	5
1.8	insertmark	5
1.9	insertname	6
1.10	insertnumber	6
1.11	insertpage	8
1.12	insertrevision	8
1.13	insertsoft	9
1.14	insertspace	10
1.15	inserttext	10
1.16	inserttime	11
1.17	liveupdate	13
1.18	loadfontprefs	13
1.19	loadprintprefs	14
1.20	loadsettings	14
1.21	loadsnapshot	14
1.22	lock	15
1.23	lockinterface	15
1.24	logrevision	15
1.25	macropalette	16
1.26	maintoolbox	17
1.27	makepagespread	17
1.28	mergepaths	18
1.29	move	18

---

---

1.30	movechapter . . . . .	18
1.31	movemasterpage . . . . .	19
1.32	movepage . . . . .	19
1.33	movetopage . . . . .	20
1.34	msys . . . . .	20
1.35	newchapter . . . . .	21
1.36	newdocument . . . . .	22
1.37	newmasterpage . . . . .	22
1.38	newstyletag . . . . .	23
1.39	newvariable . . . . .	23
1.40	nudge . . . . .	24
1.41	open . . . . .	24
1.42	openbusyrequester . . . . .	25
1.43	opendocument . . . . .	26
1.44	openwindow . . . . .	26
1.45	p2d . . . . .	27
1.46	pagepalette . . . . .	28
1.47	pasteobject . . . . .	28
1.48	pastetext . . . . .	29
1.49	placegraphic . . . . .	29
1.50	printdocument . . . . .	30
1.51	quit . . . . .	31
1.52	redo . . . . .	31
1.53	refresh . . . . .	32
1.54	refreshwindow . . . . .	32
1.55	removefontpath . . . . .	33
1.56	removepairkern . . . . .	33
1.57	replacetext . . . . .	33
1.58	revealwindow . . . . .	34
1.59	revertdocument . . . . .	34
1.60	rotate . . . . .	35
1.61	rx . . . . .	35

---

# Chapter 1

## PGS.MD4

### 1.1 insert

INSERT

Purpose: Inserts text at the insertion point.

Syntax: insert <text/S> [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
text is the text string to insert.  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)

Example: insert 'Now is the time' document 'project.doc'  
insert "I can't get no satisfaction."  
insert 'She said "How to use quote marks in strings."'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

### 1.2 insertbreak

INSERTBREAK

Purpose: Inserts a break at the insertion point.

Syntax: insertbreak <COLUMN | PAGE> [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
COLUMN inserts a column break. Text is forced to the top of the next column in the article.  
PAGE inserts a page break. Text is forced to the top of the next frame on the next page in the article.  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)

Example: insertbreak column

---

```
insertbreak page document 'project.doc'
```

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.3 insertchar

INSERTCHAR

Purpose: Inserts special characters at the insertion point.

Syntax: insertchar <OPENDBLQUOTE | CLOSEDBLQUOTE | OPENSINGLQUOTE |  
CLOSESINGLQUOTE | OPENDBLGUILLEMOT | CLOSEDBLGUILLEMOT |  
OPENSINGLGUILLEMOT | CLOSESINGLGUILLEMOT | INCH | FOOT | TM | SM |  
R | C | P | UNICODE number/I> [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	OPENDBLQUOTE	inserts a double open quote.
	CLOSEDBLQUOTE	inserts a double close quote.
	OPENSINGLQUOTE	inserts a single open quote.
	CLOSESINGLQUOTE	inserts a single close quote.
	OPENDBLGUILLEMOT	inserts a double open guillemot.
	CLOSEDBLGUILLEMOT	inserts a double close guillemot.
	OPENSINGLGUILLEMOT	inserts a single open guillemot.
	CLOSESINGLGUILLEMOT	inserts a single close guillemot.
	INCH	inserts a single prime.
	FOOT	inserts a double prime.
	TM	inserts a trademark symbol.
	SM	inserts a service mark symbol.
	R	inserts a registration mark symbol.
	C	inserts a copyright symbol.
	P	inserts a sound recording copyright.
	UNICODE	inserts a unicode character.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Example: insertchar.opendblquote  
insertchar foot document 'project.doc'  
insertchar unicode 8223

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.4 insertcontrol

INSERTCONTROL

Purpose: Inserts a control character at the insertion point.

Syntax: insertcontrol <TAB | RIGHTINDENTTAB | NEWLINE | NEWPARAGRAPH |  
INDENTHERE> [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	TAB	inserts a tab control.
	RIGHTINDENTTAB	inserts a right indent tab control.
	NEWLINE	inserts a newline control.
	NEWPARAGRAPH	inserts a newparagraph control.
	INDENTHERE	inserts a paragraph indent.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Example: insertcontrol tab  
 insertcontrol newline document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.5 insertdash

INSERTDASH

Purpose: Inserts a dash at the insertion point.

Syntax: insertdash <HYPHEN | NBHYPHEN | SOFTHYPHEN | EN | NBEN | SOFTEN | EM | NBEM | SOFTEM> [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	HYPHEN	inserts a normal hyphen.
	NBHYPHEN	inserts a non-breaking normal hyphen.
	SOFTHYPHEN	inserts a soft normal hyphen.
	EN	inserts an en dash.
	NBEN	inserts a non-breaking en dash.
	SOFTEN	inserts a soft en dash.
	EM	inserts an em dash.
	NBEM	inserts a non-breaking em dash.
	SOFTEM	inserts a soft em dash.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Example: insertdash soften  
 insertdash nbhyphen document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.6 insertdate

INSERTDATE

Purpose: Inserts a date variable at the insertion point.

Syntax: insertdate [CURRENT | CREATED | MODIFIED | SAVED | LOADED | PRINTED | DOCUMENTREVISION | CHAPTERREVISION | PAGEREVISION | ARTICLEREVISION] [FORMAT format/S] [LANGUAGE <DEFAULT | name/S>]

[DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter

CURRENT	inserts the current date, which is never updated. (Default)
CREATED	inserts the date when the document was created.
MODIFIED	inserts the date when the document was last modified.
SAVED	inserts the date when the document was last saved.
LOADED	inserts the date when the document was last loaded.
DOCUMENTREVISION	inserts the date when the last document revision log entry was made.
CHAPTERREVISION	inserts the date when the last chapter revision log entry was made.
PAGEREVISION	inserts the date when the last page revision log entry was made.
ARTICLEREVISION	inserts the date when the last article revision log entry was made.
DOCUMENTREVISION	inserts the date when the last document revision log entry was made.
FORMAT	is the format of the date. See below. (Default='mm/dd/yy')
LANGUAGE	is the language of the inserted date variable.
DEFAULT	specifies the default language used in the chapter or document.
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)

Notes: The options available for LANGUAGE depend on which languages are installed in the SoftLogik:Engines directory.

