

Arexx_PerfectPaint

Georges HALVADJIAN"

Copyright © Copyright 1998-1999 Georges HALVADJIAN"

COLLABORATORS

	<i>TITLE :</i> Arexx_PerfectPaint		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Georges HALVADJIAN"	July 10, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Arexx_PerfectPaint	1
1.1	Commandes Arexx	1
1.2	Commandes Arexx par Fonction	7
1.3	Inverse	13
1.4	New	13
1.5	Rot90	13
1.6	pp_Rotate	14
1.7	FlipBrushX	14
1.8	FlipBrushY	14
1.9	FlipX	14
1.10	pp_AliasOff	14
1.11	pp_AliasOn	14
1.12	pp_GetAlias	14
1.13	pp_GetPenSize	15
1.14	pp_GetPenType	15
1.15	FlipY	15
1.16	Cycle	15
1.17	pp_BrushEffect	16
1.18	pp_ClearRange	16
1.19	pp_FishEyesLens	16
1.20	Get color value	16
1.21	pp_Gradient	17
1.22	pp_RestoreBrush	17
1.23	pp_Ripple	17
1.24	pp_SetRange	17
1.25	Gradient exemple	18
1.26	pp_Sin	18
1.27	pp_Straw	18
1.28	pp_Twirl	19
1.29	pp_Wave	19

1.30 Integer	19
1.31 Button	20
1.32 CheckBox	20
1.33 Slider	21
1.34 String	22
1.35 Exemple de dialogue	23
1.36 Getfile	23
1.37 Text	24
1.38 DialogInit	25
1.39 Dialog	26
1.40 GetDialog	26
1.41 Fausses couleurs	27
1.42 Transparence	27
1.43 Degradé de gris	27
1.44 Threshold	28
1.45 Spray	28
1.46 Airbrush	28
1.47 NextFrame	28
1.48 PreviousFrame	28
1.49 GotoFrame	29
1.50 CountFrames	29
1.51 MakeAnim	29
1.52 GetApen	29
1.53 GetBpen	29
1.54 Plot	29
1.55 MultiRequest	30
1.56 LongRequest	30
1.57 AddPoly	30
1.58 Exemple	30
1.59 Asay	31
1.60 AskFile	31
1.61 Box	31
1.62 BoxF	32
1.63 CircleF	32
1.64 CircleF	32
1.65 CloseAsay	32
1.66 ClosestColor	32
1.67 Cls	32
1.68 EffectOn	33

1.69 EffectOff	33
1.70 Ellipse	33
1.71 EllipseF	33
1.72 EndPoly	33
1.73 EndPoint	33
1.74 EndPolyF	34
1.75 EndPolyS	34
1.76 EndPolySF	34
1.77 Fill	34
1.78 GetDepth	34
1.79 GetHeight	35
1.80 GetWidth	35
1.81 GetDepthB	35
1.82 GetHeightB	35
1.83 GetWidthB	35
1.84 Light	35
1.85 Line	36
1.86 Load	36
1.87 Bload	36
1.88 MaskOn	36
1.89 MaskOff	36
1.90 PenSize	36
1.91 PenType	37
1.92 Refresh	37
1.93 Request	37
1.94 Save	37
1.95 Bsave	37
1.96 Psave	38
1.97 SetApen	38
1.98 SetBpen	38
1.99 Shade	38
1.100ShowScreen	38
1.101SareOnOff	38
1.102Spline	38
1.103SetBrush	39
1.104Pload	39
1.105SplineF	39
1.106StartPoly	39
1.107UpdateUndo	39

1.108Undo	39
1.109Warn	40
1.110Warp	40
1.111TextAntialiasing	40
1.112TextColorMake	40
1.113TextSize	41
1.114TextRotate	41
1.115TextItalic	41
1.116TextXDPI	41
1.117TextYDPI	41
1.118TextForceType	42
1.119TextStyle	42
1.120TextMake	42
1.121TextDraw	43
1.122TextEffect	43
1.123TextFontName	43
1.124Construction d'une brosse texte (exemple)	43
1.125PictoSpare	44
1.126SparetoPict	44
1.127Spare	44
1.128GetSizeText	44
1.129Displace	45
1.130ClosestColorD	45
1.131Askmultifile	45
1.132BrushOpacity	45
1.133Compose	46
1.134ComposeReqOff	46
1.135ComposeReqOn	46
1.136ConvInit	46
1.137Conv	47
1.138FreeBrush	47
1.139GetWidthT	47
1.140GetHeightT	47
1.141PickBrush	47
1.142Render	47
1.143RenderReq	48
1.144SavePrefs	48
1.145ScaleBrush	49
1.146UpdateUndoBox	49
1.147ClosestColorL	49

Chapter 1

Arexx_PerfectPaint

1.1 Commandes Arexx

IMPORTANT: Ne pas rentrer d'espace avec une fonction Arexx, ←
utilisez le signe "*"

Ex: pp_Warn 'Make*an*anim*first.'
et non
pp_Warn 'Make an Anim first.'

