

**01261410-0**

Rodrigo Reyes

**COLLABORATORS**

	<i>TITLE :</i> 01261410-0		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Rodrigo Reyes	July 26, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>01261410-0</b>	<b>1</b>
1.1	GfxLab : ARexx port	1
1.2	Port ARexx : LOAD	2
1.3	Port ARexx : Save	2
1.4	Port ARexx : Colour To Gray	3
1.5	Port ARexx : Brightness	3
1.6	Port ARexx : Filter	4
1.7	Port ARexx : Bicolor	4
1.8	Port ARexx : Flip	5
1.9	Port ARexx : Crop	5
1.10	Port ARexx : Scale	6
1.11	Port ARexx : Negative	6
1.12	Port ARexx : Visual	7
1.13	Port ARexx : Emboss	7
1.14	Port ARexx : Sketch	8
1.15	Port ARexx : Quake	9
1.16	Port ARexx : Disperse Pixel	9
1.17	Port ARexx : Convolve	10
1.18	Port ARexx : Remove Isolated Pixels	10
1.19	Port ARexx : Quantize	11
1.20	Port ARexx : Load Palette	11
1.21	Port ARexx : Save Palette	11
1.22	Port ARexx : Count Colors	12
1.23	Port ARexx : GetWidth	12
1.24	Port ARexx : GetHeight	13
1.25	Port ARexx : FastColorMap	13
1.26	Port ARexx : Request	13
1.27	Port ARexx : Printinfo	14
1.28	Port ARexx : GetFromPattern	15
1.29	Port ARexx : BalanceFilter	15

---

1.30 Port ARexx : Rotate . . . . .	16
1.31 Port ARexx : Fade . . . . .	16
1.32 Port ARexx : CopyBrush . . . . .	17
1.33 Port ARexx : CreatePicture . . . . .	18
1.34 Port ARexx : AddText . . . . .	18
1.35 Port ARexx : TextColor . . . . .	19
1.36 Port ARexx : TextFont . . . . .	20
1.37 Port ARexx : TextSize . . . . .	20
1.38 Port ARexx : ClearBank . . . . .	20
1.39 Port ARexx : BitmapToRaw . . . . .	21
1.40 Port ARexx : TransparentColor . . . . .	21
1.41 Port ARexx : DisplayMode . . . . .	22
1.42 Port ARexx : Whirl . . . . .	23
1.43 Port ARexx : Noise . . . . .	23
1.44 Port ARexx : IsBitmap . . . . .	24
1.45 Port ARexx : Is24Bits . . . . .	24
1.46 Port ARexx : Mosaic . . . . .	25

---

# Chapter 1

## 01261410-0

### 1.1 GfxLab : ARexx port

AREXX PORT

=====

The ARexx port of GfxLab24 allows to easily automate boring processing of pictures.

The ARexx programm should be launched in an extern shell as follow:

[Open a shell or a cli]

[type as follow]

rx <SCRIPT NAME> [PARAMETERS]

Here is a list of the ARexx commands of GfxLab24. They interest you only if you are able to write ARexx scripts. Otherwise, just use the scripts included in the package.

The parameters in <> are necessary. Those in [ ] are optional.

Load      Emboss      GetWidth      TextColor  
Save      Sketch      FastColorMap      TextFont  
ColourToGray      Quake      Request      TextSize  
Brightness      DispersePixel      PrintInfo      ClearBank  
Filter      Convolve      GetFromPattern      BitmapToRaw  
BiColor      RemIsolated      BalanceFilter      TransparentColor  
Flip      Quantize      Rotate      DisplayMode  
Crop      LoadPalette      Fade      Whirl  
Scale      SavePalette      CopyBrush      Noise  
Negative      CountColor      CreatePicture      Is24Bits  
Visual      GetHeight      AddText      IsBitmap  
Mosaic

---

## 1.2 Port ARexx : LOAD

Nom de la commande: Load

Syntax: Load <NOM> [BANK 1-5]

Parameters:

NAME : Name of the file to load.

All the formats supported by GfxLab24 are accepted.

BANK [1-5] : The command operates on the specified bank number.

Result:

Return 'OK' if no error occurred, otherwise Return a message beginning with "ERROR", and which explains what happened.