The FORMAT string contains the order, length and separators for the date.

/ Separate with a slash.  
 - Separate with a hyphen.  
 , Separate with a comma.  
 Separate with a space.

\ One character following a backslash or text inside double  
 " " quotation marks will appear as is.

If the first letter of a format is capitalized (Mmm), that part of the date will be capitalized (if applicable). If at least the first two letters are capitalized (MMm), it will be uppercase. If the first two letters are lowercase, it will be lowercase.

y	The last 2 digits of the year.	94, 95, 96
yy	Same as y.	
yyy	All 4 digits of the year.	1994, 1995, 1996
yyyy	Same as yyy.	
m	Number of the month, no leading zero.	1, 6, 7, 10
mm	Number of the month, leading zero.	01, 06, 07, 10
mmm	3 letter abbreviated month.	jan, jun, jul
mmmm	3-4 letter abbreviated month.	jan, june, july
mmmmm	Full name of the month.	january, june, july
d	Number of the day, no leading zero.	1, 2, 12

dd	Number of the day, leading zero.	01, 02, 12
dddd	Ordinal number of the day.	1st, 2nd, 12th
w	Number of the day of the week.	1, 2, 3, 4, 5, 6, 7
ww	1-2 letter abbreviated day of the week.	su, m, tu, w, th, f
www	3 letter abbreviated day of the week.	sun, mon, tue, wed
www	Full day of the week.	sunday, monday

Format examples:

y-m-d	94-12-19, 1-2-3
mm/dd/yy	12/19/94, 02/03/01
d/mm/yyyy	19/12/1994, 3/02/2001
Mmm d, yyyy	Dec 19, 1994, Feb 3, 2001
Wwww "the" ddd	Monday the 19th, Saturday the 3rd

```
Example: insertdate current format 'Wwww "the" ddd'
insertdate saved format 'Wwww' language 'français'
insertdate printed format 'd/Mmm/yyyy'
insertdate current document 'project.doc'
```

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.7 insertindexmark

INSERTINDEXMARK

Purpose: Inserts an index mark at the insertion point.

Syntax: insertindexmark [NAME name/s] [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	NAME	is the index entry format.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Result: The mark number is returned to the RESULT variable.

```
Example: insertindexmark name 'next, "Heading 1","Government"'
marknum=result
```

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.8 insertmark

INSERTMARK

Purpose: Inserts a text mark at the insertion point.

Syntax: insertmark [NAME name/s] [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 NAME is the name of the text mark.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Result: The mark number is returned to the RESULT variable.

Example: insertmark name "This is a test."  
 marknum=result

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.9 insertname

INSERTNAME

Purpose: Inserts a name variable at the insertion point.

Syntax: insertname <DOCUMENT | CHAPTER | MASTERPAGE | PAGE |  
 MARK number/I | INDEXMARK number/I | VARIABLE name/S >  
 [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 DOCUMENT inserts the document name.  
 CHAPTER inserts the chapter name.  
 MASTERPAGE inserts the master page name.  
 PAGE inserts the page name.  
 MARK inserts the text mark name.  
 INDEXMARK inserts the indexmark string.  
 VARIABLE inserts the user variable name.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: insertname document  
 insertname chapter document 'project.doc'

\*\* CONFLICT between document and document?

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.10 insertnumber

INSERTNUMBER

Purpose: Inserts a number variable at the insertion point.

Syntax: insertnumber <PAGE | PREVPAGE | NEXTPAGE | FIRSTPAGE |  
 LASTPAGE | PAGECOUNT | PAGESTART | PAGEEND | CHAPTER | CHAPTERCOUNT |  
 CHAPTERSTART | CHAPTEREND | SUBCHAPTERCOUNT | SUBCHAPTERSTART |  
 SUBCHAPTEREND | COLUMN | PREVCOLUMN | NEXTCOLUMN | COLUMNCOUNT |

```
MARK number/D | INDEXMARK number/D>
[FORMAT <DEFAULT | LONG | ARABIC | ROMANUPPER | ROMANLOWER |
ALPHAUPPER | ALPHALOWER>] [LANGUAGE <DEFAULT | name/S>]
[DOCUMENT name/S | WINDOW name/S]
```

Format:	Parameter	Values to enter
	PAGE	inserts the page number of the current frame.
	PREVPAGE	inserts the page number of the article's previous frame.
	NEXTPAGE	inserts the page number of the article's next frame.
	FIRSTPAGE	inserts the page number of the article's first frame.
	LASTPAGE	inserts the page number of the article's last frame.
	PAGECOUNT	inserts the number of pages.
	PAGESTART	inserts the starting page number.
	PAGEEND	inserts the ending page number.
	CHAPTER	inserts the current chapter number.
	CHAPTERCOUNT	inserts the number of chapters.
	CHAPTERSTART	inserts the starting chapter number.
	CHAPTEREND	inserts the ending chapter number.
	SUBCHAPTERCOUNT	inserts the number of subchapters.
	SUBCHAPTERSTART	inserts the starting subchapter number.
	SUBCHAPTEREND	inserts the ending subchapter number.
	COLUMN	inserts the number of the current column.
	PREVCOLUMN	inserts the number of the previous column in the series.
	NEXTCOLUMN	inserts the number of the next column in the series.
	COLUMNCOUNT	inserts the number of columns in the current series.
	MARK	inserts the text mark number.
	INDEXMARK	inserts the index mark number.
	FORMAT	specifies the number format.
	LONG	specifies long format. i.e., One, Two, Three
	ARABIC	specifies Arabic format. i.e., 1, 2, 3 (Default)
	ROMANUPPER	specifies Roman uppercase format. i.e., I, II, III
	ROMANLOWER	specifies Roman lowercase format. i.e., i, ii, iii
	ALPHAUPPER	specifies Latin uppercase format. i.e., A, B, C
	ALPHALOWER	specifies Latin lowercase format. i.e., a, b, c
	DEFAULT	specifies the default format used in the chapter or document.
	LANGUAGE	is the language of the inserted number.
	DEFAULT	specifies the default language used in the chapter or document.
	PREFIX	is the page number prefix.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Notes: These commands refer to the current document level. i.e., PAGECOUNT is the number of pages in the current chapter or subchapter, if a chapter is the current level. Otherwise it refers to the number of pages in the document.

The options available for LANGUAGE depend on which languages are installed in the SoftLogik:Engines directory.

Example: insertnumber page  
insertnumber pagecount document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.11 insertpage

INSERTPAGE

Purpose: Inserts page(s) in a document or chapter.

