MBS GraphicsMagick Plugin Documentation

Christian Schmitz

March 10, 2024

0.1 Introduction

This is the PDF version of the documentation for the Xojo Plug-in from Monkeybread Software Germany. Plugin part: MBS GraphicsMagick Plugin

0.2 Content

Chapter 1

List of Topics

Chapter 2

List of all classes

Chapter 3

List of all modules

Chapter 4

GraphicsMagick

4.1 class GM16BlobMBS

4.1.1 class GM16BlobMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for binary large objects.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GM16ImageMBS
dim blob as new GM16BlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

Notes: Blob provides the means to contain any opaque data. It is named after the term "Binary Large OBject" commonly used to describe unstructured data (such as encoded images) which is stored in a database. While the function of Blob is very simple (store a pointer and and size associated with allocated data), the Blob class provides some very useful capabilities. In particular, it is fully reference counted just like the Image class.

The Blob class supports value assignment while preserving any outstanding earlier versions of the object. Since assignment is via a pointer internally, Blob is efficient enough to be stored directly in an STL container or any other data structure which requires assignment. In particular, by storing a Blob in an associative container (such as STL's 'map') it is possible to create simple indexed in-memory "database" of Blobs.

Magick++ currently uses Blob to contain encoded images (e.g. JPEG) as well as ICC and IPTC profiles. Since Blob is a general-purpose class, it may be used for other purposes as well.

4.1.2 Methods

4.1.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Default constructor creating an empty blob object.
See also:

- 4.1.4 Constructor(data as memoryblock, offset as Integer, size as Integer) 88
- 4.1.5 Constructor(data as string) 88
- 4.1.6 Constructor(other as GM16BlobMBS) 89

4.1.4 Constructor(data as memoryblock, offset as Integer, size as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct object with data, making a copy of the supplied data.
See also:

- 4.1.3 Constructor 88
- 4.1.5 Constructor(data as string) 88
- 4.1.6 Constructor(other as GM16BlobMBS) 89

4.1.5 Constructor(data as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct object with data, making a copy of the supplied data.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GM16ImageMBS
dim blob as new GM16BlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

See also:

- 4.1.3 Constructor 88
- 4.1.4 Constructor(data as memoryblock, offset as Integer, size as Integer) 88
- 4.1.6 Constructor(other as GM16BlobMBS) 89

4.1.6 Constructor(other as GM16BlobMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copy constructor (reference counted).
See also:

- 4.1.3 Constructor 88
- 4.1.4 Constructor(data as memoryblock, offset as Integer, size as Integer) 88
- 4.1.5 Constructor(data as string) 88

4.1.7 CopyMemory as memoryblock

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a copy of the data as a memoryblock.
Notes: Returns nil on any error like low memory.

4.1.8 CopyString as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a copy of the data as a string.

4.1.9 Data as Ptr

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A memoryblock with the data from this blob.
Example:

```
dim b as new GM16BlobMBS("Hello")

dim m as memoryblock = b.Data
MsgBox m.StringValue(0,5) // shows "Hello"
```

Notes: This is a memoryblock referencing the data of the blob. It has no size set. The memoryblock can only be used as long as the blob object exists. if you use it after you destroyed the blob object, you can crash you application.

4.1.10 Update(data as memoryblock, offset as Integer, size as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces the content of this blob with a copy of the bytes in the memoryblock.
See also:

- 4.1.11 Update(data as string) 90

4.1.11 Update(data as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces the content of this blob with a copy of the bytes in the string.
Notes: Offset is zero based.
See also:

- 4.1.10 Update(data as memoryblock, offset as Integer, size as Integer) 90

4.1.12 Properties

4.1.13 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal handle of the blob object.
Notes: (Read and Write property)

4.1.14 length as UInt64

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain data length in bytes.
Example:

```
dim b as new GM16BlobMBS(”Hello”)

MsgBox str(B.length) // shows 5
```

Notes: (Read only property)

4.1.15 base64 as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blob content as a string in Base64 format.
Example:

```
dim b as new GM16BlobMBS(”Hello”)

MsgBox b.base64 // shows ”SGVsbG8=”
```

Notes: (Read and Write computed property)

4.2 class GM16CoderInfoMBS

4.2.1 class GM16CoderInfoMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class used to get information about all registered coders.
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GM16CoderInfoMBS in coders
names.Append coder.name
next

MsgBox Join(names,EndOfLine)
```

Notes: The CoderInfo class provides the means to provide information regarding GraphicsMagick support for an image format (designated by a magick string). It may be used to provide support for a specific named format (provided as an argument to the constructor), or as an element of a container when format support is queried using the coderInfoList() templated function.

4.2.2 Methods

4.2.3 CoderInfoList as GM16CoderInfoMBS()

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a list of all coders.
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
```

4.2.4 Properties

4.2.5 description as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format description (e.g. ”CompuServe graphics interchange format”).
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GM16CoderInfoMBS in coders
names.Append coder.name+" "+coder.description
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.2.6 isMultiFrame as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format supports multiple frames.
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GM16CoderInfoMBS in coders
names.Append coder.name+" "+str(coder.isMultiFrame)
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.2.7 isReadable as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format is readable.
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GM16CoderInfoMBS in coders
names.Append coder.name+" "+str(coder.isReadable)
next
```

```
MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.2.8 isWritable as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format is writeable.
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GM16CoderInfoMBS in coders
names.Append coder.name+" "+str(coder.isWritable)
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.2.9 ModuleName as String

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Name of loadable module.
Notes: (Read and Write property)

4.2.10 name as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format name (e.g. "GIF").
Example:

```
dim coders(-1) as GM16CoderInfoMBS = GM16CoderInfoMBS.CoderInfoList
dim coder as GM16CoderInfoMBS = coders(0) // pick first one

MsgBox coder.name
```

Notes: (Read and Write property)

4.2.11 Note as String

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Usage note for user.
Notes: (Read and Write property)

4.2.12 Version as String

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Support library version.
Notes: (Read and Write property)

4.3 class GM16ColorGrayMBS

4.3.1 class GM16ColorGrayMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color subclass for a grayscale color.
Example:

```
dim g as new GM16ColorGrayMBS(0.5)
MsgBox str(g.shade)
```

Notes: Representation of grayscale RGB color.
Equal parts red, green, and blue specified as a ratio (0 to 1).
Subclass of the GM16ColorMBS class.