Example:

```
LOAD 'dh5:MyFile'
```

## 1.3 Port ARexx : Save

Nom de la commande: Save

Syntax: Save <NOM> [FORMAT]

Parameters:

NAME: Name of the file to save. This name can contain a path, otherwise the current GfxLab24 path is used.

FORMAT: This optionnal parameter allows to save the picture in a specified format. FORMAT can be either "JPEG", "IFF24", "IFF", "TARGA", or "PCX".

The default format, if not specified, is IFF24.

BANK [1-5] : The command operates on the specified bank number.

Result:

Returns 'OK' if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
Save 'df0:MyFile' JPEG
```

---

## 1.4 Port ARexx : Colour To Gray

Nom de la commande: ColorToGray

Syntax: ColorToGray <RED> <GREEN> <BLUE> [BANK 1-5]

ou ColorToGray <PAL | HDTV> [BANK 1-5]

Parameters:

<RED> : Red component weighting, contained between 0 and 10000.

The default is 2989.

<GREEN>: Green component weighting. The default is 5866.

<BLUE> : Blue component weighting. The default is 1144.

<PAL | HDTV>: If one of the keyword "PAL" or "HDTV" is found, the components weighting corresponds to the specified format.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

ColorToGray 1200 5600 7899

ColorToGray PAL BANK 3

## 1.5 Port ARexx : Brightness

Nom de la commande: Brightness

Syntax: Brightness <VALEUR>

Parameters:

VALUE: Can be any numerical value between 1 and 400. This can also be a keyword: "HALF", which means 50, or "DOUBLE" which means 200.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

Brightness 75

Brightness HALF

## 1.6 Port ARexx : Filter

Nom de la commande: Filter

Syntax: Filter <RED> <GREEN> <BLUE> [BANK 1-5]

ou Filter <"RED" | "GREEN" | "BLUE"> [BANK 1-5]

Parameters:

<RED> : Weighting of the red component that must be filtered.

<GREEN> : Weighting of the green component that must be filtered.

<BLUE> : Weighting of the blue component that must be filtered.

<"RED" | "GREEN" | "BLUE"> : If one of this keywords are used, the weighting used correspond to the specified preset filter.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Filter RED BANK 2

Filter 110 80 60 BANK 1

Filter GREEN

Filter 100 50 100

## 1.7 Port ARexx : Bicolor

Nom de la commande: BiColor

Syntax: Bicolor [METHOD] [THRESHOLD]

Parameters:

[METHOD], is a keyword which can be any of the following:

"NONE" (no dithering), is the default value

"FLOYD" (Floyd-Steinberg)

"BURKES"

"STUCKI"

"SIERRA"

"JARVIS"

"STEVENSON"

"EDGES"

"BOUNDARY"

[THRESHOLD] is a number which represents the minimal threshold between two colours.

---



[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

BiColor

BiColor BOUNDARY BANK 1

## 1.8 Port ARexx : Flip

Nom de la commande: Flip

Syntax: Flip [DIRECTION] [BANK 1-5]

Parameters:

[DIRECTION] : Can be any of the keywords "HORIZONTAL" or "VERTICAL". The default for this operator is "HORIZONTAL".

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Flip VERTICAL BANK 3

Flip

Flip Bank 2

Flip HORIZONTAL

## 1.9 Port ARexx : Crop

Nom de la commande: Crop

Syntax: Crop <X> <Y> <WIDTH> <HEIGHT> [BANK 1-5]

Parameters:

<X> : Numerical value for the left offset of the picture.

<Y> : Numerical value for the top offset of the picture.

<WIDTH>: Numerical value for the width of the picture.

<HEIGHT>: Numerical value for the height of the picture.

[BANK 1-5] : The command operates on the specified bank number.

Result:

---

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
CROP 10 10 200 200 BANK 5
```

```
CROP -10 -10 512 512
```

## 1.10 Port ARexx : Scale

Nom de la commande: Scale

Syntax: Scale <WIDTH> <HEIGHT> [BANK 1-5]

Parameters:

<WIDTH>: Numerical value for the new width of the picture.