Syntax: insertpage [PAGE number/S] [BEFORE | AFTER] [COUNT number/I]

Format: Parameter Values to enter  
 PAGE is the reference page number. (Default=current)  
 BEFORE inserts the pages before the reference page. (Default)  
 AFTER inserts the pages after the reference page.  
 COUNT is the number of pages to insert. (Default=1)

Example: insertpage page 1 before  
 insertpage page 5 after count 2

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.12 insertrevision

INSERTREVISION

Purpose: Inserts a revision number variable at the insertion point.

Syntax: insertrevision [DOCUMENTREVISION | CHAPTERREVISION | PAGEREVISION |  
 ARTICLEREVISION] [FORMAT format/S] [LANGUAGE <DEFAULT | name/S>]  
 [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 DOCUMENTREVISION inserts the number of the last document revision  
 log entry. (Default)  
 CHAPTERREVISION inserts the number of the last chapter revision  
 log entry.  
 PAGEREVISION inserts the number of the last page revision log  
 entry.  
 ARTICLEREVISION inserts the number of the last article revision  
 log entry.  
 DOCUMENTREVISION inserts the number of the last document revision  
 log entry.  
 FORMAT is the format of the revision number. See below.  
 (Default='h:m PM')  
 LANGUAGE is the language of the inserted revision variable.  
 DEFAULT specifies the default language used in the chapter  
 or document.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Notes: The options available for LANGUAGE depend on which languages are installed in the SoftLogik:Engines directory.

The FORMAT string contains the order, length and separators for the revision number.

/ Separate with a slash.  
 - Separate with a hyphen.  
 , Separate with a comma.  
 Separate with a space.

\ One character following a backslash or text inside double  
 " " quotation marks will appear as is.

If the first letter of a format is capitalized (Mm), that part of the revision number will be capitalized (if applicable). If both letters are capitalized (MMm), it will be uppercase. If both letters are lowercase, it will be lowercase.

j	major revision number, no leading zero.	5
jj	major revision number, leading zero.	05
jjj	major, long format	one, two, three...
n	major revision number, no leading zero.	5
nn	major revision number, leading zero.	05
nnn	major, long format	one, two, three...

Format examples:

j.n	6.2
j/nn	6/02
jj.nn	06.02
jjj-nnn	six-two
Jjj."Oh"	Six.Oh
JJJ,n	SIX,2

Example: insertrevision format 'j.nn'  
 insertrevision chapterrevision format 'JJJ.nn'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.13 insertsoft

INSERTSOFT

Purpose: Inserts a soft (conditional) control at the insertion point.

Syntax: insertsoft <NEWLINE | HYPHEN | TEXT string/S> [DOCUMENT name/S | WINDOW ↵  
 name/S]

Format: Parameter Values to enter  
 NEWLINE inserts a soft newline without inserting a character.  
 HYPHEN inserts a soft normal hyphen before the newline.  
 TEXT inserts this string before the newline.  
 DOCUMENT is the document name. (Default=current)

WINDOW is the window name. (Default=current)

Notes: INSERTDASH SOFT is identical to INSERTSOFT HYPHEN.

Example: insertsoft newline  
 insertsoft hyphen document 'project.doc'  
 insertsoft text \08226 /\* inserts a bullet \*/

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.14 insertspace

INSERTSPACE

Purpose: Inserts a space at the insertion point.

Syntax: insertspace <SPACE | NBSPACE | THIN | NBTHIN | EN | NBEN | EM | NBEM> [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 SPACE inserts a normal space.  
 NBSPACE inserts a non-breaking normal space.  
 THIN inserts a thin space.  
 NBTHIN inserts a non-breaking thin space.  
 EN inserts an en space.  
 NBEN inserts a non-breaking en space.  
 EM inserts an em space.  
 NBEM inserts a non-breaking em space.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: insertspace thin  
 insertspace nben document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.15 inserttext

INSERTTEXT

Purpose: Inserts text from a text file at the insertion point.

Syntax: inserttext [FILE filepath/F] [FILTER name/S] [STATUS | NOSTATUS]  
 [CONVERTQUOTE <TRUE|FALSE>] [CONVERTDASH <TRUE|FALSE>]  
 [LINEHASLF <TRUE|FALSE>] [AMIGA | MSDOS | MACINTOSH]  
 [RETAINFORMAT <TRUE|FALSE>] [TEXTCODE name/S]  
 [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 FILE is the filename and path of the file.

FILTER	is the import/export filter to use.
STATUS	displays a status indicator while loading the file.
NOSTATUS	does not display a status indicator. (Default)
CONVERTQUOTE	converts typewriter quotation marks to typographic quotation marks. (Not applicable to IFF CTXT.) (DEFAULT=TRUE)
CONVERTDASH	converts double dashes to en dashes and triple dashes to em dashes. (Not applicable to IFF CTXT.) (DEFAULT=TRUE)
LINEHASLF	assumes that each line ends with a line feed rather than each paragraph. (Applicable to ASCII and IFF FTXT only.) (Default=FALSE)
RETAINFORMAT	imports the attributes as well as the text. (Not applicable to ASCII.) (Default=TRUE)
AMIGA	specifies that the Amiga character set should be used. (Applicable to ASCII only.) (Default)
MSDOS	specifies that the MS-DOS character set should be used. (Applicable to ASCII only.)
MACINTOSH	specifies that the Mac character set should be used. (Applicable to ASCII only.)
TEXTCODE	is the text code engine to use. (Default=NONE)
DOCUMENT	is the document name. (Default=current)
WINDOW	is the window name. (Default=current)

Notes: If the FILE parameter is not specified, a file requester will open.

If the filter parameter is not specified, the appropriate filter will be used, but the Place Graphic requester will open. Valid filter parameters at the date of this writing are: ASCII, IFFDTEXT, IFFFTXT, IFFCTXT, FinalCopy2, FinalWriter, WordWorth, ProWrite, WordPerfect and Word. The names of filters are normally the same as the filename, less the '.filter' extension. You can determine the filter name by typing 'version name.filter' in a shell, where name is the filter to examine.

Example: inserttext file text:loremipsum.txt filter ascii status

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.16 inserttime

INSERTTIME

Purpose: Inserts a time variable at the insertion point.

Syntax: inserttime [CURRENT | CREATED | MODIFIED | SAVED | LOADED | PRINTED | DOCUMENTREVISION | CHAPTERREVISION | PAGEREVISION | ARTICLEREVISION] [FORMAT format/S] [LANGUAGE <DEFAULT | name/S>] [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	CURRENT	inserts the current time, which is never updated. (Default)
	CREATED	inserts the time when the document was created.