COMMANDES AREXX PAR ORDRE ALPHABETIQUE

Commandes arexx par fonction

A

pp_AddPoly
pp_Airbrush
pp_AliasOff
pp_AliasOn
pp_Asay
pp_AskFile
pp_AskMultiFile

B

pp_Box
pp_BoxF
pp_Bload

pp_BrushEffect
V2

pp_BrushOpacity

pp_Bsave

pp_Button
C

pp_CheckBox

pp_Circle

pp_CircleF

pp_ClearRange
V2

pp_CloseAsay

pp_ClosestColor

pp_ClosestColorD

pp_ClosestColorL

pp_Cls

pp_Compose

pp_ComposeReqOff

pp_ComposeReqOn

pp_Conv

pp_ConvInit

pp_CountFrames

pp_Cycle
D

pp_Dialog

pp_DialogInit

pp_Displace
E

pp_EffectOff

pp_EffectOn

pp_Ellipse

pp_EllipseF
pp_EndPoint
pp_EndPoly
pp_EndPolyF
pp_EndPolyS
pp_EndPolySF
F
pp_False
pp_Fill
pp_FishEyesLens
V2
pp_FlipX
pp_FlipY
pp_FlipBrushX
pp_FlipBrushY
pp_FreeBrush
G
pp_GetAlias
pp_GetApen
pp_GetBlueApen
V2
pp_GetBlueBpen
V2
pp_GetBpen
pp_GetDepth
pp_GetDepthB
pp_GetDialog
pp_GetFile
pp_GetGreenApen
V2
pp_GetGreenBpen
V2

pp_GetHeight
pp_GetHeightB
pp_GetHeightT
pp_GetJiffies
pp_GetPenSize
pp_GetPenType
pp_GetRedApen
V2
pp_GetRedBpen
V2
pp_GetSizeText
pp_GetWidth
pp_GetWidthB
pp_GetWidthT
pp_GotoFrame
pp_Gradient
V2
pp_Grey
I
pp_Inv
pp_Integer
L
pp_Light
pp_Line
pp_Load
pp_LongRequest
M
pp_MakeAnim
pp_MaskOff
pp_MaskOn
pp_MultiRequest
N

pp_NextFrame

pp_New
P

pp_PenSize

pp_PenType

pp_PicttoSpare

pp_PickBrush

pp_Pload

pp_Plot

pp_PreviousFrame

pp_Psave
R

pp_Refresh

pp_Render

pp_RenderReq

pp_Request

pp_RestoreBrush
V2

pp_Ripple
V2

pp_Rot90

pp_Rotate
S

pp_Save

pp_SavePrefs

pp_ScaleBrush

pp_SetApen

pp_SetBpen

pp_SetBrush
pp_SetJiffies

pp_SetRange
V2

pp_Shade
pp_ShowScreen
pp_Sin
V2
pp_Slider
pp_SparetoPict
pp_Spare
pp_SpareOnOff
pp_Spline
pp_SplineF
pp_Spray
pp_StartPoly
pp_Straw
V2
pp_String
T
pp_Text
pp_TextAntialiasing
pp_TextColorMake
pp_TextDraw
pp_TextEffect
pp_TextFontName
pp_TextForceType
pp_TextItalic
pp_TextMake
pp_TextRotate
pp_TextSize
pp_TextStyle
pp_TextXDPI
pp_TextYDPI

pp_Threshold

pp_Trans

pp_Twirl
V2

U

pp_Undo

pp_UpdateUndo

pp_UpdateUndoBox
W

pp_Warn

pp_Warp

pp_Wave
V2

1.2 Commandes Arexx par Fonction

IMPORTANT: Ne pas rentrer d'espace avec une fonction Arexx, ←
utilisez le signe "*"

Ex: pp_Warn 'Make*an*anim*first.'
et non
pp_Warn 'Make an Anim first.'

COMMANDES AREXX PAR FONCTION

Commandes arexx par ordre alphabétique

Dessin

pp_AddPoly

pp_Box

pp_BoxF

pp_Circle

pp_CircleF

pp_Cls
pp_Ellipse
pp_EllipseF
pp_EndPoint
pp_EndPoly
pp_EndPolyF
pp_EndPolyS
pp_EndPolySF
pp_Fill
pp_FreeBrush
pp_GetApen
pp_GetBlueApen
V2
pp_GetBlueBpen
V2
pp_GetBpen
pp_GetDepth
pp_GetDepthB
pp_GetGreenApen
V2
pp_GetGreenBpen
V2
pp_GetHeight
pp_GetHeightB
pp_GetHeightT
pp_GetPenSize
pp_GetPenType
pp_GetWidth
pp_GetWidthB
pp_GetWidthT
pp_Line

pp_PenSize
pp_PenType
pp_PicttoSpare
pp_PickBrush
pp_Plot
pp_GetRedApen
V2
pp_GetRedBpen
V2
pp_SetApen
pp_SetBpen
pp_SetBrush
pp_SparetoPict
pp_Spline
pp_SplineF
pp_StartPoly
Text
pp_TextAntialiasing
pp_TextColorMake
pp_TextDraw
pp_TextEffect
pp_TextFontName
pp_TextForceType
pp_TextItalic
pp_TextMake
pp_TextRotate
pp_GetSizeText
pp_TextSize
pp_TextStyle
pp_TextXDPI