4.3.2 Methods

4.3.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GM16ColorGrayMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.3.4 Constructor(other as GM16ColorMBS) 96
- 4.3.5 Constructor(shade as Double) 97

4.3.4 Constructor(other as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GM16ColorGrayMBS(0.5)
dim o as new GM16ColorGrayMBS(g)
```

```
MsgBox str(o.shade)
```

See also:

- 4.3.3 Constructor 96
- 4.3.5 Constructor(shade as Double) 97

4.3.5 Constructor(shade as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given value.
Example:

```
dim g as new GM16ColorGrayMBS(1.0)
MsgBox str(g.colorValue)
```

Notes: Range is 0.0 to 1.0.
See also:

- 4.3.3 Constructor 96
- 4.3.4 Constructor(other as GM16ColorMBS) 96

4.3.6 Properties

4.3.7 shade as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The gray value for this color.
Example:

```
dim g as new GM16ColorGrayMBS(1.0)
MsgBox str(g.shade)
```

Notes: Range is 0.0 to 1.0
(Read and Write property)

4.4 class GM16ColorHSLMBS

4.4.1 class GM16ColorHSLMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for a HSL color.
Example:

```
dim g as new GM16ColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.colorValue)
```

Notes: Subclass of the GM16ColorMBS class.

4.4.2 Methods

4.4.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GM16ColorHSLMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.4.4 Constructor(hue as Double, saturation as Double, luminosity as Double) 98
- 4.4.5 Constructor(other as GM16ColorMBS) 99

4.4.4 Constructor(hue as Double, saturation as Double, luminosity as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim g as new GM16ColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.hue)+" "+str(g.saturation)+" "+str(g.luminosity)
```

See also:

- 4.4.3 Constructor 98
- 4.4.5 Constructor(other as GM16ColorMBS) 99

4.4.5 Constructor(other as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GM16ColorHSLMBS(0.1,0.2,0.3)
dim o as new GM16ColorHSLMBS(g)

MsgBox str(o.colorValue)
```

See also:

- 4.4.3 Constructor 98
- 4.4.4 Constructor(hue as Double, saturation as Double, luminosity as Double) 98

4.4.6 Properties

4.4.7 hue as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The hue value.
Example:

```
dim g as new GM16ColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.hue)
```

Notes: (Read and Write property)

4.4.8 luminosity as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The luminosity value.
Example:

```
dim g as new GM16ColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.luminosity)
```

Notes: (Read and Write property)

4.4.9 saturation as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The saturation value.
Example:

```
dim g as new GM16ColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.saturation)
```

Notes: (Read and Write property)

4.5 class GM16ColorMBS

4.5.1 class GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color is the base color class.
Example:

```
dim c as new GM16ColorMBS(127,255,127) // light green
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

Notes: It is a simple container class for the pixel red, green, blue, and alpha values scaled to fit GraphicsMagick's Quantum size. Normally users will instantiate a class derived from Color which supports the color model that fits the needs of the application. The Color class may be constructed directly from an X11-style color string. As a perhaps odd design decision, the value transparent black is considered to represent an unset value (invalid color) in many cases. This choice was made since it avoided using more memory. The default Color constructor constructs an invalid color (i.e. transparent black) and may be used as a parameter in order to remove a color setting.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 16.0pr7

4.5.2 Methods

4.5.3 Black as GM16ColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries a black color.
Example:

```
dim black as GM16ColorMBS = GM16ColorMBS.Black

MsgBox str(black.colorValue)
```

4.5.4 Color(ColorValue as Color) as GM16ColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts color from Xojo to GM16ColorMBS.
Example:

```
dim c as GM16ColorMBS = GM16ColorMBS.Color(&cFF0000)

MsgBox str(c.colorValue)
```

See also:

- 4.5.5 Color(ColorValue as Color, alpha as Integer) as GM16ColorMBS 102
- 4.5.6 Color(red as integer, green as integer, blue as integer) as GM16ColorMBS 102
- 4.5.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GM16ColorMBS 103

4.5.5 Color(ColorValue as Color, alpha as Integer) as GM16ColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts color from Xojo to GM16ColorMBS with separate alpha value.
Example:

```
dim c as GM16ColorMBS = GM16ColorMBS.Color(&cFF0000, 128)

MsgBox str(c.colorValue)+" "+str(c.alpha)
```

Notes: Alpha in range from 0 to 255.
See also:

- 4.5.4 Color(ColorValue as Color) as GM16ColorMBS 101
- 4.5.6 Color(red as integer, green as integer, blue as integer) as GM16ColorMBS 102
- 4.5.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GM16ColorMBS 103

4.5.6 Color(red as integer, green as integer, blue as integer) as GM16ColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates color with RGB values.
Example:

```
dim c as GM16ColorMBS = GM16ColorMBS.Color(127, 191, 255)

MsgBox str(c.colorValue)
```

Notes: Range in 0 to 255 for 8bit and 0 to 65535 for 16bit class.
See also:

- 4.5.4 Color(ColorValue as Color) as GM16ColorMBS 101
- 4.5.5 Color(ColorValue as Color, alpha as Integer) as GM16ColorMBS 102
- 4.5.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GM16ColorMBS 103

4.5.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GM16ColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates color with RGB values.
Example:

```
dim c as GM16ColorMBS = GM16ColorMBS.Color(127, 191, 255, 127)

MsgBox str(c.colorValue)+" "+str(c.alphaQuantum)
```

Notes: Range in 0 to 255 for 8bit and 0 to 65535 for 16bit class.
See also:

- 4.5.4 Color(ColorValue as Color) as GM16ColorMBS 101
- 4.5.5 Color(ColorValue as Color, alpha as Integer) as GM16ColorMBS 102
- 4.5.6 Color(red as integer, green as integer, blue as integer) as GM16ColorMBS 102

4.5.8 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GM16ColorMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.5.9 Constructor(ColorName as string) 104
- 4.5.10 Constructor(ColorValue as color) 104
- 4.5.11 Constructor(ColorValue as color, alpha as Integer) 105
- 4.5.12 Constructor(other as GM16ColorMBS) 105

- 4.5.13 Constructor(red as Integer, green as Integer, blue as Integer) 106
- 4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 106

4.5.9 Constructor(ColorName as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color based on the X11 color name.
Example:

```
dim c as new GM16ColorMBS(”red”)

MsgBox str(c.redQuantum)+”-”+str(c.greenQuantum)+”-”+str(c.blueQuantum) // shows ”255-0-0”

dim d as new GM16ColorMBS(”#77FF00”)

MsgBox str(d.redQuantum)+”-”+str(d.greenQuantum)+”-”+str(d.blueQuantum) // shows ”119-255-0”
```

Notes: An alternate way to construct the class is via an X11-compatible color specification string (e.g. Color(”red”) or Color (”#FF0000”)). Since the class may be constructed from a string, convenient strings may be passed in place of an explicit Color object in methods which accept a reference to Color. Color may also be converted to a std::string for convenience in user interfaces, and for saving settings to a text file.
See also:

- 4.5.8 Constructor 103
- 4.5.10 Constructor(ColorValue as color) 104
- 4.5.11 Constructor(ColorValue as color, alpha as Integer) 105
- 4.5.12 Constructor(other as GM16ColorMBS) 105
- 4.5.13 Constructor(red as Integer, green as Integer, blue as Integer) 106
- 4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 106

4.5.10 Constructor(ColorValue as color)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GM16ColorMBS(&cFF0000)
MsgBox str(c.redQuantum)+” ”+str(c.greenQuantum)+” ”+str(c.blueQuantum)
```

See also:

- 4.5.8 Constructor 103
- 4.5.9 Constructor(ColorName as string) 104
- 4.5.11 Constructor(ColorValue as color, alpha as Integer) 105
- 4.5.12 Constructor(other as GM16ColorMBS) 105
- 4.5.13 Constructor(red as Integer, green as Integer, blue as Integer) 106
- 4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 106

4.5.11 Constructor(ColorValue as color, alpha as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GM16ColorMBS(&cFF0102, 127)
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)+" "+str(c.alpha)
```

See also:

- 4.5.8 Constructor 103
- 4.5.9 Constructor(ColorName as string) 104
- 4.5.10 Constructor(ColorValue as color) 104
- 4.5.12 Constructor(other as GM16ColorMBS) 105
- 4.5.13 Constructor(red as Integer, green as Integer, blue as Integer) 106
- 4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 106

4.5.12 Constructor(other as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim r as new GM16ColorMBS(1,2,3)
dim c as new GM16ColorMBS(r)
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.5.8 Constructor 103
- 4.5.9 Constructor(ColorName as string) 104
- 4.5.10 Constructor(ColorValue as color) 104
- 4.5.11 Constructor(ColorValue as color, alpha as Integer) 105
- 4.5.13 Constructor(red as Integer, green as Integer, blue as Integer) 106
- 4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 106

4.5.13 Constructor(red as Integer, green as Integer, blue as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GM16ColorMBS(1,2,3)
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
See also:

- 4.5.8 Constructor 103
- 4.5.9 Constructor(ColorName as string) 104
- 4.5.10 Constructor(ColorValue as color) 104
- 4.5.11 Constructor(ColorValue as color, alpha as Integer) 105
- 4.5.12 Constructor(other as GM16ColorMBS) 105
- 4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 106

4.5.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GM16ColorMBS(1,2,3,4)

// display color, alpha is double...
MsgBox str(C.redQuantum)+” ”+str(c.greenQuantum)+” ”+str(c.blueQuantum)+” ”+str(c.alpha)
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
See also:

- 4.5.8 Constructor 103
- 4.5.9 Constructor(ColorName as string) 104
- 4.5.10 Constructor(ColorValue as color) 104
- 4.5.11 Constructor(ColorValue as color, alpha as Integer) 105
- 4.5.12 Constructor(other as GM16ColorMBS) 105
- 4.5.13 Constructor(red as Integer, green as Integer, blue as Integer) 106

4.5.15 QuantumByteSize as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum byte size.
Example:

```
MsgBox str(GM16ColorMBS.QuantumByteSize)
```

Notes: As the plugin uses 8 bit this value should be 1.

4.5.16 scaleDoubleToQuantum(value as Double) as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales a double value to a value in the range of a quantum.
Example:

```
dim d as Double = 1.0
dim v as Integer = GM16ColorMBS.scaleDoubleToQuantum(d)
MsgBox str(v)
```

Notes: As the plugin uses 8 bit quantums, this is basicly a multiplication by 255.0

4.5.17 scaleQuantumToDouble(value as Integer) as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales a quantum to a double value.
Example:

```
dim v as Integer = 255
dim d as Double = GM16ColorMBS.scaleQuantumToDouble(v)
MsgBox str(d)
```

Notes: The plugin uses 8bit quantums, so this is basicly the division of value by 255.0

4.5.18 White as GM16ColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries a white color.
Example:

```
dim White as GM16ColorMBS = GM16ColorMBS.White

MsgBox str(White.colorValue)
```

4.5.19 Properties

4.5.20 alpha as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha value of this color.
Example:

```
dim c as new GM16ColorMBS(1,2,3,1.0)
MsgBox str(c.alpha)
```

Notes: Range is 0.0 to 1.0. If you pass values higher, they are divided by 255.
(Read and Write property)

4.5.21 alphaQuantum as Integer

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha color value.
Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.5.22 blueQuantum as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color value.
Example:

```
dim c as new GM16ColorMBS(1,2,3)
MsgBox str(c.redQuantum) // 3
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.5.23 colorValue as color

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Xojo color for the GraphicsMagick color.
Example:

```
dim c as new GM16ColorMBS(&cFF0102)
MsgBox str(c.ColorValue)
```

Notes: (Read and Write property)

4.5.24 greenQuantum as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color value.
Example:

```
dim r as new GM16ColorMBS(1,2,3)
MsgBox str(r.greenQuantum) // shows 2
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.5.25 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal color reference.
Example:

```
dim r as new GM16ColorMBS(1,2,3)
MsgBox str(r.handle)
```

Notes: (Read and Write property)

4.5.26 intensity as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The intensity of this color.
Example:

```
dim c as new GM16ColorMBS(1,2,3)
MsgBox str(c.intensity)
```

Notes: (Read only property)

4.5.27 isValid as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Does object contain valid color?
Example:

```
dim c as new GM16ColorMBS(1,2,3)
MsgBox str(c.isValid)
```

Notes: (Read and Write property)

4.5.28 redQuantum as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color value.
Example:

```
dim c as new GM16ColorMBS(1,2,3)
MsgBox str(c.redQuantum) // 1
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.6 class GM16ColorMonoMBS

4.6.1 class GM16ColorMonoMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Representation of a black/white color (true/false)
Example:

```
dim g as new GM16ColorMonoMBS(false)
MsgBox str(g.colorValue)
```

Notes: Subclass of the GM16ColorMBS class.

4.6.2 Methods

4.6.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GM16ColorMonoMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.6.4 Constructor(mono as boolean) 112
- 4.6.5 Constructor(other as GM16ColorMBS) 113

4.6.4 Constructor(mono as boolean)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim g as new GM16ColorMonoMBS(false)
MsgBox str(g.mono)
```

See also:

- 4.6.3 Constructor 112
- 4.6.5 Constructor(other as GM16ColorMBS) 113

4.6.5 Constructor(other as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GM16ColorMonoMBS(false)
dim o as new GM16ColorMonoMBS(g)
MsgBox str(o.mono)
```

See also:

- 4.6.3 Constructor 112
- 4.6.4 Constructor(mono as boolean) 112

4.6.6 Properties

4.6.7 mono as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color value.
Example:

```
dim g as new GM16ColorMonoMBS(true)
MsgBox str(g.mono)
```

Notes: (Read and Write property)

4.7 class GM16ColorRGBMBS

4.7.1 class GM16ColorRGBMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color class for RGB colors.
Example:

```
dim c as new GM16ColorRGBMBS(1.0,0.0,0.0) // red
MsgBox str(C.red)+" "+str(c.green)+" "+str(c.blue)
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

Notes: Representation of RGB color with red, green, and blue specified as ratios (0 to 1)
Subclass of the GM16ColorMBS class.

4.7.2 Methods

4.7.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GM16ColorRGBMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.7.4 Constructor(other as GM16ColorMBS) 114
- 4.7.5 Constructor(red as Double, green as Double, blue as Double) 115

4.7.4 Constructor(other as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GM16ColorRGBMBS(1,2,3)
dim o as new GM16ColorRGBMBS(g)
```

```
MsgBox str(o.colorValue)
```

See also:

- 4.7.3 Constructor 114
- 4.7.5 Constructor(red as Double, green as Double, blue as Double) 115

4.7.5 Constructor(red as Double, green as Double, blue as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GM16ColorRGBMBS(0.1,0.2,0.3)
```

Notes: Range is 0.0 to 1.0.
See also:

- 4.7.3 Constructor 114
- 4.7.4 Constructor(other as GM16ColorMBS) 114

4.7.6 Properties

4.7.7 blue as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color component.
Example:

```
dim c as new GM16ColorRGBMBS(0.0,0.0,1.0)
MsgBox str(c.blue)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.7.8 green as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color component.
Example:

```
dim c as new GM16ColorRGBMBS(0.0,1.0,0.0)
MsgBox str(c.green)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.7.9 red as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color component.
Example:

```
dim c as new GM16ColorRGBMBS(1.0,0.0,0.0) // red
MsgBox str(C.red)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.8 class GM16ColorYUVMBS

4.8.1 class GM16ColorYUVMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Representation of a color in the YUV colorspace
Example:

```
dim g as new GM16ColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.y)+" "+str(g.u)+" "+str(g.v)
```

Notes: Subclass of the GM16ColorMBS class.

4.8.2 Methods

4.8.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GM16ColorYUVMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.8.4 Constructor(other as GM16ColorMBS) 117
- 4.8.5 Constructor(y as Double, u as Double, v as Double) 118

4.8.4 Constructor(other as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GM16ColorYUVMBS(0.1, 0.2, 0.3)
dim o as new GM16ColorYUVMBS(g)
MsgBox str(o.colorValue)
```

See also:

- 4.8.3 Constructor 117
- 4.8.5 Constructor(y as Double, u as Double, v as Double) 118

4.8.5 Constructor(y as Double, u as Double, v as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim g as new GM16ColorYUVMBS(0.1, 0.2, 0.3)
```

See also:

- 4.8.3 Constructor 117
- 4.8.4 Constructor(other as GM16ColorMBS) 117

4.8.6 Properties

4.8.7 u as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The u color component.
Example:

```
dim g as new GM16ColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.u)
```

Notes: Range is -0.5 to +0.5.
(Read and Write property)

4.8.8 v as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The v color component.
Example:

```
dim g as new GM16ColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.v)
```

Notes: Range is -0.5 to +0.5.
(Read and Write property)

4.8.9 y as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y color component.
Example:

```
dim g as new GM16ColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.y)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.9 class GM16ConvertMBS

4.9.1 class GM16ConvertMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class to convert images thread friendly.
Notes: This class is intended to process lots of images in several instances in several threads.
If you for example use 8 Xojo threads, to process thousands of images to scale them down for thumbnails, you can easily keep 8 CPU cores busy.

Please make a new instance, set options and call run method. When run is done, please read output properties.
Do not modify properties while thread is running.
Blog Entries

- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.1
- MBS Xojo Plugins, version 19.1pr2

4.9.2 Methods

4.9.3 Constructor

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.9.4 Run

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Starts converter.
Notes: The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.
If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

4.9.5 Properties

4.9.6 AutoOrient as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to call autoOrient method to change orientation of image data to 0¬∞.
Notes: (Read and Write property)

4.9.7 Enhance as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to run enhance command.
Notes: If true, we call enhance on the image to minimize noise.
(Read and Write property)

4.9.8 Equalize as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to run equalize command.
Notes: If set to true, we call equalize command on image (histogram equalization).
(Read and Write property)

4.9.9 ImageType as Integer

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the image type.
Notes: If value is >0, the image type is change to the given type.
(Read and Write property)

4.9.10 InputData as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input data string.
Notes: If set, we read image from this data.
(Read and Write property)

4.9.11 InputFile as FolderItem

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input folderitem.
Notes: If set input file to read.
(Read and Write property)

4.9.12 InputGeometry as GM16GeometryMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input geometry.
Notes: Some formats can be loaded with different scales, so this geometry is passed to read method to define the format requested.
(Read and Write property)

4.9.13 InputImage as GM16ImageMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input image.
Notes: (Read and Write property)

4.9.14 InputMagick as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image magick type.
Notes: If empty, the type of file is automatically determined.
(Read and Write property)

4.9.15 InputMemory as MemoryBlock

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input data memoryblock.
Notes: If set, we read image from this data.
(Read and Write property)

4.9.16 InputPath as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input file path.
Notes: If set input file to read.
(Read and Write property)

4.9.17 OutputData as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output data as string.
Notes: (Read and Write property)

4.9.18 OutputFile as FolderItem

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The folderitem where to write file.
Notes: If set, the image will be written to this file.
(Read and Write property)

4.9.19 OutputImage as GM16ImageMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output image object.
Notes: (Read and Write property)

4.9.20 OutputMagick as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output magick.
Notes: You can set this to a magick codec type to define output format, e.g. ”jpeg”.
(Read and Write property)

4.9.21 OutputMemory as MemoryBlock

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output data as memory block.
Notes: (Read and Write property)

4.9.22 OutputPath as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The native file path for output.
Notes: If set, the image will be written to this path.
(Read and Write property)

4.9.23 Quality as Integer

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Quality of images in range 1 to 100.
Notes: If value is >0, we assign it to the image for setting image quality.
(Read and Write property)

4.9.24 Running as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether this converter is running.
Notes: Set to true while Run method runs.
(Read only property)

4.9.25 ScaleGeometry as GM16GeometryMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The geometry for call to scale method.
Notes: scale method is called with this geometry (if set) to reduce image size.
(Read and Write property)

4.9.26 Strip as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to remove all profiles and text attributes from the image.
Notes: (Read and Write property)

4.9.27 ThumbnailGeometry as GM16GeometryMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The geometry for call to thumbnail method.
Notes: thumbnail method is called with this geometry (if set) to reduce image size.
(Read and Write property)

4.9.28 Trim as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether trim command is called on image to remove extra blank space around image.
Notes: (Read and Write property)

4.9.29 WantOutputData as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether you want image compressed and stored in OutputData property.
Notes: (Read and Write property)

4.9.30 WantOutputMemory as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether you want image compressed and stored in OutputMemory property.
Notes: (Read and Write property)

4.10 class GM16CoordinateMBS

4.10.1 class GM16CoordinateMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Graphics Magick class for a coordinate.
Example:

```
dim c as new GM16CoordinateMBS(5,6)
MsgBox str(c.x)+" "+str(c.y)
```

4.10.2 Methods

4.10.3 Constructor

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create a new coordinate.
See also:

- 4.10.4 Constructor(x as Double, y as Double) 126

4.10.4 Constructor(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create a new coordinate.
Example:

```
dim c as new GM16CoordinateMBS(5,6)
MsgBox str(c.x)+" "+str(c.y)
```

See also:

- 4.10.3 Constructor 126

4.10.5 Properties

4.10.6 x as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x value.
Example:

```
dim c as new GM16CoordinateMBS
c.x = 5
MsgBox str(c.x)
```

Notes: (Read and Write property)

4.10.7 y as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y value.
Example:

```
dim c as new GM16CoordinateMBS
c.y = 5
MsgBox str(c.y)
```

Notes: (Read and Write property)

4.11 class GM16ErrorExceptionMBS

4.11.1 class GM16ErrorExceptionMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception to report errors in the GraphicMagick plugin.
Notes: Check the message property for details.
Subclass of the RuntimeException class.