<HEIGHT>: Numerical value for the new height of the picture.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
Scale 80 80 BANK 2
```

```
Scale 100 100
```

## 1.11 Port ARexx : Negative

Nom de la commande: Negative

Syntax: Negative <RED> <GREEN> <BLUE> [BANK 1-5]

ou Negative <"RED"|"GREEN"|"BLUE"|"DEFAULT"|"ALL"> [BANK 1-5]

Parameters:

<RED>

<GREEN>

<BLUE> : Shall contain a boolean value (0 or 1) for each component. If the value for a component is 0, then it will not be modified.

<"RED"|"GREEN"|"BLUE"|"DEFAULT"|"ALL"> is a keyword which can be any of those specified. If the keyword is "RED", then only the red component will be inversed (or the green one with "GREEN", or the blue one with "BLUE", or all with "DEFAULT" or "ALL").

[BANK 1-5] : The command operates on the specified bank number.

---

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Negative 1 0 1

Negative ALL BANK 4

Negative RED

Negative 1 1 1 BANK 2

## 1.12 Port ARexx : Visual

Nom de la commande: Visual

Syntax: Visual <RES> <LACE> <XRATIO> <YRATIO> [BANK 1-5]

Parameters:

<RES> : Shall be one of the following keyword: "LORES", "HIRES", or "SUPERHIRES".

<LACE> : shall be one of the following keyword: "LACE" or "NOLACE".

<XRATIO> : numerical value containing the X ratio for the picture.

<YRATIO> : numerical value containing the Y ratio for the picture.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Visual SUPERHIRES LACE 22 44

Visual LORES NOLACE 22 22 BANK 2

## 1.13 Port ARexx : Emboss

Nom de la commande: Emboss

Syntax: Emboss [x] [y] [lumi] [thres] [BANK 1-5]

Parameters:

[x] : a numerical value which represents the horizontal shifting.

The default value is 1.

[y] : a numerical value which represents the vertical shifting.

The default value is 1.

---

[lumi] : is a numerical value, contained between 0 and 225. This represents the global brightness of the picture. The default is 150.

[Thres] : Threshold of the operator. Default is 1.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Emboss BANK 1

Emboss 2 -1 150

Emboss -1 -1 250 20 BANK 2

## 1.14 Port ARexx : Sketch

Nom de la commande: Sketch

Syntax: Sketch [PENSIZE] [THRES] [DEEP] [BANK 1-5]

Parameters:

[PENSIZE] : a numerical value which represents the pen width. The default is 2.

[THRES] : The threshold for the operator. The default is 255 (in the range 0-255).

[DEEP] : The base color pour the pen. The default is 255. (in the range 0-255).

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

Sketch

Sketch 3 150 200 BANK 1

Sketch 2 BANK 2

---

## 1.15 Port ARexx : Quake

Nom de la commande: Quake

Syntax: Quake <RADIUS> [DIRECTION] [BANK 1-5]

Parameters:

<RADIUS>: The Radius for the operation, in pixels.

[DIRECTION]: Is a keyword, either "HORIZONTAL" or "VERTICAL". The default is "HORIZONTAL".

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

Quake 5

Quake 10 VERTICAL BANK 1

Quake 50 HORIZONTAL

## 1.16 Port ARexx : Disperse Pixel

Nom de la commande: DispersePixel

Syntax: DispersePixel <RADIUS> <PROBA> [BANK 1-5]

Parameters:

<RADIUS>: The Radius for the operation, in pixels.

<PROBA>: The percentage of probability for the pixel displacement to occur. At 100%, all the pixels will be moved.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

DisplacePixel 5 100 BANK 1

DisplacePixel 10 50

## 1.17 Port ARexx : Convolve

Nom de la commande: Convolve

Syntax: Convolve <FILENAME> [THRES] [BANK 1-5]

Parameters:

<FILENAME>: The name of a convolution file. This filename shall contain a correct path to the file.

[THRES]: Threshold to the operator. Please refer to the operator documentation for further details.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

```
Convolve "Convolve/Blur5x5" 50
```

```
Convolve "Convolve/Triangle" BANK 5
```

```
Convolve "Convolve/Sharpening_1" 10 BANK 2
```

## 1.18 Port ARexx : Remove Isolated Pixels

Nom de la commande: RemIsolated

Syntax: RemIsolated <THRES> <CONTEXT> [BANK 1-5]

Parameters:

<THRES>: The threshold for the operator. Should be contained between 0 and 255.

<CONTEXT>: The context threshold. Should be between 0 and 255. See the operator documentation for further details.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
RemIsolated 200 50
```

```
RemIsolated 100 0 BANK 1
```