pp_TextYDPI
GUI

pp_Asay

pp_AskFile

pp_AskMultiFile

pp_Button

pp_CheckBox

pp_CloseAsay

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

pp_GetFile

pp_Integer

pp_LongRequest

pp_MultiRequest

pp_Request

pp_Slider

pp_String

pp_Text

pp_Warn

Entrées/Sorties

pp_Bload

pp_Bsave

pp_Load

pp_New

pp_Pload

pp_Psave

pp_Render

pp_RenderReq
pp_Save
pp_SavePrefs
Misc
pp_AliasOff
pp_AliasOn
pp_BrushEffect
V2
pp_BrushOpacity
pp_ClosestColor
pp_ClosestColorD
pp_ClosestColorL
pp_FlipX
pp_FlipY
pp_FlipBrushX
pp_FlipBrushY
pp_RestoreBrush
V2
pp_GetAlias
pp_MaskOff
pp_MaskOn
pp_PicttoSpare
pp_Refresh
pp_Rot90
pp_ScaleBrush
pp_ShowScreen
pp_SpareOnOff
pp_SparetoPict
pp_Undo
pp_UpdateUndo

pp_UpdateUndoBox
Animation

pp_CountFrames
pp_GetJiffies

pp_GotoFrame

pp_MakeAnim

pp_NextFrame

pp_PreviousFrame
pp_SetJiffies

Effets

pp_Airbrush

pp_ClearRange
V2

pp_Compose

pp_ComposeReqOff

pp_ComposeReqOn

pp_Conv

pp_ConvInit

pp_Displace

pp_EffectOff

pp_EffectOn

pp_False

pp_FishEyesLens
V2

pp_Gradient
V2

pp_Grey

pp_Inv

pp_Light

pp_Ripple
V2

pp_Rotate

pp_SetRange
V2

pp_Shade

pp_Sin
V2

pp_Spare

pp_Spray

pp_Straw
V2

pp_Threshold

pp_Trans

pp_Twirl
V2

pp_Warp

pp_Wave
V2

1.3 Inverse

pp_Inv

Active l'effet Inversion des couleurs.

1.4 New

pp_New width height bits

Crée une nouvelle image de dimension width x height x bits

1.5 Rot90

pp_Rot90

Entraine une rotation de 90\textdegree{} de l'image courante

1.6 pp_Rotate

pp_Rotate x

Active l'effet Rotation

x: (-180\textdegree{};180\textdegree{})

1.7 FlipBrushX

pp_FlipBrushX

Entraine un retournement de la brosse sur l'axe des X

1.8 FlipBrushY

pp_FlipBrushY

Entraine un retournement de la brosse sur l'axe des Y

1.9 FlipX

pp_FlipX

Entraine un retournement de l'Image courante sur l'axe des X

1.10 pp_AliasOff

pp_AliasOff

Désactive l'Antialiasing

1.11 pp_AliasOn

pp_AliasOn

Active l'Antialiasing

1.12 pp_GetAlias

pp_GetAlias

Retourne 0, si l'antialiasing est desactivé ou 1 si il est activé.

1.13 pp_GetPenSize

pp_GetPenSize

Retourne la taille du crayon en cours.

1.14 pp_GetPenType

pp_GetPenType

Retourne le type du crayon en cours.

1.15 FlipY

pp_FlipY

Entraine un retournement de l'Image courante sur l'axe des Y

1.16 Cycle

```
pp_Cycle N\textdegree{} x y w h text$ flags options default
```

Initialise le Gadget Cycle

N\textdegree{}: Numéro du gadget (max 10)

x: Position x

y: Position y

w: Largeur du gadget

h: Hauteur du gadget

text\$: Titre du Gadget

flags: flags

options

default

Ex de dialogue

Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

```
pp_GetFile  
pp_Integer  
pp_Slider  
pp_String  
pp_Text
```

1.17 pp_BrushEffect

```
pp_BrushEffect type x y z A  
type: 0 perspective/Rotation  
      1 Shear x  
      2 Shear y  
  
x, y, z :rotation  
  
A: 0 no antialiasing  
   1 Low antialiasing  
   2 high antialiasing
```

1.18 pp_ClearRange

```
pp_ClearRange r  
  
r: Number of the range (1-3)  
  
Clear Range number r
```

Gradient exemple

1.19 pp_FishEyesLens

```
pp_FishEyesLens x  
  
Activate l'effet "Fish eyes lens"  
  
x: -60,60
```

1.20 Get color value

```
pp_GetRedApen  
pp_GetGreenApen  
pp_GetBlueApen
```

```
pp_GetRedBpen  
pp_GetGreenBpen  
pp_GetBlueBpen
```

Retourne la valeur Rouge, verte ou bleue de la couleur de premier plan (Apen) ou d'arrière plan (Bpen).

1.21 pp_Gradient

```
pp_Gradient Nb Type x y
```

Nb: Numéro du range
Type: 0 Horizontal
1 Vertical
2 Radial Sun
3 Warp H
4 Warp V
5 Radial Warp

x and y: centre de l'effet Radial.