4.12 class GM16GeometryMBS

4.12.1 class GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Geometry provides a convenient means to specify a geometry argument.
Example:

```
dim g as new GM16GeometryMBS(300,400)
MsgBox str(G.width)+" "+str(G.height)
```

Notes: The object may be initialized from a string containing a geometry specification. It may also be initialized by more efficient parameterized constructors.
Xojo Developer Magazine

- 20.2, page 83: Wifi QR Code, Embedding your Wifi password in a QR code by Stefanie Juchmes

4.12.2 Methods

4.12.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates empty geometry.
Example:

```
dim g as new GM16GeometryMBS
MsgBox str(G.width)+" "+str(G.height)
```

See also:

- 4.12.4 Constructor(geometry as string) 129
- 4.12.5 Constructor(other as GM16GeometryMBS) 130
- 4.12.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) 130

4.12.4 Constructor(geometry as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct geometry from string.
Example:

```
dim g as new GM16GeometryMBS("600x600")

MsgBox str(G.width)+" "+str(G.height)
```

Notes: See the GraphicsMagick website for details.
http://www.graphicsmagick.org/Magick++/Geometry.html
See also:

- 4.12.3 Constructor 129
- 4.12.5 Constructor(other as GM16GeometryMBS) 130
- 4.12.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) 130

4.12.5 Constructor(other as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new geometry object by copying an existing one.
Example:

```
dim g as new GM16GeometryMBS(600,600)
dim h as new GM16GeometryMBS(g)
MsgBox str(h.width)
```

See also:

- 4.12.3 Constructor 129
- 4.12.4 Constructor(geometry as string) 129
- 4.12.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) 130

4.12.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates geometry with the given values.
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.width)
```

See also:

- 4.12.3 Constructor 129
- 4.12.4 Constructor(geometry as string) 129
- 4.12.5 Constructor(other as GM16GeometryMBS) 130

4.12.7 Make(geometry as string) as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct geometry from string.
Example:

```
dim g as GM16GeometryMBS = GM16GeometryMBS.Make("600x600")

MsgBox str(G.width)+" "+str(G.height)
```

Notes: See the GraphicsMagick website for more details:
http://www.graphicsmagick.org/Magick++/Geometry.html
See also:

- 4.12.8 Make(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) as GM16GeometryMBS 131

4.12.8 Make(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates geometry with the given values.
Example:

```
dim g as GM16GeometryMBS = GM16GeometryMBS.Make(600,600)
MsgBox str(g.width)
```

See also:

4.12.9 Properties

4.12.10 aspect as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize without preserving aspect ratio (!).
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.aspect)
```

Notes: (Read and Write property)

4.12.11 fillArea as Boolean

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image to fit total pixel area specified by dimensions.
Notes: Same as @ in the geometry specification.
(Read and Write property)

4.12.12 greater as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize if image is greater than size (>).
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.greater)
```

Notes: (Read and Write property)

4.12.13 height as Uint32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height value.
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.height)
```

Notes: (Read and Write property)

4.12.14 isValid as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Does object contain a valid geometry?
Example:

```
dim g as new GM16GeometryMBS(100,200)
MsgBox str(G.isValid)
```

Notes: May be set to false in order to invalidate an existing geometry object.
(Read and Write property)

4.12.15 less as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize if image is less than size (<).
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.less)
```

Notes: (Read and Write property)

4.12.16 limitPixels as Boolean

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Dimensions are treated as minimum rather than maximum values.
Notes: Same as ^in the geometry specification.

(Read and Write property)

4.12.17 percent as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height are expressed as percentages.
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.percent)
```

Notes: (Read and Write property)

4.12.18 StringValue as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The string representation of the geometry object.
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.StringValue)
```

Notes: (Read and Write property)

4.12.19 width as Uint32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width value.
Example:

```
dim g as new GM16GeometryMBS(600,600)
MsgBox str(g.width)
```

Notes: (Read and Write property)

4.12.20 xNegative as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sign of X offset negative? (X origin at right)
Example:

```
dim g as new GM16GeometryMBS(100,200,30,40,true,false)
MsgBox str(G.xNegative)
```

Notes: (Read and Write property)

4.12.21 xOff as Uint32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: X offset from origin.
Example:

```
dim g as new GM16GeometryMBS(100,200,30,40,true,true)
MsgBox str(G.xOff)+" "+str(G.yOff)
```

Notes: (Read and Write property)

4.12.22 yNegative as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sign of Y offset negative? (Y origin at bottom)
Example:

```
dim g as new GM16GeometryMBS(100,200,30,40,false,true)
MsgBox str(G.yNegative)
```

Notes: (Read and Write property)

4.12.23 yOff as Uint32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Y offset from origin
Example:

```
dim g as new GM16GeometryMBS(100,200,30,40,true,true)
MsgBox str(G.xOff)+" "+str(G.yOff)
```

Notes: (Read and Write property)

4.13 class GM16GraphicsMBS

4.13.1 class GM16GraphicsMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for drawing commands targeting a GM16ImageMBS.
Notes: Please remember that all commands are collected till you call the Draw method.

4.13.2 Methods

4.13.3 Arc(startX as Double, startY as Double, endX as Double, endY as Double, startDegrees as Double, endDegrees as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an arc using the stroke color and based on the circle starting at coordinates startX,startY, and ending with coordinates endX,endY, and bounded by the rotational arc startDegrees,endDegrees.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.type = image.TrueColorType
image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.arc(250, 250, 100, 100,50,300)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.4 Bezier(values() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a bezier curve using the stroke color and based on the coordinates specified by the coordinates array.

4.13.5 Circle(originX as Double, originY as Double, perimX as Double, perimY as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a circle using the stroke color and thickness using specified origin and perimeter coordinates.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.type = image.TrueColorType
image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.13.6 ClipPath(id as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Select a drawing clip path matching id.

4.13.7 ColorPixel(x as Double, y as Double, paintMethod as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color image according to paintMethod.
Notes: The point method recolors the target pixel. The replace method recolors any pixel that matches the color of the target pixel. Floodfill recolors any pixel that matches the color of the target pixel and is a neighbor, whereas filltoborder recolors any neighbor pixel that is not the border color. Finally, reset recolors all pixels.

4.13.8 CompositeImage(x as Double, y as Double, file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through.
See also:

- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142

- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.10 CompositeImage(x as Double, y as Double, path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, using specified composition algorithm, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, using specified composition algorithm, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140

- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 144

4.13.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, using specified composition algorithm, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.13.8 CompositeImage(x as Double, y as Double, file as folderitem) 139
- 4.13.9 CompositeImage(x as Double, y as Double, image as GM16ImageMBS) 139
- 4.13.10 CompositeImage(x as Double, y as Double, path as string) 140
- 4.13.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 140
- 4.13.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 141
- 4.13.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS) 142
- 4.13.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GM16ImageMBS, CompositeOperator as Integer) 142
- 4.13.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 143

4.13.17 Constructor(image as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new object referencing the given image.

4.13.18 DashArray(values() as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the pattern of dashes and gaps used to stroke paths.
Notes: The strokeDashArray represents a zero-terminated array of numbers that specify the lengths of alternating dashes and gaps in pixels. If an odd number of values is provided, then the list of values is repeated to yield an even number of values. A typical strokeDashArray array might contain the members 5 3 2 0, where the zero value indicates the end of the pattern array.

4.13.19 DashOffset(offset as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the distance into the dash pattern to start the dash.
Notes: See documentation on SVG's stroke-dashoffset property for usage details.

4.13.20 Draw

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws all draw commands collected.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.type = image.TrueColorType
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.StrokeColor new GM16ColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.21 DrawPath

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw on image using vector path.
Example:

```
// new picture, 500x500 and filled with white
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)


dim draw as GM16GraphicsMBS = image.Graphics

// Draw path
```

```
dim cr as new GM16ColorRGBMBS("red")
dim gr as new GM16ColorRGBMBS("green")
draw.StrokeColor cr
draw.FillColor gr
draw.PathMovetoAbs(30,10)
draw.PathLinetoAbs(20,55)
draw.PathLinetoAbs(70,50)
draw.PathLinetoAbs(80,5)
draw.DrawPath

draw.Draw

// show picture
image.type = image.TrueColorType // make sure it's a bitmap
Backdrop=image.CopyPicture
```

4.13.22 Ellipse(originX as Double, originY as Double, perimX as Double, perimY as Double, arcStart as Double, arcEnd as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an ellipse using the stroke color and thickness, specified origin, x & y radius, as well as specified start and end of arc in degrees.
Notes: If a fill color is specified, then the object is filled.

4.13.23 FillColor(c as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify drawing object fill color.

4.13.24 FillOpacity(opacity as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify opacity to use when drawing using fill color.

4.13.25 FillRule(fillRule as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the algorithm which is to be used to determine what parts of the canvas are included inside the shape.
Notes: See documentation on SVG's fill-rule property for usage details.

4.13.26 Font(fontname as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify font name to use when drawing text.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)

dim draw as GM16GraphicsMBS = image.Graphics

// draw red text
draw.strokeColor(new GM16ColorRGBMBS(”red”)) // Outline color
draw.strokeWidth(1)
draw.Font(”/Library/Fonts/Verdana.ttf”)
draw.Text(50, 50, ”Hello”, ””)
draw.Draw

Backdrop=image.CopyPicture
```

See also:

- 4.13.27 Font(fontname as string, StyleType as Integer, weight as Integer, StretchType as Integer) 147

4.13.27 Font(fontname as string, StyleType as Integer, weight as Integer, StretchType as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the font.
Notes: Specify font family, style, weight (one of the set { 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 } with 400 being the normal size), and stretch to be used to select the font used when drawing text. Wildcard matches may be applied to style via the AnyStyle enumeration, applied to weight if weight is zero, and applied to stretch via the AnyStretch enumeration.
See also:

4.13.28 Gravity(GravityType as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify text positioning gravity.

4.13.29 Line(startX as Double, startY as Double, endX as Double, endY as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a line using stroke color and thickness using starting and ending coordinates
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a line
draw.Line(100,100,400,400)
draw.Draw

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

4.13.30 Matte(x as Double, y as Double, paintMethod as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Change the pixel matte value to transparent.
Notes: The point method changes the matte value of the target pixel. The replace method changes the matte value of any pixel that matches the color of the target pixel. Floodfill changes the matte value of any pixel that matches the color of the target pixel and is a neighbor, whereas filltoborder changes the matte value of any neighbor pixel that is not the border color, Finally reset changes the matte value of all pixels.

4.13.31 MiterLimit(miterlimit as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify miter limit.
Notes: When two line segments meet at a sharp angle and miter joins have been specified for 'lineJoin', it is possible for the miter to extend far beyond the thickness of the line stroking the path. The miterLimit' imposes a limit on the ratio of the miter length to the 'lineWidth'. The default value of this parameter is 4.

4.13.32 PathArcAbs(c as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GM16PathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.13.33 PathArcAbs(c() as GM16PathArgsMBS) 149
- 4.13.34 PathArcAbs(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 150

4.13.33 PathArcAbs(c() as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GM16PathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.13.32 PathArcAbs(c as GM16PathArgsMBS) 149

- 4.13.34 PathArcAbs(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 150

4.13.34 PathArcAbs(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS(”red”) // Outline color
image.fillColor = new GM16ColorRGBMBS(”green”) // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw an arc

draw.PathMovetoAbs 100,100
draw.PathArcAbs(100,100, 0, false, false, 200,200)
draw.DrawPath
draw.Draw

Backdrop=image.CopyPicture
```

Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.
See also:

- 4.13.32 PathArcAbs(c as GM16PathArgsMBS) 149

- 4.13.33 PathArcAbs(c() as GM16PathArgsMBS) 149

4.13.35 PathArcRel(c as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GM16PathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.13.36 PathArcRel(c() as GM16PathArgsMBS) 151
- 4.13.37 PathArcRel(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 152

4.13.36 PathArcRel(c() as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GM16PathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.13.35 PathArcRel(c as GM16PathArgsMBS) 151
- 4.13.37 PathArcRel(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 152

4.13.37 PathArcRel(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.
See also:

- 4.13.35 PathArcRel(c as GM16PathArgsMBS) 151
- 4.13.36 PathArcRel(c() as GM16PathArgsMBS) 151

4.13.38 PathClosePath

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Close the current subpath by drawing a straight line from the current point to current subpath's most recent starting point (usually, the most recent moveto point).

4.13.39 PathCurvetoAbs(c as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.40 PathCurvetoAbs(c() as GM16PathArgsMBS) 153
- 4.13.41 PathCurvetoAbs(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 153

4.13.40 PathCurvetoAbs(c() as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.39 PathCurvetoAbs(c as GM16PathArgsMBS) 152
- 4.13.41 PathCurvetoAbs(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 153

4.13.41 PathCurvetoAbs(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.39 PathCurvetoAbs(c as GM16PathArgsMBS) 152
- 4.13.40 PathCurvetoAbs(c() as GM16PathArgsMBS) 153

4.13.42 PathCurvetoRel(c as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.43 PathCurvetoRel(c() as GM16PathArgsMBS) 154
- 4.13.44 PathCurvetoRel(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 154

4.13.43 PathCurvetoRel(c() as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.42 PathCurvetoRel(c as GM16PathArgsMBS) 153
- 4.13.44 PathCurvetoRel(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 154

4.13.44 PathCurvetoRel(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.42 PathCurvetoRel(c as GM16PathArgsMBS) 153
- 4.13.43 PathCurvetoRel(c() as GM16PathArgsMBS) 154

4.13.45 PathLinetoAbs(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various ”lineto” commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point.

PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.13.46 PathLinetoAbs(c() as GM16CoordinateMBS) 155
- 4.13.47 PathLinetoAbs(x as Double, y as Double) 155

4.13.46 PathLinetoAbs(c() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.13.45 PathLinetoAbs(c as GM16CoordinateMBS) 154
- 4.13.47 PathLinetoAbs(x as Double, y as Double) 155

4.13.47 PathLinetoAbs(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.13.45 PathLinetoAbs(c as GM16CoordinateMBS) 154
- 4.13.46 PathLinetoAbs(c() as GM16CoordinateMBS) 155

4.13.48 PathLinetoHorizontalAbs(v as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draws a horizontal line from the current point (cpx, cpy) to (x, cpy). PathLinetoHorizontalAbs indicates that absolute coordinates are supplied; PathLinetoHorizontalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (x, cpy) for the final value of x.

4.13.49 PathLinetoHorizontalRel(v as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draws a horizontal line from the current point (cpx, cpy) to (x, cpy). PathLinetoHorizontalAbs indicates that absolute coordinates are supplied; PathLinetoHorizontalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (x, cpy) for the final value of x.

4.13.50 PathLinetoRel(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.13.51 PathLinetoRel(c() as GM16CoordinateMBS) 156
- 4.13.52 PathLinetoRel(x as Double, y as Double) 156

4.13.51 PathLinetoRel(c() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.13.50 PathLinetoRel(c as GM16CoordinateMBS) 156
- 4.13.52 PathLinetoRel(x as Double, y as Double) 156

4.13.52 PathLinetoRel(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Example:

```
// new picture, 500x500 and filled with white
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)


dim draw as GM16GraphicsMBS = image.Graphics

// Draw path

dim cr as new GM16ColorRGBMBS("red")
dim gr as new GM16ColorRGBMBS("green")
draw.StrokeColor cr
draw.FillColor gr
draw.PathMovetoAbs(30,10)
draw.PathLinetoAbs(20,55)
draw.PathLinetoAbs(70,50)
draw.PathLinetoAbs(80,5)
draw.DrawPath

draw.Draw

// show picture
image.type = image.TrueColorType // make sure it's a bitmap
Backdrop=image.CopyPicture
```

Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.13.50 PathLinetoRel(c as GM16CoordinateMBS) 156
- 4.13.51 PathLinetoRel(c() as GM16CoordinateMBS) 156

4.13.53 PathLinetoVerticalAbs(v as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draws a vertical line from the current point (cpx, cpy) to (cpx, y). PathLinetoVerticalAbs indicates that absolute coordinates are supplied; PathLinetoVerticalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (cpx, y) for the final value of y.

4.13.54 PathLinetoVerticalRel(v as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various ”lineto” commands draw straight lines from the current point to a new point.
Notes: Draws a vertical line from the current point (cpx, cpy) to (cpx, y). PathLinetoVerticalAbs indicates that absolute coordinates are supplied; PathLinetoVerticalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (cpx, y) for the final value of y.

4.13.55 PathMovetoAbs(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.13.56 PathMovetoAbs(x as Double, y as Double) 158

4.13.56 PathMovetoAbs(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.13.55 PathMovetoAbs(c as GM16CoordinateMBS) 158

4.13.57 PathMovetoRel(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.13.58 PathMovetoRel(x as Double, y as Double) 159

4.13.58 PathMovetoRel(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.13.57 PathMovetoRel(c as GM16CoordinateMBS) 159

4.13.59 PathQuadraticCurvetoAbs(c as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.60 PathQuadraticCurvetoAbs(c() as GM16PathArgsMBS) 160
- 4.13.61 PathQuadraticCurvetoAbs(x1 as Double, y1 as Double, x as Double, y as Double) 160

4.13.60 PathQuadraticCurvetoAbs(c() as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.59 PathQuadraticCurvetoAbs(c as GM16PathArgsMBS) 159
- 4.13.61 PathQuadraticCurvetoAbs(x1 as Double, y1 as Double, x as Double, y as Double) 160

4.13.61 PathQuadraticCurvetoAbs(x1 as Double, y1 as Double, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.59 PathQuadraticCurvetoAbs(c as GM16PathArgsMBS) 159
- 4.13.60 PathQuadraticCurvetoAbs(c() as GM16PathArgsMBS) 160

4.13.62 PathQuadraticCurvetoRel(c as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.63 PathQuadraticCurvetoRel(c() as GM16PathArgsMBS) 161
- 4.13.64 PathQuadraticCurvetoRel(x1 as Double, y1 as Double, x as Double, y as Double) 161

4.13.63 PathQuadraticCurvetoRel(c() as GM16PathArgsMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.62 PathQuadraticCurvetoRel(c as GM16PathArgsMBS) 160
- 4.13.64 PathQuadraticCurvetoRel(x1 as Double, y1 as Double, x as Double, y as Double) 161

4.13.64 PathQuadraticCurvetoRel(x1 as Double, y1 as Double, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.62 PathQuadraticCurvetoRel(c as GM16PathArgsMBS) 160

- 4.13.63 PathQuadraticCurvetoRel(c() as GM16PathArgsMBS) 161

4.13.65 PathSmoothCurvetoAbs(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.66 PathSmoothCurvetoAbs(c() as GM16CoordinateMBS) 162
- 4.13.67 PathSmoothCurvetoAbs(x as Double, y as Double) 163

4.13.66 PathSmoothCurvetoAbs(c() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.65 PathSmoothCurvetoAbs(c as GM16CoordinateMBS) 162
- 4.13.67 PathSmoothCurvetoAbs(x as Double, y as Double) 163

4.13.67 PathSmoothCurvetoAbs(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.65 PathSmoothCurvetoAbs(c as GM16CoordinateMBS) 162
- 4.13.66 PathSmoothCurvetoAbs(c() as GM16CoordinateMBS) 162

4.13.68 PathSmoothCurvetoRel(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.69 PathSmoothCurvetoRel(c() as GM16CoordinateMBS) 163
- 4.13.70 PathSmoothCurvetoRel(x as Double, y as Double) 164

4.13.69 PathSmoothCurvetoRel(c() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was

not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.13.68 PathSmoothCurvetoRel(c as GM16CoordinateMBS) 163
- 4.13.70 PathSmoothCurvetoRel(x as Double, y as Double) 164

4.13.70 PathSmoothCurvetoRel(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.68 PathSmoothCurvetoRel(c as GM16CoordinateMBS) 163
- 4.13.69 PathSmoothCurvetoRel(c() as GM16CoordinateMBS) 163

4.13.71 PathSmoothQuadraticCurvetoAbs(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous
command relative to the current point. (If there is no previous command or if the previous command was not
a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.72 PathSmoothQuadraticCurvetoAbs(c() as GM16CoordinateMBS) 165
- 4.13.73 PathSmoothQuadraticCurvetoAbs(x as Double, y as Double) 165

4.13.72 PathSmoothQuadraticCurvetoAbs(c() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.71 PathSmoothQuadraticCurvetoAbs(c as GM16CoordinateMBS) 164
- 4.13.73 PathSmoothQuadraticCurvetoAbs(x as Double, y as Double) 165

4.13.73 PathSmoothQuadraticCurvetoAbs(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.71 PathSmoothQuadraticCurvetoAbs(c as GM16CoordinateMBS) 164
- 4.13.72 PathSmoothQuadraticCurvetoAbs(c() as GM16CoordinateMBS) 165

4.13.74 PathSmoothQuadraticCurvetoRel(c as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous
command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.75 PathSmoothQuadraticCurvetoRel(c() as GM16CoordinateMBS) 166
- 4.13.76 PathSmoothQuadraticCurvetoRel(x as Double, y as Double) 166

4.13.75 PathSmoothQuadraticCurvetoRel(c() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous
command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GM16PathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.13.74 PathSmoothQuadraticCurvetoRel(c as GM16CoordinateMBS) 166
- 4.13.76 PathSmoothQuadraticCurvetoRel(x as Double, y as Double) 166

4.13.76 PathSmoothQuadraticCurvetoRel(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous

command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.13.74 PathSmoothQuadraticCurvetoRel(c as GM16CoordinateMBS) 166
- 4.13.75 PathSmoothQuadraticCurvetoRel(c() as GM16CoordinateMBS) 166

4.13.77 Point(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a point using stroke color and thickness at coordinate.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.fillColor = new GM16ColorRGBMBS("red") // set color

dim draw as GM16GraphicsMBS = image.Graphics

// draw cross with pixels
for x as Integer = 240 to 260
draw.Point(x, 250)
next
for y as Integer = 240 to 260
draw.Point(250,y)
next
draw.Draw

Backdrop=image.CopyPicture
```