---

## 1.19 Port ARexx : Quantize

Nom de la commande: Quantize

Syntax: Quantize <COLOURS> [BANK 1-5]

Parameters:

[COLOURS]: The maximum number of colours that must be computed by the colour quantization. A new palette is created for the picture, containing this number of palette entries.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
Quantize 256 BANK 1
```

```
Quantize 16
```

## 1.20 Port ARexx : Load Palette

Nom de la commande: LoadPalette

Syntax: LoadPalette <FILENAME> [BANK 1-5]

Parameters:

<FILENAME>: The name for a palette file. The name shall contain a correct path for the file.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
LoadPalette "dh0:palettes/green.pal" BANK 1
```

```
LoadPalette "Palette/mypal" /* This is in GfxLab24's  
directory!! */
```

## 1.21 Port ARexx : Save Palette

Nom de la commande: SavePalette

Syntax: SavePalette <FILENAME> [BANK 1-5]

Parameters:

---

<FILENAME>: The name for a palette file. The name shall contain a correct path for the file.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
SavePalette "dh0:palettes/green.pal" BANK 1
```

```
SavePalette "Palette/mypal"
```

## 1.22 Port ARexx : Count Colors

Nom de la commande: CountColors

Syntax: CountColors <LIMITED> [BANK 1-5]

Parameters:

<LIMITED>: If the keywords "LIMITED" or "STOP" appear in the command line, the operator stops the counting when it reaches 4096 colours.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Return the number of colours counted in the picture, if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
CountColor LIMITED BANK 1
```

```
CountColor Bank 2
```

```
CountColor STOP
```

## 1.23 Port ARexx : GetWitdh

Nom de la commande: GetWidth

Syntax: GetWidth [BANK 1-5]

Parameters:

[BANK 1-5] : The command operates on the specified bank number.

Result:

If the bank exists, returns the picture width. Otherwise, returns an error message beginning with "ERROR"...

Example:

```
width = GetWidth BANK 1
```

```
width = GetWidth
```

---



## 1.24 Port ARexx : GetHeight

Nom de la commande: GetHeight

Syntax: GetHeight [BANK 1-5]

Parameters:

[BANK 1-5] : The command operates on the specified bank number.

Result:

If the bank exists, returns the picture height. Otherwise, returns an error message beginning with "ERROR"...

Example:

```
height = GetHeight BANK 1
```

```
height = GetHeight
```

## 1.25 Port ARexx : FastColorMap

Nom de la commande: FastColorMap

Syntax: FastColorMap <"ON" | "OFF">

Parameters:

<"ON" | "OFF">: Is either "ON" or "OFF" to set ON or OFF the fast colormap option.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
FastColorMap ON
```

```
FastColorMap OFF
```

## 1.26 Port ARexx : Request

Nom de la commande: Request

Syntax: Request <TEXT> <CHOICE>

This function pops up a requester, containing a text and one or more button gadgets. The result is returned to the script.

Parameters:

<TEXT>: Shall contain the text to display. The text must begin with <"> and end with another <"> (double-quote).

In an ARexx script, in order pass correctly the quotes, you must put all the text with double-quote between two

---

<'> (coma) characters. Thus, a text should be sent as:

'"This is a correct text"'. A return can be specified in the text by adding <\n> where needed.

ie: '"This the first line\nAnd this is the second."'

<CHOICE>: Shall contain the available choice(s) for the user (at least one!). The choices are contained between two <"> and by two <'>, just like explained above, and are separated by the <|> symbol.

ie: '"Choice 1 | Choice 2 | Choice 3"'.

Result:

The choice number is returned in the following sequence:

1,2,3,...,0. Thus, the first gadget (on the left) is always one, while the last gadget is always 0 (the standart choice for "Cancel").

Exemple:

Request '"Do you wish to continue ?"' '" Yes | No "'

Request '"Which format do you\nwish to use ?"' '"JPEG|TARGA"'

## 1.27 Port ARexx : PrintInfo

Nom de la commande: PrintInfo

Syntax: PrintInfo <TEXT>

This function prints a text in the GfxLab24's screen.

Parameters:

<TEXT>: Shall contain the text to display. The text must begin with <"> and end with another <"> (double-quote).

In an ARexx script, in order pass correctly the quotes, you must put all the text with double-quote between two

<'> (coma) characters. Thus, a text should be sent as:

'"This is a correct text"'. A return can be specified in the text by adding <\n> where needed.

ie: '"This the first line\nAnd this is the second."'

Result:

Always returns "OK".

Example:

PrintInfo '"Starting EmptyScript.rexx script"'

PrintInfo '"Spock ! What are you doing with that turtle??"'

PrintInfo '"Operation cancelled."'

## 1.28 Port ARexx : GetFromPattern

Nom de la commande: GetFromPattern

Syntax: GetFromPattern <PATTERN>

This function is used to make script writing easy. It takes any pattern, and return all the dos files matching with it.

Parameters:

<PATTERN> : Any AmigaDos pattern, or just a single filename. ie "~(#?.info)" will return every files in the path that is not an icon. #?.jpeg will return every file which extension is ".jpeg". Pic#?.iff will return all the pictures beginning with "Pic", end ending with ".iff" (ie: pic001.iff, pic002.iff, pic003.iff, ...).

Result:

Returns a string that contains all the AmigaDos file matching with the pattern. This string can easely be used by the WORD() ARexx command.

Example:

```
GetFromPattern #?.jpeg
```

```
GetFromPattern hello.iff
```

```
GetFromPattern dh0:pictures/#?
```

## 1.29 Port ARexx : BalanceFilter

Nom de la commande: BalanceFilter

Syntax: BalanceFilter <FILENAME> [RED\_OFF] [GREEN\_OFF]  
[BLUE\_OFF] [BANK 1-5]

Applies the Balance Filter operator on a picture.

Parameters:

<FILENAME>: The name of a BalanceFilter file. It shall contains a correct path to the file.

[RED\_OFF]

[GREEN\_OFF]

[BLUE\_OFF]: If one of this keyword is found, the specified component will not be affected by the operator.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
BalanceFilter Filter/AllGreen RED_OFF BANK 1
```

```
BalanceFilter dh0:myfilters/myfilter GREEN_OFF BLUE_OFF
```

### 1.30 Port ARexx : Rotate

Nom de la commande: Rotate

Syntax: Rotate <X> <Y> <ANGLE> <RADIUS | "FULLPICTURE">

[BANK 1-5]

Parameters:

<X> : X coordinate for the central point of the rotation.

<Y> : Y Coordinate for the central point of the rotation.

<ANGLE>: Radius of the rotation in degree.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Rotate 100 100 45 FULLPICTURE BANK 1

Rotate 200 200 80 50

Rotate 200 150 145 100 BANK 2

### 1.31 Port ARexx : Fade

Nom de la commande: Fade

Syntax: Fade <LEVEL> <FIRST bank> <LAST bank> <DEST bank>

Parameters:

<LEVEL>: A value that determinates the percentage of the last picture that should be seen on the first one. At 0, the resulting picture will be the first one. At 100, the resulting picture will be the last one.

<FIRST bank>: The keyword "FIRST" shall be followed by the number of the bank to use, or by one the keywords: "BLACK" for a black screen, "WHITE" for a white screen, or "CURRENT" for the current bank.

<LAST bank>: The keyword "LAST", followed by the number of the bank to use, or by one the keywords: "BLACK" for a black screen, "WHITE" for a white screen, or "CURRENT" for the current bank.

<DEST bank>: The keyword "DEST", followed by the number of the bank to use, or by the keyword "CURRENT" for the current bank.

---

**Result:**

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

**Example:**

```
Fade 50 FIRST 1 LAST 2 DEST CURRENT
Fade 80 FIRST CURRENT LAST BLACK DEST CURRENT
Fade 10 First 5 LAST 4 DEST 3
```

## 1.32 Port ARexx : CopyBrush

Nom de la commande: CopyBrush

Syntax: CopyBrush <DEST> <SOURCE> <X> <Y> [WIDTH] [HEIGHT]  
[FADE] ["TRANSPARENTCOLOR"]

**Parameters:**

<DEST>: The number of the destination bank, or the keyword "CURRENT" for the current bank.

<SOURCE>: The number of the source bank, or the keyword "CURRENT" for the current bank.