Gradient exemple

1.22 pp_RestoreBrush

```
pp_RestoreBrush
```

1.23 pp_Ripple

```
pp_Ripple x
```

Active l'effet Ripple

x: 1-50

1.24 pp_SetRange

```
pp_SetRange r p Red Green Blue
```

Construit le range numero r (0-2) et place la couleur RGB à la position p (0-19)

Gradient exemple

1.25 Gradient exemple

```
pp_EffectOn
/* Active le mode effet */

pp_Gradient 0 2 100 100
/* Active l'effet Gradient
   range=0
   Type=2 (Radian sun)
   100, 100: centre du soleil */

pp_ClearRange 0
/* Efface le Range=0 */

pp_setRange 0 4 0 0 0
/* remplit le range 0 avec à la position 4
   la couleur RGB=(0,0,0) */

pp_setRange 0 19 255 255 255
/* remplit le range 0 avec à la position 19
   la couleur RGB=(255,255,255) */

pp_CircleF 100 100 100
/* Dessine un cercle */
```

1.26 pp_Sin

```
pp_Sin Ay Fx Ax Fy

Active l'effet Sin

Ay (-200,200)
Fx (100-3000)

Ax (-200,200)
Fy (100-3000)
```

1.27 pp_Straw

```
pp_Straw a b

Active l'effet Straw

a:(1-50) Espace entre 2 straws
```

b:(2-50) Taille de chaque straw

1.28 pp_Twirl

pp_Twirl a

Active l'effet Twirl

a:(-720,720) angle

1.29 pp_Wave

pp_Wave a n

Active l'effet Wave

a:(-300,300) Amplitude

b:(1-300) nombre de vagues

1.30 Integer

```
pp_Integer N\textdegree{} x y w h text$ flags default
```

Initialise le Gadget Integer

N\textdegree{}: Numéro du gadget (max 10)

x: Position x

y: Position y

w: Largeur du gadget

h: Hauteur du gadget

text\$: Titre du Gadget

flags: flags

default

Ex de dialogue

Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

pp_GetFile

```
pp_Integer  
pp_Slider  
pp_String  
pp_Text
```

1.31 Button

```
pp_Integer N\textdegree{} x y w h text$
```

Initialise le Gadget Button

```
N\textdegree{}: Numéro du gadget (max 10)  
x: Position x  
y: Position y  
w: Largeur du gadget  
h: Hauteur du gadget  
text$: Titre du Gadget
```

```
Ex de dialogue  
Voir aussi:
```

```
pp_CheckBox  
pp_Cycle  
pp_Dialog  
pp_DialogInit  
pp_GetDialog  
pp_GetFile  
pp_Integer  
pp_Slider  
pp_String  
pp_Text
```

1.32 CheckBox

```
pp_CheckBox N\textdegree{} x y w h text$ flags
```

Initialise le Gadget CheckBox

```
N\textdegree{}: Numéro du gadget (max 10)
x: Position x
y: Position y
w: Largeur du gadget
h: Hauteur du gadget
text$: Titre du Gadget
flags: 0 ou 1 (non activé ou activé)
```

Ex de dialogue
Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

pp_GetFile

pp_Integer

pp_Slider

pp_String

pp_Text

1.33 Slider

```
pp_Slider N\textdegree{} x y w h text$ flags min max default
```

Initialise le Gadget Slider

```
N\textdegree{}: Numéro du gadget (max 10)
x: Position x
y: Position y
w: Largeur du gadget
h: Hauteur du gadget
text$: Titre du Gadget
flags
min: valeur minimum
max: valeur maximum
default
```

Ex de dialogue
Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

pp_GetFile

pp_Integer

pp_Slider

pp_String

pp_Text

1.34 String

```
pp_Sring N\textdegree{} x y w h text$ flags max default
```

Initialise le Gadget String

N\textdegree{}: Numéro du gadget (max 10)
x: Position x
y: Position y
w: Largeur du gadget
h: Hauteur du gadget
text\$: Titre du Gadget
flags
max: Nombre max de \$
default

Ex de dialogue
Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

```
pp_GetDialog  
  
pp_GetFile  
  
pp_Integer  
  
pp_Slider  
  
pp_String  
  
pp_Text
```

1.35 Exemple de dialogue

```
/**/  
  
options results  
parse ARG Port b  
  
ADDRESS value Port  
  
/* Initialisation d'une boite de dialogue comprenant 2 Gadget */  
pp_DialogInit 320 200 "*ESSAI*" 2  
  
/* Initialisation du gadget slider  
pp_Slider 0 100 5 100 16 "Slider" 1 10 200 50  
  
/* Initialisation du gadget Cycle */  
pp_Cycle 1 100 25 100 16 "Cycle" 1 "1|2|3|4|5|6" 2  
  
/* Affichage de la boite de dialogue */  
pp_Dialog  
  
rc=result  
if rc=0 then          /* le gadget 'Cancel' a été sélectionné, on */  
do                   /* quitte le programme */  
    EXIT  
end  
  
pp_GetDialog 0        /* on récupère les contenu du gadget 0 */  
S=result  
  
pp_GetDialog 1        /* on récupère les contenu du gadget 1 */  
C=result  
  
/* ..... */
```

1.36 Getfile

```
pp_Getfile N\textdegree{} x y gt titre$ type path$
```

Initialise le Gadget GetFile

```
N\textdegree{}: Numéro du gadget (max 10)
x: Position x
y: Position y
gt: N\textdegree{} du gadget String associé
titre$
type: 0 Requester de fichier
      1 Requester de chemin
path$: Chemin par défaut
```

Ex de dialogue
Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

pp_GetFile

pp_Integer

pp_Slider

pp_String

pp_Text

1.37 Text

```
pp_Text N\textdegree{} x y text$
```

Initialise le Gadget Text

```
N\textdegree{}: Numéro du gadget (max 10)
x: Position x
y: Position y
text$
```

