

Arexx_Savers

Copyright © 1992-1996 Nova Design, Inc.

COLLABORATORS

	<i>TITLE :</i> Arexx_Savers		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 30, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Arexx_Savers	1
1.1	ImageFX Saver Module Arguments	1
1.2	ANIM Module Arguments	1
1.3	ANIM7 Module Arguments	2
1.4	FLC Module Arguments	2
1.5	FLYERCLIP Module Arguments	3
1.6	ICON Module Arguments	3
1.7	ILBM Module Arguments	4
1.8	JPEG Module Arguments	4
1.9	PCX Module Arguments	4
1.10	TARGA Module Arguments	5
1.11	TIFF Module Arguments	5

Chapter 1

Arexx_Savers

1.1 ImageFX Saver Module Arguments

Select A Saver Module:

```
    ANIM
    ANIM7
    FLC
*   FLYERCLIP          (2.6)
    ICON
    ILBM
*   JPEG
    PCX
    TARGA
    TIFF
```

Savers not listed have no Arexx arguments.

1.2 ANIM Module Arguments

ANIM

FORMAT

Append/S,Overwrite/S,Keep/S,Close/S

INPUTS

Append

If writing to an existing file, specifies that this new frame should be appended to the existing animation.

Overwrite

If writing to an existing file, specifies that a new animation should be started.

Keep

Specifies that the animation should be constructed in memory, which speeds up the creation time a LOT. However, using this method you MUST close the animation with the Close option when

it is complete. This method is used by IMP when creating animations.

Close

Specified when you are finished creating a Keep animation and are ready to close the file.

EXAMPLE

```
SaveRenderedAs ANIM DH0:MyAnimation.anim Keep
SaveRenderedAs ANIM DH0:MyAnimation.anim Close
```

1.3 ANIM7 Module Arguments

ANIM7

FORMAT

Close/S

INPUTS

Close

Specified when you are finished creating an animation and are ready to close the file.

EXAMPLE

```
SaveRenderedAs ANIM7 DH0:MyAnimation.anim
SaveRenderedAs ANIM7 DH0:MyAnimation.anim Close
```

1.4 FLC Module Arguments

FLC

FORMAT

Append/S,Overwrite/S,Keep/S,Close/S,NoPalette/S

INPUTS

Append

If writing to an existing file, specifies that this new frame should be appended to the existing animation.

Overwrite

If writing to an existing file, specifies that a new animation should be started.

Keep

Specifies that the animation should be constructed in memory, which speeds up the creation time a LOT. However, using this method you MUST close the animation with the Close option when it is complete. This method is used by IMP when creating animations.

Close

Specified when you are finished creating a Keep animation and

are ready to close the file.

NoPalette (2.0)

Forces the saver to use the same palette for every frame. Ie. only the first frame will contain palette; the rest of the frames will not have any.

EXAMPLE

```
SaveRenderedAs FLC DH0:MyAnimation.anim Keep
SaveRenderedAs FLC DH0:MyAnimation.anim Close
```

1.5 FLYERCLIP Module Arguments

FLYERCLIP (2.6)

FORMAT

Append/S,Overwrite/S,Compression/N

INPUTS

Append

If writing to an existing file, specifies that this new frame should be appended to the existing animation.

Overwrite

If writing to an existing file, specifies that a new animation should be started.

Compression

Compression setting.

EXAMPLE

```
SaveBufferAs FLYERCLIP DH0:MyAnim Append
```

1.6 ICON Module Arguments

ICON (2.0)

FORMAT

Stack/N,CurrentX/N,CurrentY/N,Disk/S,Drawer/S,Tool/S,Project/S

INPUTS

Stack

Sets the stack size for the icon. Defaults to 4096.

CurrentX, CurrentY

Sets the icon position. Defaults to NO_ICON_POSITION.

Disk, Drawer, Tool, Project

Sets the type of icon. Defaults to Project.

EXAMPLE

```
SaveRenderedAs Icon DH0:MyTool Stack 60000 Tool
```

1.7 ILBM Module Arguments

ILBM

The ILBM saver technically has no Arexx arguments, however to save an ILBM file with alpha channel information, use a command similar to the following:

```
SaveBufferAs ILBM+Alpha "DH0:MyPicture.24"
```

This method was chosen for backwards compatibility reasons.

1.8 JPEG Module Arguments

JPEG

FORMAT

Quality/N, Progressive/S, Optimize/S, Worst/S, Best/S

INPUTS

Quality

Specifies the quality setting for the JPEG compression. The usable range is between 25 and 100.

Progressive (2.6)

Write a progressive JPEG file.

Optimize (2.6)

Attempts to optimize the compression before writing. This results in a slightly smaller file, but takes longer to save.

Worst (2.6)

Uses a slightly faster, but less accurate method of saving.

Best (2.6)

Uses a slightly slower, but more accurate method of saving.

EXAMPLE

```
SaveBufferAs JPEG DH0:MyPicture.jpg 85 PROGRESSIVE
```

1.9 PCX Module Arguments

PCX

FORMAT

VGA/S, EGA/S

INPUTS

VGA, EGA

Specifies the type of PCX file to write. This only applies to rendered images.

EXAMPLE

```
SaveRenderedAs PCX DH0:MyPicture.pcx VGA
```

1.10 TARGA Module Arguments

TARGA

FORMAT

```
RLE/S,NoCompress/S,Alpha/S,NoAlpha/S
```

INPUTS

RLE

Specifies RLE compression for the Targa file.

NoCompress

Specifies no compression for the Targa file.

Alpha (2.0)

Attaches an alpha channel to the Targa file if one is present.
(Ie. saves a Targa32 file instead of a Targa24 file.)

NoAlpha (2.0)

Does not save alpha channel information with the Targa file.

EXAMPLE

```
SaveBufferAs Targa DH0:MyPicture.tga RLE  
SaveBufferAs Targa DH0:MyPicture.tga ALPHA
```

1.11 TIFF Module Arguments

TIFF

FORMAT

```
LZW/S,NoCompress/S
```

INPUTS

LZW

Specify LZW compression for the TIFF file.

NoCompress

Specify no compression for the TIFF file.

EXAMPLE

```
SaveBufferAs TIFF DH0:MyPicture.tif NoCompress
```
