

**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	ARexx port : LOAD	2
1.3	ARexx port : Save	2
1.4	ARexx port : Colour To Gray	3
1.5	ARexx port : Brightness	3
1.6	ARexx port : Filter	4
1.7	ARexx port : Bicolor	4
1.8	ARexx port : Flip	5
1.9	ARexx port : Crop	6
1.10	ARexx port : Scale	6
1.11	ARexx port : Negative	7
1.12	ARexx port : Visual	7
1.13	ARexx port : Emboss	8
1.14	ARexx port : Sketch	9
1.15	ARexx port : Quake	9
1.16	ARexx port : Disperse Pixel	10
1.17	ARexx port : Convolve	10
1.18	ARexx port : Remove Isolated Pixels	11
1.19	ARexx port : Quantize	11
1.20	ARexx port : Load Palette	12
1.21	ARexx port : Save Palette	12
1.22	ARexx port : Count Colors	12
1.23	ARexx port : GetWidth	13
1.24	ARexx port : GetHeight	13
1.25	ARexx port : FastColorMap	14
1.26	ARexx port : Request	14
1.27	ARexx port : Printinfo	15
1.28	ARexx port : GetFromPattern	16
1.29	ARexx port : BalanceFilter	16

---

1.30 ARexx port : Rotate . . . . .	17
1.31 ARexx port : Fade . . . . .	18
1.32 ARexx port : CopyBrush . . . . .	18
1.33 ARexx port : CreatePicture . . . . .	19
1.34 ARexx port : AddText . . . . .	20
1.35 ARexx port : TextColor . . . . .	21
1.36 ARexx port : TextFont . . . . .	21
1.37 ARexx port : TextSize . . . . .	22
1.38 ARexx port : ClearBank . . . . .	22
1.39 ARexx port : BitmapToRaw . . . . .	22
1.40 ARexx port : TransparentColor . . . . .	23
1.41 ARexx port : DisplayMode . . . . .	24
1.42 ARexx port : Whirl . . . . .	24
1.43 ARexx port : Noise . . . . .	25
1.44 ARexx port : IsBitmap . . . . .	26
1.45 ARexx port : Is24Bits . . . . .	26
1.46 ARexx port : Mosaic . . . . .	27

---

# Chapter 1

## 01261410-0

### 1.1 GfxLab : ARexx port

AREXX PORT

=====

Le port ARexx de GfxLab24 permet d'automatiser facilement des opérations répétitives sur un fichier ou une série de fichier.

Le programme ARexx doit se lancer sur un shell externe de la manière suivante :

[Ouvrez un shell ou un cli]

[tapez la séquence suivante :]

rx [NOM DU SCRIPT] [paramètres éventuels]

Voici la liste des commandes ARexx comprises par GfxLab24. Elles ne vous intéressent que si vous êtes en mesure d'écrire des scripts

ARexx, dans le cas contraire vous pouvez utiliser les scripts

fournis avec GfxLab24, dans le tiroir REXX\_Scripts.

Les paramètres entre < > sont obligatoires, tandis que ceux spécifiés entre crochets [ ] sont optionnels.

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 ARexx port : LOAD

Nom de la commande: Load

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

Paramètres:

NOM : Nom du fichier à charger.

Tous les formats supportés par GfxLab24 sont acceptés.

BANK [1-5] : La commande agit sur la banque dont le numéro suit le mot-clef.

Retour:

Renvoie 'OK' si le chargement s'est correctement effectué, ou un message d'erreur commençant par "ERROR" et suivi de l'explication de l'erreur autrement.

## 1.3 ARexx port : Save

Nom de la commande: Save

Syntaxe: Save <NOM> [FORMAT]

Paramètres:

NOM: Ce paramètre indique le nom avec lequel l'image présente dans la mémoire principale de GfxLab24 doit être sauvegardée.

Le nom peut contenir un chemin d'accès. S'il n'en contient pas, le répertoire courant est utilisé.