Ex de dialogue

Voir aussi:

pp_CheckBox
pp_Cycle
pp_Dialog
pp_DialogInit
pp_GetDialog
pp_GetFile
pp_Integer
pp_Slider
pp_String
pp_Text

1.38 DialogInit

pp_DialogInit w h titre\$ Nb

Initialise une boite de dialogue
Deux boutons seront ajoutés , "Ok" et "Cancel"
Ainsi que le design spécifique de PerfectPaint

w: Largeur de la boite
h: Hauteur de la boite
titre\$
Nb: Nombre de gadget a initialiser

Ex de dialogue
Voir aussi:

pp_CheckBox
pp_Cycle
pp_Dialog
pp_DialogInit
pp_GetDialog
pp_GetFile

```
pp_Integer  
pp_Slider  
pp_String  
pp_Text
```

1.39 Dialog

```
pp_Dialog
```

Affiche la fenêtre de dialogue ainsi que tous les gadgets initialisés.

La valeur retournée:

```
0: Si le gadget 'Cancel' a été sélectionné  
1: Si le gadget 'Ok' a été sélectionné
```

```
Ex de dialogue  
Voir aussi:
```

```
pp_CheckBox  
pp_Cycle  
pp_Dialog  
pp_DialogInit  
pp_GetDialog  
pp_GetFile  
pp_Integer  
pp_Slider  
pp_String  
pp_Text
```

1.40 GetDialog

```
pp_GetDialog gt
```

retourne la valeur contenu dans le gadget numéro gt

Ex de dialogue
Voir aussi:

pp_CheckBox

pp_Cycle

pp_Dialog

pp_DialogInit

pp_GetDialog

pp_GetFile

pp_Integer

pp_Slider

pp_String

pp_Text

1.41 Fausses couleurs

pp_False

Active l'effet Fausses couleurs.

1.42 Transparence

pp_Trans x

Active l'effet Transparence.
x: 0-100%

1.43 Dégradé de gris

pp_Grey

Active l'effet Grey.

1.44 Threshold

pp_Threshold x

Active l'effet Threshold.
x: 0-255

1.45 Spray

pp_Spray Couleur x

Active l'effet Spray.

Couleur: 0 Couleur courante
1 dégradé de couleurs entre Bg et Fg
2 Page spare

x: (Amplitude) 1-100%

1.46 Airbrush

pp_Airbrush Opacité Flow

Active l'effet Airbrush.

Opacité: 0-100%
Flow: 0-30

1.47 NextFrame

pp_NextFrame

Active l'image suivante
(si une animation a été créée).

1.48 PreviousFrame

pp_PreviousFrame

Active l'image précédente
(si une animation a été créée).

1.49 GotoFrame

pp_GotoFrame n

Active l'image numéro n
(si une animation a été créée).

1.50 CountFrames

pp_CountFrames

Retourne le nombre d'images de l'animation en cours.
0: Pas d'animation créée

1.51 MakeAnim

pp_MakeAnim n

Construit une animation de n images.
Si une animation est déjà présente, elle sera détruite.

1.52 GetApen

pp_GetApen

Retourne la couleur de premier plan en cours

1.53 GetBpen

pp_GetBpen

Retourne la couleur de second plan en cours

1.54 Plot

pp_Plot x y

Affiche l'objet courant a la position x,y de l'image courante.
(brosse, crayon...)

1.55 MultiRequest

```
pp_MultiRequest n1$ n2$ n3$
```

Affiche une requête au centre de l'écran.
Cette requête affichera un ou plusieurs boutons et retournera le numéro du bouton sélectionné.

n1\$: Titre de la requête
n2\$: Texte de la requête
n3\$: Bouton(s) séparé(s) par "|"

Le bouton à l'extrême droite retournera toujours la valeur 0 (il est habituellement utilisé pour le 'Cancel')
La numérotation des autres boutons est normale (1,2,...), de la gauche vers la droite.

1.56 LongRequest

```
pp_LongRequest n1$ n2$
```

Affiche une requête au centre de l'écran.
Cette requête attendra l'entrée d'une valeur numérique.

n1\$: Titre de la requête
n2\$: Texte de la requête

1.57 AddPoly

```
pp_AddPoly x/N y/N
```

Ajoute un point de coordonnées x,y a la construction d'un polygone ou un ensemble de points

Exemple

1.58 Exemple

```
pp_StartPoly
```

```
pp_AddPoly  
x y
```

```
pp_AddPoly  
x1 y1
```

```
...
```

```
pp_EndPoly
    /* Construit un polygone vide */
ou

pp_EndPolyF
    /* Construit un polygone plein */
ou

PP_EndPolyS
    /* lisse le polygone avec un algo de Bezier */
ou

PP_EndPolyS
    /* lisse le polygone avec un algo de Bezier et le remplit ←
    */
ou

PP_EndPoint
    /* trace juste l'ensemble de points */
```

1.59 Asay

```
pp_Asay message$
```

Affiche un message asynchrone
Ce message doit être ensuite refermé par:
pp_CloseAsay

1.60 AskFile

```
pp_AskFile nom$
```

Ouvre un requester de fichier et retourne le nom du fichier
selectionné.

1.61 Box

```
pp_Box x y x1 y1
```

Dessine une boîte vide de coordonnées x,y (coin haut gauche de la boîte) et
 $x1,y1$ (coin bas droit de la boîte).

