

Picture

COLLABORATORS

	<i>TITLE :</i> Picture		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		February 8, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Picture	1
1.1	Picture V1.00	1
1.2	freepicture	1
1.3	initpicture	1
1.4	loadpicture	2
1.5	usepicture	2
1.6	picturetobitmap	2
1.7	picturewidth	3
1.8	pictureheight	3
1.9	picturedepth	3
1.10	pictureid	3

Chapter 1

Picture

1.1 Picture V1.00

Pure Basic Picture library V1.00

The pictures handled by the Pure Basic are stored in IFF/ILBM format, the Amiga picture standard. The `$picture` can be compressed or not, both forms are supported. the read/decompression routines have been highly optimized and even big pictures load in less than a seconds on any amiga. Enjoy...

Commands summary:

```
FreePicture
InitPicture
LoadPicture
PictureDepth
PictureHeight
PictureID
PictureToBitMap
PictureWidth
UsePicture
```

Example:

```
Picture viewer
```

1.2 freepicture

SYNTAX

```
FreePicture(#Picture)
```

STATEMENT

Remove the given Picture from memory.

1.3 initpicture

SYNTAX

```
result.l = InitPicture(#NumPictureMax)
```

FUNCTION

Init all the Picture environments for later use. You must put this function at the top of your source code if you want to use the NPicture commands.

#NumPictureMax : Maximum number of Pictures to handle.

1.4 loadpicture

SYNTAX

```
PictureID.l = LoadPicture(#Picture, FileName$)
```

FUNCTION

LoadPicture() will try to load an IFF/ILBM picture into memory. It will be loaded in ANY memory, ie FAST ram if available. The picture will be kept in compressed form (if this is a compressed IFF/ILBM file, but normally, it is) so no memory is wasted.

The result can take 3 special values:

If PictureID = -1: File is not found

If PictureID = -2: File is not an IFF/ILBM picture

If PictureID = -3: Not enough free memory to load the picture

If PictureID >= 0: All is good.

1.5 usepicture

SYNTAX

```
UsePicture(#Picture)
```

STATEMENT

Change the used Picture to given #Picture.

1.6 picturetobitmap

SYNTAX

```
Result = PictureToBitMap(#Picture, BitMapID)
```

COMMAND

Copy the given IFF/IBLM picture data on the given bitmap. Note, this function is very optimized. It supports compressed and uncompressed IFF pictures.

The result can take 2 specials values:

```
If Result = -1: BitmapHeader or Body chunk aren't found (ie: IFF
    palette only)
If Result = -2: Picture size is too big for the bitmap.
```

1.7 picturewidth

SYNTAX

```
Width.w = PictureWidth()
```

FUNCTION

Returns the current picture width.

1.8 pictureheight

SYNTAX

```
Height.w = PictureHeight()
```

FUNCTION

Returns the current picture Height.

1.9 picturedepth

SYNTAX

```
Depth.b = PictureDepth()
```

FUNCTION

Returns the current picture depth.

1.10 pictureid

SYNTAX

```
*PictureAddr = PictureID()
```

FUNCTION

Returns the current picture memory location. It points to the start of the IFF file ('FORM' chunk).