<X> : Numerical value for the left offset of the position where the brush will be copied.

<Y> : Numerical value for the top offset of the position where the brush will be copied.

[WIDTH]: Width of the area where the brush will be copied.

The default is the width of the picture/brush.

[HEIGHT]: Height of the area where the brush will be copied.

The default is the height of the picture/brush.

<LEVEL>: A value that determinates the percentage of the original picture that should be seen beyond the brush. At 0, the brush is not copied, while at 100 it is fully copied with no fade effect.

["TRANSPARENTCOLOR"] if specified, this keyword indicates that the transparent color of the brush will not be copied.

**Result:**

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

**Example:**

```
CopyBrush CURRENT 4 50 50
CopyBrush 4 CURRENT 0 0 200 200 75 TRANSPARENTCOLOR
```

### 1.33 Port ARexx : CreatePicture

Nom de la commande: CreatePicture

Syntax: CreatePicture <WIDTH> <HEIGHT> [RED] [GREEN]  
[BLUE] [BANK 1-5]

This function creates a new picture.

Parameters:

<WIDTH>: width of the picture to create.

<HEIGHT>: height of the picture to create.

[RED]: The red component value for the background color (between 0 and 255).

[GREEN]: The green component value for the background color (between 0 and 255).

[BLUE]: The blue component value for the background color (between 0 and 255).

[BANK 1-5] : The command operates on the specified bank number.

suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Exemple:

```
CreatePicture 512 512
```

```
CreatePicture 256 256 255 0 0 BANK 2
```

```
CreatePicture 500 500 BANK 1
```

### 1.34 Port ARexx : AddText

Nom de la commande: AddText

Syntax: AddText <TEXT> <X> <Y> [BANK 1-5]

This function put some text on a picture.

Parameters:

<TEXT>: Shall contain the text to display. The text must begin with <"> and end with another <"> (double-quote).

In an ARexx script, in order pass correctly the quotes, you must put all the text with double-quote between two

<'> (coma) characters. Thus, a text should be sent as:

'"This is a correct text"'. Line feeds are not allowed

with this operator.

<X> : Numerical value for the left offset of the position where the text will be copied.

<Y> : Numerical value for the top offset of the position where the text will be copied.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
AddText "Hello World !" 100 100
```

```
AddText "How do you do ?" 100 100 BANK 2
```

### 1.35 Port ARexx : TextColor

Nom de la commande: TextColor

Syntax: TextColor <RED> <GREEN> <BLUE>

This function defines the color of the text for the TEXT operator.

Parameters:

[RED]: The red component value for the text color (between 0 and 255).

[GREEN]: The green component value for the text color (between 0 and 255).

[BLUE]: The blue component value for the text color (between 0 and 255).

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