MODIFIED inserts the time when the document was last modified.  
 SAVED inserts the time when the document was last saved.  
 LOADED inserts the time when the document was last loaded.  
 DOCUMENTREVISION inserts the time when the last document revision log entry was made.  
 CHAPTERREVISION inserts the time when the last chapter revision log entry was made.  
 PAGEREVISION inserts the time when the last page revision log entry was made.  
 ARTICLEREVISION inserts the time when the last article revision log entry was made.  
 DOCUMENTREVISION inserts the time when the last document revision log entry was made.  
 FORMAT is the format of the time. See below.  
 (Default='h:m PM')  
 LANGUAGE is the language of the inserted time variable.  
 DEFAULT specifies the default language used in the chapter or document.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Notes: The options available for LANGUAGE depend on which languages are installed in the SoftLogik:Engines directory. The language can affect the syntax. For example, times in French are usually expressed as 18h15 instead of as 6:15 PM.

The FORMAT string contains the order, length and separators for the time.

/ Separate with a slash.  
 - Separate with a hyphen.  
 , Separate with a comma.  
   Separate with a space.

\ One character following a backslash or text inside double  
 " " quotation marks will appear as is.

: separate with a colon.  
   Separate with a space.

If the first letter of a format is capitalized (Mm), that part of the time will be capitalized (if applicable). If both letters are capitalized (MMm), it will be uppercase. If both letters are lowercase, it will be lowercase.

h	hour, no leading zero.	5
hh	hour, leading zero.	05
m	minutes, no leading zero.	8
mm	minutes, leading zero.	08
s	seconds, no leading zero.	3
s	seconds, leading zero.	03
ampm	shows am or pm as appropriate.	5 am
pm	shows only pm when time>12:00.	7 pm
	no indicator specifies 24 hour time.	19

Format examples:

```

h:m          5:15, 19:30
h:m ampm    5:15 am, 7:30 pm
h:m PM      5:15, 7:30 PM
hh:mm:ss    05:15:21, 19:30:46
hh\hmm      5h15

```

```

Example: inserttime current format 'hh:mm:ss'
inserttime printed

```

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.17 liveupdate

LIVEUPDATE

Purpose: This command sets the live update and drag settings.

Syntax: liveupdate [SCROLL <ON | OFF>] [DRAG <ON | OFF>] [SAVE]

```

Format: Parameter Values to enter
        SCROLL      sets the livescroll option. (Default=current)
        DRAG        sets the livedrag option. (Default=current)
        SAVE        saves the settings to PageStream3.prefs.

```

```

Example: liveupdate scroll off drag off
liveupdate scroll on save

```

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.18 loadfontprefs

LOADFONTPREFS

Purpose: This loads a font prefs file and replaces the font paths in the Font Preferences requester.

Syntax: loadfontprefs <FILE filepath/F | DEFAULT>

```

Format: Parameter Values to enter
        FILE        is the filename and path of the fontprefs file to load.
        DEFAULT     loads the PageStream3.fontprefs file.

```

```

Example: loadfontprefs file 'NewsletterFonts.fontprefs'
loadfontprefs default

```

Command Format

## 1.19 loadprintprefs

### LOADPRINTPREFS

**Purpose:** This loads the printing preferences from a saved settings file.

**Syntax:** loadprintprefs <DEFAULT | FILE filepath/F>

**Format:**

Parameter	Values to enter
DEFAULT	loads the printing preferences from the default PageStream3:PageStream3.prefs file.
FILE	is the filename and path of the settings file to load.

**Notes:** You can load printing preferences from a pref file that contains more than just printing preferences items.

**Example:** loadprintprefs default  
loadprintprefs file 'PageStream3:ServiceBureau.prefs'

Command Format

## 1.20 loadsettings

### LOADSETTINGS

**Purpose:** This loads program settings from a saved settings file.

**Syntax:** loadsettings <DEFAULT | FILE filepath/F>

**Format:**

Parameter	Values to enter
DEFAULT	loads the default PageStream3:PageStream3.prefs file.
FILE	is the filename and path of the settings file to load.

**Example:** loadsettings default  
loadsettings file 'PageStream3:MyPrefs'

Command Format

## 1.21 loadsnapshot

### LOADSNAPSHOT

**Purpose:** This loads the default position and size of the window and all palettes.

**Syntax:** loadsnapshot <DEFAULT | FILE filepath/F>

**Format:**

Parameter	Values to enter
DEFAULT	loads the values in the PageStream3.prefs file.
FILE	is the filename and path of the settings file to load.

**Example:** loadsnapshot default

---

```
loadsnapshot file 'PageStream3:tiled.prefs'
```

Command Format

## 1.22 lock

LOCK

Purpose: Locks selected objects so that they cannot be edited.

Syntax: lock [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)  
OBJECTID is the number of the object to manipulate.

Example: lock  
lock document 'project.doc'

See also UNLOCK

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.23 lockinterface

LOCKINTERFACE

Purpose: Locks the PageStream interface to prevent the user from doing anything.

Syntax: lockinterface <TRUE | FALSE>

Format: Parameter Values to enter  
TRUE locks the interface.  
FALSE unlocks the interface.

Example: lockinterface true

Command Format

## 1.24 logrevision

LOGREVISION

Purpose: Adds revision history and increases the revision number for a document, chapter, page or masterpage.

---

Syntax: logrevision [DESCRIPTION text/S]  
 [VERSION major/I minor/I | BUMPREV] [CREATED | MODIFIED]  
 [USER username/S]  
 [CHAPTER name/S | PAGE number/S | MASTERPAGE name/S]

Format: Parameter Values to enter  
 DESCRIPTION is the text to add to the revision history.  
 VERSION is the major and minor version numbers.  
 BUMPREV increments the revision number by 1 minor version.  
 (Default)  
 CREATED is a creation revision type.  
 MODIFIED is a modification revision type.  
 USERNAME is the name of the user saving the document.  
 CHAPTER is the document/chapter name. (Default=current)  
 PAGE is the page name. (Default=current)  
 MASTERPAGE is the master page name. (Default=current)

Notes: Revision numbers are tracked by major and minor versions in the format of major.minor. i.e., 2.1.

Example: logrevision bumprev  
 logrevision description "Fixed color problem" user "Colleen"

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.25 macropalette

MACROPALETTE

Purpose: Displays the macro palette.

Syntax: macropalette [SHOW | HIDE | TOGGLE] [AT pointx/I pointy/I]  
 [SIZE width/I height/I] [SAVE]

Format: Parameter Values to enter  
 SHOW toggles on the palette.  
 HIDE toggles off the palette.  
 TOGGLE toggles the palette on and off.  
 AT sets the palette position.  
 SIZE sets the palette size in pixels. (Min=160x50)  
 SAVE saves the values to PageStream3.prefs.