FORMAT: Ce paramètre optionnel permet d'effectuer la sauvegarde dans un format spécifié.

FORMAT peut prendre les valeurs JPEG, IFF24, IFF, TARGA, ou PCX.

Par défaut, si le format n'est pas spécifié, il se fera en IFF24

Retour:

Retourne "OK" si la sauvegarde s'est correctement effectuée ou renvoie sinon un message d'erreur.

Exemple d'utilisation:

Save 'df0:MonFichier' JPEG

.

## 1.4 ARexx port : Colour To Gray

Nom de la commande: ColorToGray

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

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

Paramètres:

<RED> : Pondération de la composante rouge, lors de la transformation d'une couleur en dégradé de gris. Le chiffre donné doit être compris en 0 et 10000. Le chiffre est 2989 par défaut.

<GREEN>: Pondération de la composante verte.

<BLUE> : Pondération de la composante bleue.

<PAL | HDTV> : si l'un des mots-clefs "PAL" ou "HDTV" est utilisé, les pondérations correspondant à ces formats sont utilisées.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout s'est correctement passé, "ERROR" sinon.

Exemple:

```
ColorToGray 1200 5600 7899
```

```
ColorToGray PAL BANK 3
```

## 1.5 ARexx port : Brightness

Nom de la commande: Brightness

Syntaxe: Brightness <VALEUR>

Paramètres:

VALEUR: Peut être une valeur numérique entre 1 et 400, ou alors un mot clef: HALF qui est équivalent à 50, et DOUBLE qui est équivalent à 200.

Retour:

Retourne la chaîne 'OK' si tout s'est correctement passé, et un message d'erreur dans le cas contraire.

Exemple:

```
Brightness 75
```

```
Brightness HALF
```

## 1.6 ARexx port : Filter

Nom de la commande: Filter

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

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

Paramètres:

<RED> : Pourcentage de la composante rouge de la couleur qui doit être filtré.

<GREEN> : Pourcentage de la composante verte de la couleur qui doit être filtré.

<BLUE> : Pourcentage de la composante bleue de la couleur qui doit être filtré.

<"RED" | "GREEN" | "BLUE"> : Si l'un des mots-clefs "RED", "GREEN" ou "BLUE" est utilisé, les pondérations utilisées correspondent aux filtres rouge, vert, et bleus prédéfinis.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou une chaîne débutant par "ERROR", et contenant l'explication de l'erreur sinon.

Exemple:

Filter RED BANK 2

Filter 110 80 60 BANK 1

Filter GREEN

Filter 100 50 100

## 1.7 ARexx port : Bicolor

Nom de la commande: BiColor

Syntaxe: Bicolor [METHOD] [THRESHOLD]

Paramètres:

[METHOD], est un mot clef qui peut être l'un de ceux-ci :

"NONE" (aucun tramage), valeur par défaut.

"FLOYD" (Floyd-Steinberg)

"BURKES"

"STUCKI"

"SIERRA"



"JARVIS"

"STEVENSON"

"EDGES"

"BOUNDARY"

[THRESHOLD] est le chiffre indiquant le seuil entre les deux couleurs.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou une chaîne débutant par "ERROR", et contenant l'explication de l'erreur sinon.

Exemple:

BiColor

BiColor BOUNDARY BANK 1

## 1.8 ARexx port : Flip

Nom de la commande: Flip

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

Paramètres:

[DIRECTION] : Peut être l'un des deux mots clefs "HORIZONTAL" ou "VERTICAL", par défaut (si aucun n'est indiqué), l'opérateur appliquera l'effet HORIZONTAL.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou une chaîne débutant par "ERROR", et contenant l'explication de l'erreur sinon.

Exemple:

Flip VERTICAL BANK 3

Flip

Flip Bank 2

Flip HORIZONTAL

## 1.9 ARexx port : Crop

Nom de la commande: Crop

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

Paramètres:

<X> : Une valeur numérique indiquant l'abscisse du coin supérieur gauche de l'image.