Ex: pp_Box 10 10 150 30

1.62 BoxF

```
pp_BoxF x y x1 y1
```

Dessine une boîte pleine de coordonnées x,y (coin haut gauche de la boîte) et $x1,y1$ (coin bas droit de la boîte).

Ex: `pp_BoxF 10 10 150 30`

1.63 CircleF

```
pp_Circle x y r
```

Dessine un cercle vide
 x,y : Coordonné du centre du cercle
 r : rayon du cercle

1.64 CircleF

```
pp_CircleF x y r
```

Dessine un cercle plein
 x,y : Coordonné du centre du cercle
 r : rayon du cercle

1.65 CloseAsay

```
pp_CloseAsay
```

Ferme la boîte de message asynchrone

1.66 ClosestColor

```
pp_ClosestColor r g b
```

Retourne le numéro de couleur de la palette en cours dont les données RGB sont les plus proches de rgb
 r,g,b doivent être compris entre 0 et 255.

1.67 Cls

```
pp_Cls
```

Efface l'image en cours (tient en compte la couleur en cours et le stencil)

1.68 EffectOn

```
pp_EffectOn
```

Active le mode 'Effet'

1.69 EffectOff

```
pp_EffectOff
```

Desactive le mode 'Effet'

1.70 Ellipse

```
pp_Ellipse x y r1 r2
```

Dessine une ellipse vide
x,y:Coordonnés du centre de l'ellipse
r1: rayon de l'ellipse en x
r2: rayon de l'ellipse en y

1.71 EllipseF

```
pp_EllipseF x y r1 r2
```

Dessine une ellipse pleine
x,y:Coordonnés du centre de l'ellipse
r1: rayon de l'ellipse en x
r2: rayon de l'ellipse en y

1.72 EndPoly

```
pp_EndPoly
```

Construit un polygone vide.

Exemple

1.73 EndPoint

pp_EndPoint

Dessine juste l'ensemble de point

Exemple

1.74 EndPolyF

pp_EndPolyF

Construit un polygone plein.

1.75 EndPolyS

pp_EndPolyS

Lisse le polygone avec un algo de Bezier et le dessine.

Exemple

1.76 EndPolySF

pp_EndPolySF

Lisse le polygone avec un algo de Bezier, le remplit et le dessine.

Exemple

1.77 Fill

pp_Fill x y

Flood fill a partir du point de coordonnées x et y

1.78 GetDepth

pp_GetDepth

Retourne la profondeur de l'image en cours.

0: Pas d'image en cours

1.79 GetHeight

pp_GetHeight

Retourne la hauteur de l'image en cours.
0: Pas d'image en cours

1.80 GetWidth

pp_GetWidth

Retourne la largeur de l'image en cours.
0: Pas d'image en cours

1.81 GetDepthB

pp_GetDepthB

Retourne la profondeur de la brosse en cours.
0: Pas de brosse en cours

1.82 GetHeightB

pp_GetHeightB

Retourne la hauteur de la brosse en cours.
0: Pas de brosse en cours

1.83 GetWidthB

pp_GetWidthB

Retourne la largeur de la brosse en cours.
0: Pas de brosse en cours

1.84 Light

pp_Light x

Active l'effet Light
x: 0-100%

1.85 Line

```
pp_Line x y x1 y1
```

Dessine une ligne de A(x,y) vers B(x1,y1)

1.86 Load

```
pp_Load nom$
```

Charge une image

nom\$: nom de l'image avec le chemin

1.87 Bload

```
pp_Bload nom$
```

Charge une brosse

nom\$: nom de la brosse avec le chemin

1.88 MaskOn

```
pp_MaskOn
```

Active le mode 'Mask'

1.89 MaskOff

```
pp_MaskOff
```

Desactive le mode 'Mask'

1.90 PenSize

```
pp_PenSize n
```

Determine la taille du crayon

1.91 PenType

pp_PenType n

Selectionne le type de crayon.

0: le point
1: le cercle plein
2: le carré plein
3: AntiSlash
4: Réservé
5: Réservé
6: Brosse

1.92 Refresh

pp_Refresh

Raffraichit la fenêtre avec l'image courante

1.93 Request

pp_Request message\$

Affiche un message comprenant deux boutons: 'Yes' et ' No'
et retourne 0, si 'No' a été sectionné ou 1, si 'Yes' a été
selectionné.

1.94 Save

pp_Save nom\$ a

Sauvegarde de l'image en cours

a: 0 (L'image est sauvée sans icone)
1 (L'image est sauvée avec une icone)

1.95 Bsave

pp_Bsave nom\$ a b

Sauvegarde de la brosse en cours

a: 0 (La brosse est sauvée sans icone)
1 (La brosse est sauvée avec une icone)

b: 0 1-8Bits
1 24Bits sans Masque
2 24Bits avec Masque
3 24Bits Deep brush

1.96 Psave

pp_Psave nom\$ a

Sauvegarde de la palette en cours

a: 0 (La palette est sauvée sans icone)

1 (La palette est sauvée avec une icone)

1.97 SetApen

pp_SetApen a

Active une nouvelle couleur de premier plan.

1.98 SetBpen

pp_SetBpen a

Active une couleur de second plan.