Note: The TOGGLE parameter may not be used in conjunction with the SAVE parameter.

Example: macropalette toggle  
 macropalette show at 500 300 size 180 172 save  
 macropalette hide

Command Format

## 1.26 maintoolbox

MAINTOOLBOX

Purpose: Displays the Main toolbox.

Syntax: `maintoolbox [SHOW | HIDE | TOGGLE] [AT pointx/I pointy/I]  
[LARGE | SMALL] [VERTICAL | HORIZONTAL] [SAVE]`

Format: Parameter Values to enter

SHOW	toggles on the toolbox.
HIDE	toggles off the toolbox.
TOGGLE	toggles the toolbox on and off.
AT	sets the toolbox position.
LARGE	sets the toolbox size to large.
SMALL	sets the toolbox size to small.
VERTICAL	sets the toolbox orientation to vertical.
HORIZONTAL	sets the toolbox orientation to horizontal.
SAVE	saves the toolbox position to PageStream3.prefs.

Note: The TOGGLE parameter may not be used in conjunction with the SAVE parameter.

Example: `maintoolbox show horizontal`  
`maintoolbox show large at 500 300 save`  
`maintoolbox hide`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.27 makepagespread

MAKEPAGESPREAD

Purpose: Creates a page spread of two or more pages.

Syntax: `makepagespread [PAGE number/S] [TO number/S]`

Format: Parameter Values to enter

PAGE	is the number of the first page in the spread. (Default=current)
TO	is the number of the last page in the spread.

Notes: If any of the pages specified are already part of a spread, that spread will be dissolved automatically. To dissolve a page spread, leave out the 'to' parameter.

Example: `makepagespread page 1 to 3`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

---

## 1.28 mergepaths

MERGEPATHS

Purpose: Merge the paths of selected object(s) and path(s).

Syntax: mergepaths [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: mergepaths  
 mergepaths document 'project.doc'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.29 move

MOVE

Purpose: Move selected object(s).

Syntax: move [OFFSET movex/D movey/D] [SNAP | NOSNAP]  
 [DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format: Parameter Values to enter  
 OFFSET overrides the default move offsets.  
 SNAP snaps the object to the closest grid points after moving it.  
 NOSNAP ignores the snapto grid settings.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: move offset 2.5 3.125  
 move offset 2.5 -3 snapto grid document 'project.doc'

See also SETMOVEDEFAULTS.

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.30 movechapter

MOVECHAPTER

Purpose: Moves a chapter in a document.

Syntax: `movechapter <BEFORE chapternumber/I | AFTER chapternumber/I>`  
`[CHAPTER name/S | WINDOW name/S]`

Format: Parameter Values to enter  
 BEFORE is the chapter before which to insert the chapter.  
 AFTER is the chapter after which to insert the chapter.  
 CHAPTER is the document/chapter name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: `movechapter before 3`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.31 movemasterpage

MOVEMASTERPAGE

Purpose: Moves a masterpage anywhere within a document.

Syntax: `movemasterpage [MASTERPAGE name/S] [CHAPTER name/S]`

Format: Parameter Values to enter  
 MASTERPAGE is the master page name. (Default=current)  
 CHAPTER is the document/chapter name. (Default=current)

Note: If you move a master page laterally (to another chapter at the same level), which makes it unavailable to pages in the current chapter, they will use the next higher level's master page with the same name, if available. If a master page with the same name does not exist at a higher level, these pages will revert to the default master page. If you move a master page down a level, this will cause pages using that master page to revert to the default master page.

Example: `movemasterpage masterpage "TwoColumns" chapter "Introduction"`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.32 movepage

MOVEPAGE

Purpose: Moves page(s) in a document or chapter.

Syntax: `movepage [PAGE number/S] [TO number/S]`  
`<BEFORE pagenumber/I | AFTER pagenumber/I>`

Format: Parameter Values to enter  
 PAGE is the number of the first page in the range to move.

(Default=current)  
 TO is the number of the last page in the range to move.  
 BEFORE is the page before which to insert the page.  
 AFTER is the page after which to insert the page.

Notes: You cannot move a page outside of its chapter or document. You can copy pages and paste them into other chapters or documents if desired.

Example: `movepage page 5 to 8 before 17`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

### 1.33 movetopage

MOVETOPAGE

Purpose: Move the selected object(s) to another page.

Syntax: `movetopage <PAGE number/S | MPG name/S | WINDOW name/S>  
 [INFRONT | INBACK]  
 [FROM <DOCUMENT name/S | WINDOW name/S | OBJECTID number/I>]`

Format: Parameter Values to enter  
 PAGE is the number of the page to move to.  
 MPG is the master page name and side to move to.  
 WINDOW is the window displaying the page to move to.  
 INFRONT moves the object to the top of the stack. (Default)  
 INBACK moves the object to the bottom of the stack.  
 FROM specifies the document or window of the object.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: `movetopage inback from document 'project.doc'  
 movetopage page 15  
 movetopage page 'Bill.letter~7'`

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

### 1.34 msys

MSYS

Purpose: Returns the measurement system abbreviation of the specified coordinate.

Syntax: `msys(coord/S)`

Format: Parameter Values to enter  
 coord is the coordinate to get the measurement system abbreviation of.

Note: This is a function, not a command. It is only available when the Soft-Logik ARExx support library is added to your ARExx script. i.e.,

```
addlib("softlogik:libs/slarexxsupport.library", 0, -30)
```

Returns: The returned string will be set to the appropriate abbreviation:

INCHES	i
CENTIMETERS	cm
MILLIMETERS	mm
PICAS	p
POINTS	pt
PRINTERPICAS	pp
PRINTERPOINTS	ppt
CICEROS	c
DIDOTPOINTS	d
FEET	f
METERS	m

Example: `getbox position coord /* gets the coordinates of a selected box */`  
`say msys(coord.left) /* will print the measurement system abbreviation of the left coordinate to the output console */`

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.35 newchapter

NEWCHAPTER

Purpose: Creates a new chapter in a document or chapter.

Syntax: `newchapter <chaptername/S chapternumber/I>`  
`[CHAPTER name/S | WINDOW name/S]`

Format: Parameter Values to enter  
 chaptername is the new chapter name.  
 chapternumber is the new chapter number.  
 CHAPTER is the document/chapter name. (Default=current)  
 WINDOW is the window name. (Default=current)

Result: Returns the new chapter's name and path to the RESULT variable. If the chapter name specified for the new chapter is already used by an existing chapter, PageStream will append .# to it, so you may wish to check the result.