<Y> : Une valeur numérique indiquant l'ordonnée du coin supérieur gauche de l'image.

<WIDTH> : Une valeur indiquant la nouvelle largeur en pixels de l'image.

<HEIGHT> : Une valeur indiquant la nouvelle hauteur en pixels de l'image.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou une chaîne débutant par "ERROR", et contenant l'explication de l'erreur sinon.

Exemple:

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

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

## 1.10 ARexx port : Scale

Nom de la commande: Scale

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

Paramètres:

<WIDTH> : Une valeur indiquant la nouvelle largeur en pixels de l'image.

<HEIGHT> : Une valeur indiquant la nouvelle hauteur en pixels de l'image.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou une chaîne débutant par "ERROR", et contenant l'explication de l'erreur sinon.

Exemple:

```
Scale 80 80 BANK 2
```

```
Scale 100 100
```

## 1.11 ARexx port : Negative

Nom de la commande: Negative

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

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

Paramètres:

<RED>

<GREEN>

<BLUE> : Doivent contenir des valeur booleenne 0 ou 1. Si la valeur est à 0, la composante ne sera pas modifiée, alors qu'elle le sera si sa valeur est à 1.

<"RED"|"GREEN"|"BLUE"|"DEFAULT"|"ALL"> sont des mots-clefs qui peuvent être n'importe lesquels de la série ci-dessus. Si le mot clef est "RED", alors seule la composante rouge sera inversée, ce sera la composante verte pour "GREEN", bleue pour "BLUE", et les trois ensemble pour "DEFAULT" ou "ALL".

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaine débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

Negative 1 0 1

Negative ALL BANK 4

Negative RED

Negative 1 1 1 BANK 2

## 1.12 ARexx port : Visual

Nom de la commande: Visual

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

Paramètres:

<RES> : Doit contenir l'un des trois mots clefs suivant : "LORES", "HIRES", ou "SUPERHIRES".

<LACE> : Doit contenir l'un des deux mots clefs "LACE" ou "NOLACE".

<XRATIO> : Contient une valeur numérique indiquant le ratio en X de l'image.

---

<YRATIO> : Contient une valeur numérique indiquant le ratio en Y de l'image.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

Visual SUPERHIRES LACE 22 44

Visual LORES NOLACE 22 22 BANK 2

### 1.13 ARexx port : Emboss

Nom de la commande: Emboss

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

Paramètres:

[x] : contient le décalage en pixels qui va donner le relief horizontal. La valeur par défaut est 1.

[y] : contient le décalage en pixels qui va donner le relief vertical. La valeur par défaut est 1.

[lumi] : Contient une valeur numérique comprise entre 0 et 255 qui donnera la luminosité globale de l'image. Par défaut à 150.

[Thres] : La seuil de sensibilité de l'opérateur. Par défaut à 0.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

Emboss BANK 1

Emboss 2 -1 150

Emboss -1 -1 250 20 BANK 2

## 1.14 ARexx port : Sketch

Nom de la commande: Sketch

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

Paramètres:

[PENSIZE] : Le décalage en pixel qui simulera la largeur du crayon. Par défaut à 2.

[THRES] : La seuil de sensibilité de l'opérateur. Par défaut à 1.

Peut être compris entre 0 et 255.

[DEEP] : La couleur de base de l'opérateur pour le crayon. Par défaut à 1. Peut être compris entre 0 et 255.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

Sketch

Sketch 3 150 200 BANK 1

Sketch 2 BANK 2

## 1.15 ARexx port : Quake

Nom de la commande: Quake

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

Paramètres:

<RADIUS> : Le rayon d'action de l'opération, en pixels.

[DIRECTION] : Peut être l'un des mots clefs "HORIZONTAL" ou "VERTICAL", par défaut à HORIZONTAL.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

Quake 5

Quake 10 VERTICAL BANK 1

Quake 50 HORIZONTAL

## 1.16 ARexx port : Disperse Pixel

Nom de la commande: DispersePixel

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

Paramètres:

<RADIUS> : Le rayon d'action de l'opération, en pixels.