1.99 Shade

pp_Shade x

Active l'effet Shade

x: 0-100%

1.100 ShowScreen

pp_ShowScreen

Met au premier plan, l'écran de PerfectPaint.

1.101 SpareOnOff

pp_SpareOnOff

Active ou desactive la page spare

1.102 Spline

pp_Spline x y x1 y1 x2 y2

Dessine courbe vide de A(x,y) vers B(x1,y1) en passant par C(x2,y2)

1.103 SetBrush

```
pp_SetBrush n
```

Active la brosse numero n.
n: (1-2)

1.104 Pload

```
pp_Pload nom$
```

Charge la palette nom\$

1.105 SplineF

```
pp_SplineF x y x1 y1 x2 y2
```

Dessine courbe pleine de A(x,y) vers B(x1,y1) en passant par C(x2,y2)

1.106 StartPoly

```
pp_StartPoly
```

Previent PerfectPaint que l'on va commencer a dessiner un polygone ou un ensemble de points.

Exemple

1.107 UpdateUndo

```
pp_UpdateUndo
```

Capture de l'image en cours
(La commande Undo est toujours désactivée pendant le déroulement d'un script Arexx).

1.108 Undo

```
pp_Undo
```

Actionne le Undo

1.109 Warn

```
pp_Warn message$
```

Affiche une alerte avec un bouton 'OK'.

1.110 Warp

```
pp_Warp X Y r Dx Dy
```

Entraine un Warp de l'image en cours.

X: Centre x du Warp

Y: Centre y du Warp

r: Radius

Dx: Décalage en x du centre

Dy: Décalage en y du centre

1.111 TextAntialiasing

```
pp_TextAntialiasing x
```

x: 0 -> Pas d'antialiasing

1 -> Antialiasing de niveau 1

2 -> Antialiasing de niveau 2

Active l'antialiasing, se definit avant de construire une brosse texte.

Construction d'une brosse texte (exemple)

1.112 TextColorMake

```
pp_TextColorMake a$ R
```

a\$: Texte, si a\$="", le texte en cours est utilisé

R : 0 Utilise la palette de la fonte

1 Remap la fonte avec la palette courante

2 Colorize la fonte avec la couleur courante

Construit une brosse texte a l'aide d'une fonte colorisée.

Construction d'une brosse texte (exemple)

1.113 TextSize

pp_TextSize x

x: Taille de la fonte a utiliser

Construction d'une brosse texte (exemple)

1.114 TextRotate

pp_TextRotate x

x: Rotation a effectuer sur la fonte. (Fonte vectorielle seulement)

Construction d'une brosse texte (exemple)

1.115 TextItalic

pp_TextItalic x

x (-45\textdegree,45\textdegree): penchement de la fonte. (Fonte ←
vectorielle seulement)

Construction d'une brosse texte (exemple)

1.116 TextXDPI

pp_TextXDPI x

x : DPI (X) de la fonte a utiliser. (Fonte vectorielle seulement)
defaut=72

Construction d'une brosse texte (exemple)

1.117 TextYDPI


```
pp_TextYDPI x
```

x : DPI (Y) de la fonte a utiliser. (Fonte vectorielle seulement)
defaut=72

Construction d'une brosse texte (exemple)

1.118 TextForceType

```
pp_TextForceType x
```

x -> 0 : Fonte Bitmap
x -> 2 : Fonte en couleur
x -> 3 : Fonte vectorielle

Force PerfectPaint a utiliser un type de fonte

Construction d'une brosse texte (exemple)

1.119 TextStyle

```
pp_TextStyle x
```

x: Style de la fonte (B=bold,I=italique,U=souligné)
0 -> B=0 I=0 U=0
1 -> B=0 I=0 U=1
2 -> B=1 I=0 U=0
3 -> B=1 I=0 U=1
4 -> B=0 I=1 U=0
5 -> B=0 I=1 U=1
6 -> B=1 I=1 U=0
7 -> B=1 I=1 U=1

Construction d'une brosse texte (exemple)

1.120 TextMake

```
pp_TextMake a$
```

a\$: Texte, si a\$="", le texte en cours est utilisé

Construit une brosse texte a l'aide d'une fonte non colorisée.

Construction d'une brosse texte (exemple)

1.121 TextDraw

```
pp_TextDraw x y
```

x et y: coordonnées

Dessine la brosse texte sur la page courante a la position x et y.

Construction d'une brosse texte (exemple)

1.122 TextEffect

```
pp_TextEffect Effect x
```

Effect: 0 Pas d'effet
1 Outline
2 Bevel

x: Amplitude de l'effet

Construction d'une brosse texte (exemple)

1.123 TextFontName

```
pp_TextFontName fonte$
```

fonte\$: nom de la fonte à utiliser

Construction d'une brosse texte (exemple)

1.124 Construction d'une brosse texte (exemple)

```
/* exemple */
```

```
.....
```

```
pp_TextAntialiasing 2
```

```
/* antialiasing de niveau 2 */
/* si vous omettez cette commande, le parametre en cours sera utilisé */

pp_TextEffect 1 2
/* L'effet Outline est utilisé avec une amplitude de 2 */
/* si vous omettez cette commande, le parametre en cours sera utilisé */

pp_TextFontName 'CGTimes.font'
/* si vous omettez cette commande, le parametre en cours sera utilisé */

pp_TextSize 50
/* si vous omettez cette commande, le parametre en cours sera utilisé */

pp_TextStyle 0
/* Pas de Style */
/* si vous omettez cette commande, le parametre en cours sera utilisé */

pp_TextMake 'Bonjour*tout*le*monde'
/* Construction de la brosse texte avec la phrase 'Bonjour tout le monde' */
/* avec pp_TextMake "", le texte en cours sera utilisé */

pp_TextDraw 100 100
/* Dessine la brosse texte aux coordonnées 100,100 */

.....
```

1.125 PicttoSpare

```
pp_PicttoSpare
```

Copie l'image en cours sur la page Spare.