Example: `newchapter 'Part Three' 7`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

---

## 1.36 newdocument

### NEWDOCUMENT

**Purpose:** Creates a new document.

**Syntax:** newdocument <documentname/S>

**Format:** Parameter Values to enter  
documentname is the new document's name.

**Result:** Returns the new document's name to the RESULT variable. If the document name specified for the new document is already used by an open document, PageStream will append .# to it, so you may wish to check the result.

**Note:** This command is not equivalent to the menu's New command. It does not create a default master page or open a document window. To create a master page, follow this command with a  
NEWMASTERPAGE  
command, and then open its window with the  
OPENWINDOW  
command.

**Example:** newdocument 'Report.document'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.37 newmasterpage

### NEWMASTERPAGE

**Purpose:** Creates a new master page in a document or chapter.

**Syntax:** newmasterpage <masterpagename/S width/D height/D>  
[PORTRAIT | LANDSCAPE] [SINGLE | DOUBLE]  
[CHAPTER name/S | WINDOW name/S]

**Format:** Parameter Values to enter  
masterpagename is the new master page name.  
width is the master page width.  
height is the master page height.  
PORTRAIT is normal page orientation. (Default)  
LANDSCAPE reverses the page width and height.  
SINGLE is a single-sided document. (Default)  
DOUBLE is a double-sided document.  
CHAPTER is the document/chapter name. (Default=current)  
WINDOW is the window name. (Default=current)

**Note:** This command is not equivalent to the menu's Master Page Setup command. To set all the options in that requester you must also use the SETBLEED, SETMARGINGUIDES and SETCOLUMNGUIDES commands. The parameters set in this command

can be changed later with the SETDIMENSIONS command.

Example: newmasterpage '2 Column Layout' 8.5 11 landscape double

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.38 newstyletag

NEWSTYLETAG

Purpose: Creates a new character, paragraph, object or color style tag.

Syntax: newstyletag <stylename/S>  
 <CHARACTER | PARAGRAPH | OBJECT | COLOR>  
 [DOCUMENT name/S | WINDOW name/S | CHAPTER name/S]

Format: Parameter Values to enter  
 stylename is the name of the new style tag.  
 CHARACTER creates a character style tag.  
 PARAGRAPH creates a paragraph style tag.  
 OBJECT creates an object style tag.  
 COLOR creates a color style tag.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 CHAPTER is the document/chapter name. (Default=current)

Notes: Named colors are actually a type of style tag, thus their inclusion in this macro.

To add attributes to a style tag, set the attribute and append the STYLETAG keyword and style name. See each attribute command for more information. Note that while you can theoretically add paragraph attributes to a character style tag through a macro, this is not recommended, is not supported through the interface, and could cause problems in future versions of PageStream. Use each type of style tag for its intended purpose.

Style tag names must be unique; you cannot create a character style and a color style with the same name.

Example: newstyletag 'Body Text' paragraph  
 newstyletag 'Computer Style' character  
 newstyletag 'Basic Objects' object

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.39 newvariable

NEWVARIABLE

---

Purpose: Creates a new user variable.

Syntax: `newvariable <varname/S varvalue/S>`  
`[DOCUMENT name/S | WINDOW name/S | CHAPTER name/S]`

Format: Parameter Values to enter  
 varname is the name of the new variable.  
 varvalue is the variable string.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 CHAPTER is the document/chapter name. (Default=current)

Example: `newvariable author "Bill Watson"`

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.40 nudge

NUDGE

Purpose: Nudge object(s) in any direction by a preset amount.

Syntax: `nudge <[LEFT | RIGHT] | [ UP | DOWN]> [TAP]`  
`[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]`

Format: Parameter Values to enter  
 LEFT nudges objects to the left.  
 RIGHT nudges objects to the right.  
 UP nudges objects upwards.  
 DOWN nudges objects downwards.  
 TAP nudges the object 1/10th of the normal nudge amount.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)  
 OBJECTID is the number of the object to manipulate.

Example: `nudge left up document 'project.doc'`  
`nudge down`

See also SETNUDGEDEFAULTS.

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.41 open

OPEN

Purpose: This opens a document and a window.

Syntax: `open <filepath/F>`

---

Format: Parameter Values to enter  
 filepath is the filename and path of the document to open.

Example: open 'PageStream3:documents/Project.doc'

See also

OPENDOCUMENT  
 and  
 OPENWINDOW  
 .

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.42 openbusyrequester

OPENBUSYREQUESTER

External macros only!

Purpose: Displays a busy requester from an ARexx script. Use this to provide feedback during a lengthy script process. Returns the requester handle to RESULT.

Syntax: openbusyrequester [MESSAGE text/S]  
 [THERMOMETER <ENABLED | DISABLED>] [ABORT <ENABLED | DISABLED>]  
 [TOTAL value/I] [CURRENT value/I]

Format: Parameter Values to enter  
 MESSAGE is the single line of text to display.  
 THERMOMETER sets whether the thermometer gadget is used.  
 ABORT sets whether the Stop gadget is available.  
 TOTAL sets the thermometer's total value.  
 CURRENT sets the thermometer's current value.

Result: The handle (identification number) of the busy requester is returned to the RESULT variable. The handle is required in order to set the thermometer value and message text later, and to get the state of the Stop gadget.

Example: openbusyrequester message 'Thinking hard...' thermometer enabled total ↔  
 100 current 0  
 bh=RESULT  
 do for i = 1 to 100  
 /\* script does something here \*/  
 /\* abort if stop is pressed, else set thermometer \*/  
 getbusyrequester bh  
 if result=1 then break else setbusyrequester bh current i  
 end i  
 closebusyrequester bh

Command Format

## 1.43 opendocument

### OPENDOCUMENT

**Purpose:** This opens a document without opening a window.

**Syntax:** opendocument [FILE filepath/F] [FILTER name/S] [STATUS | NOSTATUS]

**Format:**

Parameter	Values to enter
FILE	is the filename and path of the document to open.
FILTER	is the open/save filter to use.
STATUS	displays a status indicator while loading the file.
NOSTATUS	does not display a status indicator. (Default)

**Notes:** This command is not equivalent to the menu's Open command. It opens a document file from disk like Open, but does not open a window for the document. To open a window, use the OPENWINDOW command. To change the page displayed in an open window, use the DISPLAY command.

If the filter parameter is not specified, the appropriate filter will be used. Valid filter parameters at the date of this writing are: IFFDOC, PageStream2, ProPage and WordworthDoc. You can determine the filter name by typing 'version name.filter' in a shell, where name is the filter to examine.

If the file parameter is not specified, a file requester will open.