<PROBA> : Le pourcentage de probabilité pour qu'un pixel soit déplacé. A 100, tous les pixels seront déplacés.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
DisplacePixel 5 100 BANK 1
```

```
DisplacePixel 10 50
```

## 1.17 ARexx port : Convolve

Nom de la commande: Convolve

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

Paramètres:

<FILENAME> : Un nom de fichier de type Convolve. Il doit contenir le chemin correct à partir du répertoire de GfxLab24.

[THRES] : La seuil de sensibilité de l'opérateur. Par défaut à 0. Peut être compris entre 0 et 255.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

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

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

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

## 1.18 ARexx port : Remove Isolated Pixels

Nom de la commande: RemIsolated

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

Paramètres:

<THRES> : La seuil de sensibilité de l'opérateur.

Peut être compris entre 0 et 255.

<CONTEXT> : La valeur numérique du contexte. Peut être compris entre 0 et 255. Sa valeur normale est de 50.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
RemIsolated 200 50
```

```
RemIsolated 100 0 BANK 1
```

## 1.19 ARexx port : Quantize

Nom de la commande: Quantize

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

Paramètres:

[COLOURS] : Le nombre maximum de couleurs que doit créer la méthode de quantisation.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
Quantize 256 BANK 1
```

```
Quantize 16
```

## 1.20 ARexx port : Load Palette

Nom de la commande: LoadPalette

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

Paramètres:

<FILENAME> : Un nom de fichier de type Palette. Il doit contenir le chemin correct à partir du répertoire de GfxLab24.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

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

```
LoadPalette "Palette/mypal"
```

## 1.21 ARexx port : Save Palette

Nom de la commande: SavePalette

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

Paramètres:

<FILENAME> : Un nom de fichier de type Palette. Il doit contenir le chemin correct à partir du répertoire de GfxLab24.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

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

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

## 1.22 ARexx port : Count Colors

Nom de la commande: CountColors

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

Paramètres:

---



<LIMITED> : Si l'un des mots clefs "LIMITED" ou "STOP" apparait dans la ligne de commande, l'opérateur s'arrêtera s'il atteint ou dépasse 4096 couleurs.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

CountColor LIMITED BANK 1

CountColor Bank 2

CountColor STOP

## 1.23 ARexx port : GetWidth

Nom de la commande: GetWidth

Syntaxe: GetWidth [BANK 1-5]

Paramètres:

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

Si la banque existe, renvoie la largeur en pixel de l'image qu'elle contient, ou sinon renvoie un message d'erreur débutant par "ERROR".

Exemple:

width = GetWidth BANK 1

width = GetWidth

## 1.24 ARexx port : GetHeight

Nom de la commande: GetHeight

Syntaxe: GetHeight [BANK 1-5]

Paramètres:

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

---

Si la banque existe, renvoie la hauteur en pixel de l'image qu'elle contient, ou sinon renvoie un message d'erreur débutant par "ERROR".

Exemple:

```
height = GetHeight BANK 1
```

```
height = GetHeight
```

## 1.25 ARexx port : FastColorMap

Nom de la commande: FastColorMap

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

Paramètres:

<"ON" | "OFF"> : L'un des mots clefs "ON" ou "OFF" doit apparaître, indiquant ainsi au programme qu'il doit respectivement utiliser ou ne pas utiliser la palette prédéfinie.

Retour:

Si le mot clef est précisé, renvoie "OK", ou sinon renvoie un message d'erreur débutant par "ERROR".

Exemple:

```
FastColorMap ON
```

```
FastColorMap OFF
```

## 1.26 ARexx port : Request

Nom de la commande: Request

Syntaxe: Request <TEXT> <CHOICE>

Definition : Cette fonction doit permettre d'afficher une boîte de requête, contenant un texte, ainsi qu'un ou plusieurs bouton, dont le résultat est renvoyé au script.

Paramètres:

<TEXT> : Doit contenir le texte à afficher. Le texte doit commencer par un <"> et se terminer par un autre <"> (double-quote). Dans un script ARexx, pour que les quotes soient correctement envoyés au programme, il faudra obligatoirement les mettre entre <'>. Ainsi un texte devra être envoyé de la manière suivante : "'Ceci est un texte"'. Un retour à la ligne sera effectué en ajoutant <\n> à l'endroit voulu. Ainsi : "'Ceci est la

première ligne\nEt ceci la seconde".

<CHOICE> : doit contenir les choix disponible pour l'utilisateur, au minimum un. Les choix sont contenus dans un même texte (soumis aux mêmes condition que précédemment), et sont séparés par le signe <|>.

Par exemple : "Choix 1 | Choix 2 | Cancel".

Retour:

Le numéro du choix renvoyé, dans l'ordre suivant : 1, 2, 3, ..., 0.

Si un seul choix est présent, 0 est renvoyé. Si plusieurs choix sont présents, le numéro le plus à gauche démarre à 1, vont en s'incrémentant en allant vers la droite, à l'exception du dernier choix, qui renvoie toujours 0 (car c'est la place du bouton CANCEL).

Exemple:

Request "Voulez-vous continuer ?" " Oui | Non "

Request "Sauvegarder sous \n quel format?" "JPEGITARGA"

## 1.27 ARexx port : PrintInfo

Nom de la commande: PrintInfo

Syntaxe: PrintInfo <TEXT>

Definition : Cette fonction doit permettre d'imprimer un texte à l'écran dans la fenêtre d'information principale.

Paramètres:

<TEXT> : Doit contenir le texte à afficher. Le texte doit commencer par un <"> et se terminer par un autre <"> (double-quote). Dans un script ARexx, pour que les quotes soient correctement envoyés au programme, il faudra obligatoirement les mettre entre <'>. Ainsi un texte devra être envoyé de la manière suivante : "Ceci est un texte". Les retours à la ligne ne sont pas autorisés dans cette fonction.

Retour:

Renvoie toujours "OK".

Exemple:

PrintInfo "Debut du script EmptyScript.rexx"

PrintInfo "Je commence les opérations..."

PrintInfo "Terminé !"

## 1.28 ARexx port : GetFromPattern

Nom de la commande: GetFromPattern

Syntaxe: GetFromPattern <PATTERN>

Definition : Cette fonction est destinée à faciliter l'écriture de scripts, et renvoie la totalité des fichiers correspondant à un motif donné en paramètre.

Paramètres:

<PATTERN> : Un motif de fichier, ou tout simplement un nom de fichier. Les motifs (patterns) doivent être conformes aux spécifications de l'AmigaDos, et peuvent être de n'importe quelle longueur. Par exemple ~(#?.info) renverra tous les fichiers qui ne sont pas des icônes, #?.jpeg renverra tous les fichiers jpeg, ou encore Image#?.iff renverra toutes les images commençant par Image et finissant par ".iff" (image001.iff, image002.iff, image003.iff, etc.)

Retour:

Renvoie une chaîne de caractères contenant tous les fichiers AmigaDos qui correspondent au motif donné. Cette chaîne est principalement destinée à être traitée par la commande WORD() de ARexx. Par exemple #?.jpeg pourra renvoyer une chaîne contenant : "Image1.jpeg tutu.jpeg image2.jpeg image5.jpeg image3.jpeg"

Exemple:

GetFromPattern #?.jpeg

GetFromPattern hello.iff

GetFromPattern dh0:pictures/#?

## 1.29 ARexx port : BalanceFilter

Nom de la commande: BalanceFilter

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

Paramètres:

<FILENAME> : Un nom de fichier de type BalanceFilter. Il doit contenir le chemin correct à partir du répertoire de GfxLab24.

[RED\_OFF]

[GREEN\_OFF]

[BLUE\_OFF] : Si l'un de ces mots clefs est présent, la composante

à laquelle ils correspondent ne sera pas affectée  
(respectivement Rouge, verte, et bleue).

[BANK 1-5] : La commande agit sur la banque dont le numéro  
suit le mot-clef. Si cette précision est omise, la  
commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant  
par "ERROR", et contenant l'explication de l'erreur.

Exemple:

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

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

### 1.30 ARexx port : Rotate

Nom de la commande: Rotate

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

[BANK 1-5]

Paramètres:

<X> : Valeur numérique indiquant l'abscisse du centre de la  
rotation.

<Y> : Valeur numérique indiquant l'ordonnée du centre de la  
rotation.

<ANGLE> : Valeur numérique exprimant l'angle en degré de la  
rotation.

<RADIUS | "FULLPICTURE"> : Soit une valeur numérique exprimant le  
rayon d'action de l'opération, soit le mot clef "FULLPICTURE" pour  
que l'opération s'effectue sur la totalité de l'image.

[BANK 1-5] : La commande agit sur la banque dont le numéro  
suit le mot-clef. Si cette précision est omise, la  
commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant  
par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
Rotate 100 100 45 FULLPICTURE BANK 1
```

```
Rotate 200 200 80 50
```

```
Rotate 200 150 145 100 BANK 2
```