1.126 SparetoPict

```
pp_SparetoPict
```

Copie la page Spare sur l'image en cours.

1.127 Spare

```
pp_Spare
```

Active l'effet Spare

1.128 GetSizeText

```
pp_GetSizeText
```

Renvoie la taille de la fonte en cours.

1.129 Displace

```
pp_Displace x
```

Active l'effet Displace.

x: Taux de dispersion

1.130 ClosestColorD

```
pp_ClosestColorD couleur Darkness
```

Couleur: numero de la couleur

Darkness: pourcentage d'assombrissement (0-100%)

Retourne le numéro de couleur de la palette en cours dont le contraste est le plus proche.

1.131 Askmultifile

```
pp_AskMultiFile Title
```

Title: Title of the requester

This command let you select multiple file.

It writes in ram: a file name 'MultiFile' which content all the selected file

This command return the number of file selected

ex:

```
pp_AskMultiFile 'Load*Pictures'  
count=result
```

```
file="Ram:MultiFile"  
IF OPEN("g",file,"read") then DO  
  DO i=1 to count  
    file=READLN("g")  
    say file  
  END  
  CALL close(file)  
END
```

1.132 BrushOpacity

```
pp_BrushOpacity x
x: 0-100%
```

1.133 Compose

```
pp_Compose Mode Opacity Alpha
Mode: 0 Opacity
      1 Add
      2 Sub
      3 Darken
      4 Alpha
      5 Inv Alpha
Opacity: 0-100%
Alpha: -255 -> 255
```

This compose between the current effect and the current picture.

1.134 ComposeReqOff

With this command, the compose requester will not appear after each effect

1.135 ComposeReqOn

With this command, the compose requester will appear after each effect

1.136 ConvInit

```
pp_ConvInit Dim Div Bias
Dim: 0 3x3
     1 5x5
```

This command initialise a convolution
Use pp_Conv to fill this convolution.

Ex:

```
C=-2
pp_ConvInit 0 1 0 ;initialise the convolution
```

```
pp_Conv 0 0 0 0 0 0 ;fill the convolution
pp_Conv 1 0 C 0 0 0
pp_Conv 2 0 0 5 0 0
pp_Conv 3 0 0 0 C 0
pp_Conv 4 0 0 0 0 0
```

```
pp_BoxF x1 y1 x2 y2 ; draw a box on the picture with the current convolution
```

1.137 Conv

```
pp_Conv y C1 C2 C3 C4 C5
  y: current line
  C1-C5: convolution
```

Use pp_Conv to initialise the convolution

Ex:

```
C=-2
pp_ConvInit 0 1 0 ;initialise the convolution

pp_Conv 0 0 0 0 0 0 ;fill the convolution
pp_Conv 1 0 C 0 0 0
pp_Conv 2 0 0 5 0 0
pp_Conv 3 0 0 0 C 0
pp_Conv 4 0 0 0 0 0

pp_BoxF x1 y1 x2 y2 ; draw a box on the picture with the current convolution
```

1.138 FreeBrush

```
pp_FreeBrush
```

1.139 GetWidthT

```
pp_GetWidthT
  Return the current width of the text
```

1.140 GetHeightT

```
pp_GetHeightT
  Return the current height of the text
```

1.141 PickBrush

```
pp_PickBrush x y width height
```

1.142 Render

```
pp_Render type file
Type: 0 Picture
      1 Brush
      2 Animation
```

file: name of the file to be save

This Command will reduce the number of color of your brush, picture or animation and will save it.

See also

```
pp_RenderReq
```

1.143 RenderReq

```
pp_RenderReq Type
Type: 0 Picture
      1 Brush
      2 Animation
```

Will open then Render requester
Return '0' if 'Cancel' was selected

See also

```
pp_Render
```

1.144 SavePrefs

```
pp_SavePrefs Format ILBMtype JpegQuality JpegSmooth JpegColor
Format: 0 iff24
        1 Jpeg
ILBMtype: 0 Uncompressed
           1 Compressed
           2 8bit gray
JpegQuality 1-100%
JpegSmooth 0-100%
JpegColor: 0 RGB
            1 Gray
```

Use this command before saving your picture with
pp_Save

1.145 ScaleBrush

```
pp_ScaleBrush Width Height
```

This command will scale the current brush.

1.146 UpdateUndoBox

```
pp_UpdateUndoBox x y x1 y1
```

Stores a part of the current image in the undo buffer
(The undo facility is always deactivated during execution of Arexx
scripts)

1.147 ClosestColorL

```
pp_ClosestColorL couleur Brightness
```

Couleur: numero de la couleur
Darkness: pourcentage de luminosité (0-100%)

Retourne le numéro de couleur de la palette en cours dont
la luminosité est la plus proche.