**Example:** opendocument 'PageStream3:documents/Project.doc' status  
opendocument filter 'PGS2.filter'

See also

OPEN  
and  
OPENWINDOW  
.

Command Format

## 1.44 openwindow

### OPENWINDOW

**Purpose:** This opens a window for an open document.

**Syntax:** openwindow <windowname/S> [DOCUMENT name/S]  
[PAGE number/S | MPG name/S]  
[SCALE <custom/P | FULLPAGE | FULLBLEED | FULLPAGEWIDTH | FULLBLEEDWIDTH | FULLPAGEHEIGHT | FULLBLEEDHEIGHT>]  
[OFFSET offsetx/D offsety/D]

**Format:** Parameter Values to enter

windowname is the name of the new window.  
 DOCUMENT is the document name. (Default=current)  
 PAGE is the page number to display.  
 MPG is the master page name and side to display.  
 SCALE is the a predefined or custom view scale. (Default=current)  
 OFFSET is the page offset from the top left of the window. (Default= ←  
 current)

Note: This command is not equivalent to the menu's Open command. It does not open a document file from disk, but instead, opens a window for an open document. To open a document from disk, use the

OPENDOCUMENT  
 command. To change the page displayed in an  
 open window, use the DISPLAY command.

Example: openwindow 'MyWindow' document 'Untitled Document.1' page 6  
 openwindow 'Workspace' scale fullpage

See also

OPENDOCUMENT  
 and  
 OPEN  
 .

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.45 p2d

P2D

Purpose: Strips the measurement system abbreviation from a coordinate and optionally converts it to another measurement system.

Syntax: p2d(coord/S[,system/S])

Format: Parameter Values to enter  
 coord is the coordinate to strip and convert.  
 system is the measurement abbreviation of the system to which  
 to convert the coordinate. (Default=current)

Note: This is a function, not a command. It is only available when the Soft-Logik ARexx support library is added to your ARexx script. i.e.,

addlib("softlogik:libs/slarexxsupport.library", 0, -30)

Returns: The SYSTEM parameter can be set to one of these abbreviations:

INCHES	i
CENTIMETERS	cm
MILLIMETERS	mm
PICAS	p
POINTS	pt
PRINTERPICAS	pp
PRINTERPOINTS	ppt

```

CICEROS      c
DIDOTPOINTS  d
FEET         f
METERS       m

```

```

Example: getbox position coord /* gets the coordinates of a selected box */
coord.right=p2d(coord.right,'i')+2 /* adds 2 to the width in inches */
coord.right=coord.right||'i'      /* appends the inch abbreviation to ←
    ensure the value is interpreted correctly, regardless of the ←
    measurement system in use */
editbox position coord.left coord.top coord.right coord.bottom /* resets ←
    the coordinates */

```

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.46 pagepalette

PAGEPALETTE

Purpose: Displays the page palette.

Syntax: pagepalette [SHOW | HIDE | TOGGLE] [AT pointx/I pointy/I]  
 [SIZE width/I height/I] [SAVE]

Format: Parameter Values to enter

SHOW	toggles on the palette.
HIDE	toggles off the palette.
TOGGLE	toggles the palette on and off.
AT	sets the palette position.
SIZE	sets the palette size in pixels. (Min=246x100)
SAVE	saves the values to PageStream3.prefs.

Note: The TOGGLE parameter may not be used in conjunction with the SAVE parameter.

```

Example: pagepalette toggle
pagepalette show at 100 200 size 400 200 save
pagepalette hide

```

Command Format

## 1.47 pasteobject

PASTEOBJECT

Purpose: Pastes the objects from the clipboard onto the current page. If there is text in the clipboard, the command will fail.

Syntax: pasteobject [STATUS | NOSTATUS] [AT x/I y/I]  
 [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 STATUS displays a status indicator while pasting.  
 NOSTATUS does not display a status indicator. (Default)  
 AT the position at which to center the pasted object.  
 (Default=position it was copied from)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: pasteobject at 3.25i 4.125i  
 pasteobject status document 'project/doc'

See also COPYOBJECT, CUTOBJECT.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.48 pastetext

PASTETEXT

Purpose: Pastes the text from the clipboard at the insertion point. If there are objects in the clipboard, the command will fail.

Syntax: pastetext [STATUS | NOSTATUS] [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 STATUS displays a status indicator while pasting.  
 NOSTATUS does not display a status indicator. (Default)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: pastetext  
 pastetext status document 'project/doc'

See also COPYTEXT and CUTTEXT.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.49 placegraphic

PLACEGRAPHIC

Purpose: Places a graphic file into a document.

Syntax: placegraphic [FILE filepath/F] [FILTER name/S] [STATUS | NOSTATUS]  
 [AT x/I y/I] [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 FILE is the filename and path of the file.  
 FILTER is the import/export filter to use.

STATUS displays a status indicator while loading the file.  
 NOSTATUS does not display a status indicator. (Default)  
 AT the position at which to center the placed graphic.  
 (Default=center of window)  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Notes: If the file parameter is not specified, a file requester will open.

If the filter parameter is not specified, the appropriate filter will be used, but the Place Graphic requester will open. Valid filter parameters at the date of this writing are: IFFILBM, TIFF, GIF, MacPaint, IFFDR2D, IFFILUS, ProDraw. The names of filters are normally the same as the filename, less the '.filter' extension. You can determine the filter name by typing 'version name.filter' in a shell, where name is the filter to examine.

Example: placegraphic 'PageStream3:Graphics/Logo.EPS' at 3.25i 4.125i status

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.50 printdocument

PRINTDOCUMENT

Purpose: Prints the current document.

Syntax: printdocument [SIDES <BOTH | EVEN | ODD>]  
 [SCALE <ACTUAL | FULLPAGE | scale/P>] [COPIES number/I]  
 [THUMBNAILS <ON|OFF>] [TILING <ON [TILINGOVERLAP amount/D] | OFF]  
 [PICTURES <ON|OFF>] [DRAWINGS <ON|OFF>] [COLLATE <ON|OFF>]  
 [REVERSEORDER <ON|OFF>] [PRINTBLANK <ON|OFF>] [MIRROR <ON|OFF>]  
 [PRINTERMARKS <ON|OFF>] [NEGATIVE <ON|OFF>]  
 [OUTPUT <GRAYSCALE | COLOR | SEPARATIONS | PROCESS | SPOT]  
 [PAGE number/S | MPG name/S | PAGERANGE range/S | DOCUMENT name/S |  
 CHAPTER name/S]