### 1.31 ARexx port : Fade

Nom de la commande: Fade

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

Paramètres:

<LEVEL> : Une valeur numérique exprimant le pourcentage de l'image de fin par rapport à celle du début. A 0, on ne verra que l'image de départ, et à 100 uniquement celle de fin.

<FIRST bank> : Le mot clef "FIRST" suivi soit du numéro de la banque à utiliser, soit des mots clefs suivants :

"CURRENT" pour utiliser l'image en cours, "BLACK" pour ne pas indiquer de banque mais une image noire, et "WHITE" pour une image blanche.

<LAST bank> : Le mot clef "LAST" suivi soit du numéro de la banque à utiliser, soit des mots clefs suivants :

"CURRENT" pour utiliser l'image en cours, "BLACK" pour ne pas indiquer de banque mais une image noire, et "WHITE" pour une image blanche.

<DEST bank> : Le mot clef "FIRST" suivi soit du numéro de la banque à utiliser, soit des mots clefs "CURRENT" pour utiliser l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
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 ARexx port : CopyBrush

Nom de la commande: CopyBrush

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

Paramètres:

<DEST> : Numéro de la banque à utiliser comme banque de destination (sur laquelle sera copiée la banque source), ou le mot clef "CURRENT" pour indiquer la banque principale.

<SOURCE> : Numéro de la banque à copier sur la banque destination.

---

Ou le mot clef "CURRENT" pour indiquer la banque principale.

<X> : Une valeur numérique indiquant l'abscisse en pixel du coin supérieur gauche de la zone ou s'effectuera la copie.

<Y> : Une valeur numérique indiquant l'ordonnée en pixel du coin supérieur gauche de la zone ou s'effectuera la copie.

[WIDTH] : La largeur de la zone de copie. Par défaut la largeur est celle l'image source. Si la largeur est différente, l'image ne sera pas tronquée mais mise à l'échelle.

[HEIGHT] : La hauteur de la zone de copie. Par défaut, il s'agit de la hauteur de l'image source. Si la hauteur est différente de l'image source, l'image ne sera pas tronquée mais mise à l'échelle pour rentrer dans le zone voulue.

<FADE> : Une valeur numérique exprimant le pourcentage de l'image source par rapport à l'image de destination. A 0, on ne verra que l'image de départ, et à 100 uniquement celle de fin.

["TRANSPARENTCOLOR"] si ce mot clef est indiqué, la copie s'effectuera en respectant la couleur transparente de l'image source.