```
TextColor 0 0 0
```

```
TextColor 255 255 0
```

```
TextColor 120 120 250
```

---

### 1.36 Port ARexx : TextFont

Nom de la commande: TextFont

Syntax: TextFont <FONT>

This function defines the name of the font to use with the Text operator.

Parameters:

<FONT> : name of the font.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

TextFont topaz.font

TextFont mybeautifulfont.font

### 1.37 Port ARexx : TextSize

Nom de la commande: TextSize

Syntax: TextSize <Size>

This function defines the size of the font to use with the Text operator.

Parameters:

<SIZE> : Size of the font.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

TextSize 30

TextSize 8

TextSize 90

### 1.38 Port ARexx : ClearBank

Nom de la commande: ClearBank

Syntax: ClearBank [BANK 1-5]

This function clears a bank. It deletes the picture from memory (not from disk, of course) by freeing all the resources allocated.

---



Parameters:

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

ClearBank

ClearBank BANK 2

### 1.39 Port ARexx : BitmapToRaw

Nom de la commande: BitmapToRaw

Syntax: BitmapToRaw [BANK 1-5]

This function creates a 24-bits buffer from a bitmap picture.

Parameters:

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

BitmapToRaw Bank 2

BitmapToRaw

BitmapToRaw Bank 5

### 1.40 Port ARexx : TransparentColor

Nom de la commande: TransparentColor

Syntax: TransparentColor <BITMAP color> [BANK 1-5]

ou TransparentColor <24BITS red green blue> [TOL] [BANK 1-5]

Indicates which is the transparent color for a bitmap or a 24-bits buffer.

Parameters:

<BITMAP color> : If the keyword "BITMAP" appears, it should be followed by a value that indicates which is the transparent color in the palette entry.

<24BITS red green blue>: If the keyword "24BITS" appears, it should be followed by the values of the red, green, and

---

blue components. Those values determinates the transparent color of that picture.

[TOL]: This argument is only available for 24-bits pictures, and indicates the percentage of tolerance for the transparent color.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

TransparentColor BITMAP 5

TransparentColor BITMAP 0 BANK 3

TransparentColor 24BITS 255 0 255 10 BANK 4

TransparentColor 24BITS 255 255 255 BANK 2

## 1.41 Port ARexx : DisplayMode

Nom de la commande: DisplayMode

Syntax: DisplayMode [MODE] ...

This functions allows to select the display mode of the 24-bits pictures.

Parameters:

[MODE] : Can be any of the following keyword:

HAM : to set HAM mode on.

NOHAM : to set HAM mode off.

LOWRES : to select the low resolution display.

HIRES : to select the high resolution.

SUPERHIRES: to select the superhigh resolution.

LACE : to select the interlaced mode.

NOLACE : to select the non-interlaced mode.

PAL : to select a PAL moniteur display

NTSC : for a NTSC monitor.

DEPTH x : to select the number of bitplanes for the display.

FASTCOLORMAP: to set the fastcolormap mode on.

NOFASTCOLORMAP: to set the fastcolormap mode off.

DITHERING <name>: To select the dithering mode, where <name> can be either "HALFTONE", "ERROR", or "NONE".

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

DisplayMode LORES LACE HAM

DisplayMode PAL SUPERHIRES DEPTH 8 HAM LACE

## 1.42 Port ARexx : Whirl

Nom de la commande: Whirl

Syntax: Whirl <X> <Y> <LEVEL> [BANK 1-5]

Parameters:

<X> : X coordinate for the central point of the rotation.

<Y> : Y Coordinate for the central point of the rotation.

<LEVEL>: Rotation level for the whirl effect.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Whirl 50 50 200

Whirl 200 200 10

Whirl 200 200 800

## 1.43 Port ARexx : Noise

Nom de la commande: Noise

Syntax: Noise <WHITE x> [BANK 1-5]

ou Noise [RED x] [GREEN x] [BLUE x] [BANK 1-5]

Parameters:

<WHITE x> : indicates the percentage of white noise for the picture. x must be  $0 \leq x \leq 100$ .

[RED x] : indicates the percentage of red noise for the picture. x must be  $0 \leq x \leq 100$ .

[GREEN x] : indicates the percentage of green noise for the picture. x must be  $0 \leq x \leq 100$ .

[BLUE x] : indicates the percentage of blue noise for the

picture. x must be  $0 \leq x \leq 100$ .

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

Noise WHITE 50 BANK 2

Noise WHITE 10

Noise RED 20 GREEN 30 BANK 2

Noise BLUE 10 RED 20

Noise GREEN 10

## 1.44 Port ARexx : IsBitmap

Nom de la commande: IsBitmap

Syntax: IsBitmap [BANK 1-5]

Parameters:

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if the bank has a bitmap buffer, or "ERROR" otherwise.

Example:

IsBitmap BANK 2

IsBitmap

## 1.45 Port ARexx : Is24Bits

Nom de la commande: Is24Bits

Syntax: Is24Bits [BANK 1-5]

Parameters:

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if the bank has a 24-bits buffer, or "ERROR" otherwise.

Example:

Is24Bits BANK 5

Is24Bits

---

## 1.46 Port ARexx : Mosaic

Nom de la commande: Mosaic

Syntax: Whirl <Size> [CRACKLING] [BANK 1-5]

Parameters:

<Size> : Size of a piece of mosaic, in pixels.

[CRACKLING] : If this keyword appears, the operator will apply some crackling effect on the mosaic.

[BANK 1-5] : The command operates on the specified bank number.

Result:

Returns "OK" if no error occurred, otherwise returns a message beginning with "ERROR", and which explains what happened.

Example:

MOSAIC 30 CRACKLING

MOSAIC 5

MOSAIC 10 CRACKLING