Format:	Parameter	Values to enter
	SIDES	is which sides to print. (Default=BOTH)
	SCALE	is the scale at which to print. (Default=100%)
	COPIES	is the number of copies to print. (Default=1)
	THUMBNAILS	prints multiple pages on a sheet (page size multiplied by scale must be less than 50% for thumbnails.) (Default=OFF)
	TILING	prints the page on as many sheets as required. (Default=OFF)
	TILINGOVERLAP	sets how much the tiles will overlap by. (Default=maximum for minimum number of sheets)
	PICTURES	sets whether pictures will be printed. (Default=ON)
	DRAWINGS	sets whether drawings will be printed. (Default=ON)
	COLLATE	sets whether multiple copies will be sorted. (Default=ON)
	REVERSEORDER	sets whether the pages will be printed in reverse order. (Default=OFF)

PRINTBLANK	sets whether blank pages will be printed. (Default=ON)
MIRROR	sets whether pages will be printed mirrored. (Default=OFF)
PRINTERMARKS	sets whether printers' marks will be printed. (Default=OFF)
NEGATIVE	sets whether pages will be printed as a negative. (Default=OFF)
OUTPUT	sets whether pages will be printed in grayscale, color, or color separated (spot or process). Color is only available for color printers. (Default=GRAYSCALE)
PAGE	is the number of the page to print. (Default)
MPG	is the master page name and side to print.
PAGERANGE	is the pages to print.
DOCUMENT	is the name of the document to print.
CHAPTER	is the name of the chapter to print.

Example: print copies 2 collate off pagerange Project.doc 1-5,7,9-14  
 print sides even chapter Astronomy.doc:Planets  
 print page ""

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.51 quit

QUIT

Purpose: Quit PageStream.

Syntax: quit [FORCE | ALERT | QUIET]

Format:	Parameter	Values to enter
	FORCE	ignores the saved status of the open documents.
	ALERT	opens an alert requester if the document has been changed.
	QUIET	ignores the saved status of the document but returns an error if the document has been changed. (Default)

Example: quit force

Command Format

## 1.52 redo

REDO

Purpose: Redoes the last command or action undone.

Syntax: redo [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter

DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

Example: redo document 'project.doc'

See also UNDO.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.53 refresh

REFRESH

Purpose: Toggles screen updates so that commands can be executed without causing a screen redraw.

Syntax: refresh <OFF | ON | WAIT | CONTINUE>  
 [ALL | DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter

OFF	turns refresh off and does not remember what needs to be updated.
ON	turns refresh on and undo all WAITed refreshes.
WAIT	will pause refreshes and remembers what needs to be updated.
CONTINUE	will unpaue refreshes and will update what was last WAITed.
ALL	sets all windows. (Default)
DOCUMENT	is the document name.
WINDOW	is the window name.

Example: refresh off  
 refresh continue document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.54 refreshwindow

REFRESHWINDOW

Purpose: Causes the whole window to refresh. If refresh is off or suspended, it will queue the refresh.

Syntax: refreshwindow [WINDOW name/S]

Format: Parameter Values to enter

WINDOW	is the window name. (Default=current)
--------	---------------------------------------

Example: refreshwindow window 'Untitled Document.1~View.1'

See also

Refresh

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.55 removefontpath

REMOVEFONTPATH

Purpose: This deletes an existing font directory path.

Syntax: removefontpath <filepath/F>

Format: Parameter Values to enter  
filepath is the filepath of the font path to delete.

Example: removefontpath 'Fonts:\_Bullet\_Outlines'  
removefontpath 'work:psfonts/AdobeFonts'

Command Format

## 1.56 removepairkern

REMOVEPAIRKERN

Purpose: Remove manual kerning for the selected text.

Syntax: removepairkern [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
DOCUMENT is the document name. (Default=current)  
WINDOW is the window name. (Default=current)

Example: removepairkern document 'project.doc'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.57 replacetext

REPLACETEXT

Purpose: Searches for a text string and replaces it with another string.

Syntax: replacetext \*\*

Format: Parameter Values to enter

Example: \*\*

---

See also FINDTEXT.

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.58 revealwindow

REVEALWINDOW

Purpose: Reveals one or more hidden windows.

Syntax: revealwindow [CURRENT | ALL | WINDOW name/S]

Format: Parameter Values to enter  
 CURRENT will bring the current window to the front.  
 ALL will reveal all hidden windows.  
 WINDOW will reveal a specific hidden window or bring a specific window to the front.

Example: revealwindow current  
 revealwindow window 'Untitled Document.1~View.1'

See also HIDEWINDOW

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.59 revertdocument

REVERTDOCUMENT

Purpose: This reopens the last saved version of a document, cancelling changes that have been made.

Syntax: revertdocument [FILTER name/S] [STATUS | NOSTATUS]  
 [FORCE | ALERT | QUIET] [DOCUMENT name/S | WINDOW name/S]

Format: Parameter Values to enter  
 FILTER is the open/save filter to use.  
 STATUS displays a status indicator while loading the file.  
 NOSTATUS does not display a status indicator. (Default)  
 FORCE ignores changes that have been made.  
 ALERT opens an alert if there are changes that will be lost. (Default)  
 QUIET ignores changes that have been made, but returns an error if changes are lost.  
 DOCUMENT is the document name. (Default=current)  
 WINDOW is the window name. (Default=current)

If the filter parameter is not specified, the appropriate filter will be used. Valid filter parameters at the date of this writing are: IFFDOC, PageStream2, ProPage and WordworthDoc. You can

determine the filter name by typing 'version name.filter' in a shell, where name is the filter to examine.

Example: revertdocument status quiet  
revertdocument filter 'PGS2.filter'

Command Format

## 1.60 rotate

ROTATE

Purpose: Rotate selected objects.

Syntax: rotate <angle/A> [ABOUT pointx/D pointy/D | ABOUTCENTER]  
[DOCUMENT name/S | WINDOW name/S | OBJECTID number/I]

Format:	Parameter	Values to enter
	angle	is the rotation angle.
	ABOUT	is the rotation point.
	ABOUTCENTER	rotates around its center. (Default)
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)
	OBJECTID	is the number of the object to manipulate.

Example: rotate 90 about 2.2 3.125 document 'project.doc'

Command Format

Object ID numbers

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE

## 1.61 rx

RX

Purpose: This executes an external ARExx macro.

Syntax: rx [filepath/F] [DOCUMENT name/S | WINDOW name/S]

Format:	Parameter	Values to enter
	filepath	is the filename and path of the macro to execute.
	DOCUMENT	is the document name. (Default=current)
	WINDOW	is the window name. (Default=current)

Example: rx 'PageStream3:Macros/Text.REXX'

Command Format

DOCUMENT, CHAPTER, WINDOW, PAGE, MASTERPAGE, MPG, STYLETAG & ARTICLE