Retour:

"OK" si tout se passe correctement, ou sinon une chaine débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
CopyBrush CURRENT 4 50 50
```

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

### 1.33 ARexx port : CreatePicture

Nom de la commande: CreatePicture

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

Cette fonction crée une nouvelle image de taille WIDTHxHEIGHT.

Paramètres:

<WIDTH> : Largeur en pixel de l'image à créer.

<HEIGHT> : Hauteur en pixel de l'image à créer.

[RED] : Valeur de la composante rouge pour la couleur de fond de l'image (entre 0 et 255).

---

[GREEN] : Valeur de la composante verte pour la couleur de fond de l'image (entre 0 et 255).

[BLUE] : Valeur de la composante bleue pour la couleur de fond de l'image (entre 0 et 255).

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
CreatePicture 512 512
```

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

```
CreatePicture 500 500 BANK 1
```

### 1.34 ARexx port : AddText

Nom de la commande: AddText

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

Cette fonction ajoute du texte sur une image.

Paramètres:

<TEXT> : Doit contenir le texte à afficher. Le texte doit commencer par un "<" et se terminer par un autre ">" (double-quote). Dans un script ARexx, pour que les quotes soient correctement envoyés au programme, il faudra obligatoirement les mettre entre '<'. Ainsi un texte devra être envoyé de la manière suivante :

'"Ceci est un texte"'. Les retours à la ligne ne sont pas autorisés dans cette fonction.

<X> : Valeur numérique exprimant l'abscisse en pixel du début du texte.

<Y> : Valeur numérique exprimant l'ordonnée en pixel du début du texte.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

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

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



### 1.35 ARexx port : TextColor

Nom de la commande: TextColor

Syntaxe: TextColor <RED> <GREEN> <BLUE>

Cette fonction définit la couleur pour la fonction d'ajout de texte.

Paramètres:

<RED> : Valeur numérique donnant la composante Rouge de la couleur qui doit être utilisée (entre 0 et 255).

<GREEN> : Valeur numérique donnant la composante Verte de la couleur qui doit être utilisée (entre 0 et 255).

<BLUE> : Valeur numérique donnant la composante Bleue de la couleur qui doit être utilisée (entre 0 et 255).

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
TextColor 0 0 0
```

```
TextColor 255 255 0
```

```
TextColor 120 120 250
```

### 1.36 ARexx port : TextFont

Nom de la commande: TextFont

Syntaxe: TextFont <FONT>

Cette fonction définit la police de caractères pour la fonction d'ajout de texte.

Paramètres:

<FONT> : Nom de la fonte à utiliser.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
TextFont topaz.font
```

```
TextFont mybeautifulfont.font
```

### 1.37 ARexx port : TextSize

Nom de la commande: TextSize

Syntaxe: TextSize <Size>

Cette fonction définit la taille de la police de caractères pour la fonction d'ajout de texte.

Paramètres:

<SIZE> : Valeur numérique exprimant la taille à utiliser.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

TextSize 30

TextSize 8

TextSize 90

### 1.38 ARexx port : ClearBank

Nom de la commande: ClearBank

Syntaxe: ClearBank [BANK 1-5]

Cette fonction efface le contenu d'une banque.

Paramètres:

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

ClearBank

ClearBank BANK 2

### 1.39 ARexx port : BitmapToRaw

Nom de la commande: BitmapToRaw

Syntaxe: BitmapToRaw [BANK 1-5]

Cette fonction crée dans une banque un buffer 24 bits correspondant au bitmap de cette banque.

Paramètres:

---

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

BitmapToRaw Bank 2

BitmapToRaw

BitmapToRaw Bank 5

## 1.40 ARexx port : TransparentColor

Nom de la commande: TransparentColor

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

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

Cette fonction indique la couleur transparente pour un bitmap ou une image 24 bits.

Paramètres:

<BITMAP color> : Si le mot clef "BITMAP" est suivi d'une valeur numérique comprise entre 0 et 255, cette valeur est positionné en tant que couleur transparente pour le bitmap de la banque.

<24BITS red green blue> : si le mot clef "24BITS" est suivi de trois valeurs numériques comprises entre 0 et 255, ces valeurs seront utilisées pour désigner la couleur transparente de l'image 24bits.

