

Pictures

Paul Manias

COLLABORATORS

	<i>TITLE :</i> Pictures		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Paul Manias	May 28, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Pictures	1
1.1	Object: Picture	1
1.2	Object: Picture	1
1.3	Object: Picture	2
1.4	Object: Picture	3
1.5	Object: Picture	3
1.6	Object: Picture	3
1.7	Object: Picture	4
1.8	Object: Picture	4
1.9	Picture: Init()	4
1.10	Picture: Load()	5
1.11	Picture: Query()	5

Chapter 1

Pictures

1.1 Object: Picture

OBJECT DOCUMENTATION

Name: PICTURE
Version: 0.9 Beta.
Date: December 1997
Author: Paul Manias
Copyright: DreamWorld Productions, 1996-1997. All rights reserved.

CHANGES VERSION 0.9B

Added: Picture->Bitmap

Removed: Picture->AmtColours
Picture->ByteWidth
Picture->Data
Picture->Height
Picture->Planes
Picture->Width
Picture->ScrType
Picture->Options: IMG_BLITMEM, IMG_VIDEOMEM

Edited: Description
Picture->Palette
Picture->Options

1.2 Object: Picture

OBJECT

Name: Picture
Module: Picture
Version: 1
Type: Complex
Children: Bitmap

DESCRIPTION

The purpose of the Picture object is to provide a standard interface that any program can use to load a picture file. The advantage is that the

source picture can be in any file format currently recognised by the system, so effectively the program in question will be able to support file formats that it does not understand.

The Picture object will clip any loaded picture so that it fits the size given in Bitmap->Width and Bitmap->Height. If you specify the RESIZE flag, the picture will be shrunk or enlarged to fit the given dimensions. If you leave the Width and Height at NULL, then the picture will be loaded at its default dimensions.

ACTIONS

The Picture object supports the following actions:

```
CopyStructure()
Free()
Get()
* Init()
* Load()
  OpenFile()
* Query()
```

STRUCTURE

The Picture object consists of the following public fields, plus all fields inherited from the Bitmap object:

```
+ Bitmap      Bitmap child object.
Options      Special flags.
Palette      Pointer to Palette.
ScrHeight    Screen Height (pixels)
ScrMode      Intended screen mode for picture.
ScrWidth     Screen Width (pixels)
Source       Where this picture comes from.
```

1.3 Object: Picture

FIELD

```
Name:      Options
Type:      LONG
On Change: Cannot change after initialisation.
Status:    Read/IWrite
```

DESCRIPTION

You can specify certain flags here that will affect the way the Picture is initialised. Valid flags are:

IMG_REMAP

Remaps the Picture data to fit the palette, as pointed to in the Picture->Palette field. If the Picture is stored as a true colour type, this flag will be ignored.

IMG_RESIZEX

Resizes the Picture so that it fits the given Bitmap->Width. If this flag is not set then the picture will be clipped.

IMG_RESIZEY

Resizes the Picture so that it fits the given Bitmap->Height. If this flag is not set then the picture will be clipped.

IMG_RESIZE

Short way of setting the RESIZEX and RESIZEY flags, above.

1.4 Object: Picture

FIELD

Name: Palette
Type: LONG *
Inheritance: Source picture.
On Change: Dynamic.
Status: Read/IWrite

DESCRIPTION

Points to a palette if it is relevant for the picture's screen type. Palettes are generally only allocated when the amount of colours used in the picture is from 2 - 256.

1.5 Object: Picture

FIELD

Name: ScrHeight
Type: WORD
Inheritance: Source picture (if possible) or estimate based on resolution.
On Change: Dynamic
Status: Read/Write

DESCRIPTION

Specifies the height of the screen/viewport when displaying the picture. Some picture file formats may not contain a suitable value to be placed in this field. In this case, the field will be initialised to a value based on the pictures resolution.

SEE ALSO

Field: ScrWidth

1.6 Object: Picture

FIELD

Name: ScrMode
Type: WORD
Inheritance: Source picture.
On Change: Dynamic
Status: Read/Write

DESCRIPTION

The screen mode that this picture is being loaded into. Applicable flags

are outlined in Screen->ScrMode (LORES, HIRES, SHIRES, LACED etc).

SEE ALSO

Screen: ScrMode

1.7 Object: Picture

FIELD

Name: ScrWidth

Type: WORD

Inheritance: Source picture (if possible) or estimate based on resolution.

On Change: Dynamic

Status: Read/Write

DESCRIPTION

Specifies the width of the screen/viewport when displaying the picture. Some picture file formats may not contain a suitable value to be placed in this field. In this case, the field will be initialised to a value based on the pictures resolution.

SEE ALSO

Field: ScrHeight

1.8 Object: Picture

FIELD

Name: Source

Type: APTR

On Change: Dynamic

Status: Read/IWrite

DESCRIPTION

Pointer to a source object - either FileName, MemPtr or a standard object recognised by the system.

1.9 Picture: Init()

ACTION

Name: Init()

Object: Picture

Short: Initialise a picture object so that it is ready for active use.

DESCRIPTION

If Picture->Source is specified, then this action will first attempt to initialise the Picture by loading the Source data. If the data is recognised, it will be unpacked to the buffer specified in Picture->Data. If you do not supply Picture->Data, then a buffer will be allocated for you and placed in Picture->Data on initialisation.

The Picture object has all of the standard features of system objects,

including auto-initialisation of NULL fields. Note that by setting certain fields you are placing restrictions on the picture that is to be loaded. For example, if the picture is wider than a specified width, the picture will have its right edge clipped. To get around this simply leave the Width field unspecified, and Init() will initialise this field, loading the picture without clipping it. Alternatively you could specify the RESIZE flag, depending on the circumstances.

NOTE

If this action cannot identify the source data, then the call will fail. The native data format for the Picture class is IFF, but other formats such as JPEG and PNG can be added via child classes.

SEE ALSO

Picture: Load()

Include: graphics/pictures.i

1.10 Picture: Load()

ACTION

Name: Load()

Object: Picture

Short: Loads in a file that has been successfully identified as belonging to the picture class.

DESCRIPTION

This action is provided as a quick and simple way of loading in a Picture object. Note that all Pictures loaded using this method will be given space in Video RAM as a preference.

If you require more versatile loading, use the Init() function instead of Load().

SEE ALSO

Picture: Init()

1.11 Picture: Query()

ACTION

Name: Query()

Object: Picture

Short: Get the information on a recognised picture type.

DESCRIPTION

Calling this action will fill out all of the information on a Picture, according to what is found from the Picture->Source or Picture->Header.

Fields that are not set to NULL will be ignored by this action. For this reason, calling Query() after a call to Init() may be a fairly useless exercise.

If the Palette field is empty and the amount of colours is 256 or less,

this function will automatically allocate a memory area for the palette and write out a list of colours.

By using this function you can find information on any picture format currently supported by GMS. If the picture format cannot be assessed, then an error code of `ERR_DATA` will be returned.

SEE ALSO

`Kernel: Init()`