[TOL] : Uniquement valable pour les images 24bits, ce paramètre précise le pourcentage de tolérance pour la couleur transparente.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

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 ARexx port : DisplayMode

Nom de la commande: DisplayMode

Syntaxe: DisplayMode [MODE] ...

Cette fonction permet de sélectionner le mode d'affichage des images pour le mode Preview.

Paramètres:

[MODE] : Peut être n'importe lequel des mots clefs suivant :

HAM : pour passer en mode HAM.

NOHAM : pour enlever le mode HAM.

LOWRES : pour passer en mode basse résolution.

HIRES : pour passer en mode haute résolution.

SUPERHIRES: pour le mode super-haute résolution.

LACE : pour passer en mode entrelacé.

NOLACE : pour passer en mode non-entrelacé.

PAL : pour sélectionner un moniteur PAL.

NTSC : pour un moniteur NTSC.

DEPTH x : pour choisir la profondeur d'écran x (en bitplans).

FASTCOLORMAP: Pour choisir le mode FastColorMap.

NOFASTCOLORMAP: Pour enlever le mode FastColorMap.

DITHERING <name>: Pour choisir le mode de dithering,

où <name> peut être "HALFTONE", "ERROR", ou "NONE".

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

```
DisplayMode LORES LACE HAM
```

```
DisplayMode PAL SUPERHIRES DEPTH 8 HAM LACE
```

## 1.42 ARexx port : Whirl

Nom de la commande: Whirl

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

Paramètres:

<X> : Valeur numérique indiquant l'abscisse du centre de l'opération.

<Y> : Valeur numérique indiquant l'ordonnée du centre de

l'opération.

<LEVEL> : Valeur numérique exprimant le degré de rotation du tourbillon.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

Whirl 50 50 200

Whirl 200 200 10

Whirl 200 200 800

### 1.43 ARexx port : Noise

Nom de la commande: Noise

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

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

Paramètres:

<WHITE x> : permet de déterminer le pourcentage de bruit de couleur blanche pour l'image. x doit donc être compris entre 0 et 100.

[RED x] : permet de déterminer le pourcentage de bruit de couleur rouge pour l'image. x doit donc être compris entre 0 et 100.

[GREEN x] : permet de déterminer le pourcentage de bruit de couleur verte pour l'image. x doit donc être compris entre 0 et 100.

[BLUE x] : permet de déterminer le pourcentage de bruit de couleur bleue pour l'image. x doit donc être compris entre 0 et 100.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

---

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 ARexx port : IsBitmap

Nom de la commande: IsBitmap

Syntaxe: IsBitmap [BANK 1-5]

Paramètres:

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

Retourne "OK" si la banque possède un buffer bitmap, ou "ERROR" sinon.

Exemple:

```
IsBitmap BANK 2
```

```
IsBitmap
```

## 1.45 ARexx port : Is24Bits

Nom de la commande: Is24Bits

Syntaxe: Is24Bits [BANK 1-5]

Paramètres:

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

Retourne "OK" si la banque possède un buffer 24Bits, ou "ERROR" sinon.

Exemple:

```
Is24Bits BANK 5
```

```
Is24Bits
```

## 1.46 ARexx port : Mosaic

Nom de la commande: Mosaic

Syntaxe: Mosaic <SIZE> [CRACKLING]

Paramètres:

<SIZE> : Taille d'une case de mosaïque, en pixels.

[CRACKLING] : Si ce mot clef est spécifié, l'opérateur donnera un effet de craquellement à la mosaïque.

[BANK 1-5] : La commande agit sur la banque dont le numéro suit le mot-clef. Si cette précision est omise, la commande agit par défaut sur l'image en cours.

Retour:

"OK" si tout se passe correctement, ou sinon une chaîne débutant par "ERROR", et contenant l'explication de l'erreur.

Exemple:

MOSAIC 10 CRACKLING

MOSAIC 5

MOSAIC 50 CRACKLING