4.13.78 PointSize(pointSize as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set font point size.

4.13.79 Polygon(values() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an arbitrary polygon using stroke color and thickness consisting of three or more coordinates contained in an array.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.fillColor = new GM16ColorRGBMBS("red") // set color
image.strokeColor = new GM16ColorRGBMBS("green") // set color

dim draw as GM16GraphicsMBS = image.Graphics
dim coordinates(-1) as GM16CoordinateMBS

coordinates.Append new GM16CoordinateMBS(70,70)
coordinates.Append new GM16CoordinateMBS(100,340)
coordinates.Append new GM16CoordinateMBS(380,200)
coordinates.Append new GM16CoordinateMBS(70,70)

draw.Polygon coordinates
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.13.80 Polyline(values() as GM16CoordinateMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an arbitrary polyline using stroke color and thickness consisting of three or more coordinates contained in an array.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("green") // set color

dim draw as GM16GraphicsMBS = image.Graphics
dim coordinates(-1) as GM16CoordinateMBS
```

```
coordinates.Append new GM16CoordinateMBS(70,70)
coordinates.Append new GM16CoordinateMBS(100,340)
coordinates.Append new GM16CoordinateMBS(380,200)

draw.Polyline coordinates
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.13.81 PopClipPath

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pop (terminate) clip path definition started by PushClipPath.

4.13.82 PopGraphicContext

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pop Graphic Context.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorMBS() // transparent fillcolor
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a Rectangle
draw.PushGraphicContext
draw.Translation(250,250)
draw.Rotation(50)
draw.Rectangle(0, 0, 100, 100) // rotated
draw.PopGraphicContext
draw.Rectangle(0, 0, 100, 100) // not rotated
draw.Draw
```

Backdrop=image.CopyPicture

Notes: Removing the current graphic context from the graphic context stack restores the options to the values they had prior to the preceding PushGraphicContext operation.

4.13.83 PopPattern

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Terminate a pattern definition started via PushPattern.

4.13.84 PushClipPath(id as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Push (create) clip path definition with id.
Notes: Clip patch definition consists of subsequent drawing commands, terminated by PopClipPath.

4.13.85 PushGraphicContext

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Push Graphic Context.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorMBS() // transparent fillcolor
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a Rectangle
draw.PushGraphicContext
draw.Translation(250,250)
draw.Rotation(50)
draw.Rectangle(0, 0, 100, 100) // rotated
draw.PopGraphicContext
draw.Rectangle(0, 0, 100, 100) // not rotated
draw.Draw

Backdrop=image.CopyPicture
```

Notes: When a graphic context is pushed, options set after the context is pushed (such as coordinate transformations, color settings, etc.) are saved to a new graphic context. This allows related options to be saved on a graphic context ”stack” in order to support heirarchical nesting of options. When PopGraphicContext is used to pop the current graphic context, the options in effect during the last PushGraphicContext operation are restored.

4.13.86 PushPattern(id as string, x as Integer, y as Integer, width as Integer, height as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Start a pattern definition with arbitrary pattern name specified by id, pattern offset specified by x and y, and pattern size specified by width and height.
Notes: The pattern is defined within the coordinate system defined by the specified offset and size. Arbitrary drawing objects (including DrawableCompositeImage) may be specified between PushPattern and PopPattern in order to draw the pattern. Normally the pair PushGraphicContext & PopGraphicContext are used to enclose a pattern definition. Pattern definitions are terminated by a PopPattern object.

4.13.87 Rectangle(upperLeftX as Double, upperLeftY as Double, lowerRightX as Double, lowerRightY as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a rectangle using stroke color and thickness from upper-left coordinates to lower-right coordinates.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS(”red”) // Outline color
image.fillColor = new GM16ColorRGBMBS(”green”) // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a rectangle
```

```
draw.Rectangle(250, 250, 100, 100)
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.13.88 Rotation(angle as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set rotation to use when drawing (coordinate transformation).
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.StrokeColor new GM16ColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Rotation 5
draw.StrokeColor new GM16ColorRGBMBS("blue")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.89 RoundRectangle(centerX as Double, centerY as Double, width as Double, height as Double, cornerWidth as Double, cornerHeight as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a rounded rectangle using stroke color and thickness, with specified center coordinate, specified width and height, and specified corner width and height.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)
```

```
image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a round rectangle
draw.RoundRectangle(250, 250, 100, 100,20,20)
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.13.90 Scaling(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply scaling in x and y direction while drawing objects (coordinate transformation).
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics
draw.FillColor new GM16ColorRGBMBS("red")

draw.StrokeColor new GM16ColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.StrokeColor new GM16ColorRGBMBS("blue")
draw.Scaling 1.2,1.1
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.91 SkewX(angle as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply Skew in X direction (coordinate transformation)
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.StrokeColor new GM16ColorRGBMBS(”red”)
draw.Line(100,100,400,400)
draw.SkewX 5
draw.StrokeColor new GM16ColorRGBMBS(”blue”)
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.92 SkewY(angle as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply Skew in Y direction.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.StrokeColor new GM16ColorRGBMBS(”red”)
draw.Line(100,100,400,400)
draw.SkewY 5
draw.StrokeColor new GM16ColorRGBMBS(”blue”)
draw.Line(100,100,400,400)
draw.Draw
```

```
Backdrop=image.CopyPicture
```

4.13.93 StrokeAntialias(flag as boolean)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Antialias while drawing lines or object outlines.

4.13.94 StrokeColor(c as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set color to use when drawing lines or object outlines.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.StrokeColor new GM16ColorRGBMBS(”red”)
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.95 StrokeLineCap(LineCap as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the end of open subpaths when they are stroked.
Notes: Values of LineCap are UndefinedCap, ButtCap, RoundCap, and SquareCap.

4.13.96 StrokeLineJoin(LineJoin as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the corners of paths (or other vector shapes) when they are stroked.
Notes: Values of LineJoin are UndefinedJoin, MiterJoin, RoundJoin, and BevelJoin.

4.13.97 StrokeOpacity(opacity as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Opacity to use when drawing lines or object outlines.

4.13.98 StrokeWidth(opacity as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set width to use when drawing lines or object outlines.

4.13.99 Text(x as Double, y as Double, text as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate image with text using stroke color, font, font pointsize, and box color (text background color), at specified coordinates.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

dim draw as GM16GraphicsMBS = image.Graphics

// draw red text
draw.strokeColor(new GM16ColorRGBMBS("red")) // Outline color
draw.strokeWidth(1)
draw.Font("/Library/Fonts/Verdana.ttf")
draw.Text(50, 50, "Hello")
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If text contains special format characters the image filename, type, width, height, or other image attributes may be incorporated in the text (see label).
See also:

- 4.13.100 Text(x as Double, y as Double, text as string, encoding as string) 177

4.13.100 Text(x as Double, y as Double, text as string, encoding as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate image with text represented with text encoding, using current stroke color, font, font pointsize, and box color (text background color), at specified coordinates.
Notes: If text contains special format characters the image filename, type, width, height, or other image attributes may be incorporated in the text (see label()).

The text encoding specifies the code set to use for text annotations. The only character encoding which may be specified at this time is ”UTF-8” for representing Unicode as a sequence of bytes. Specify an empty string to set text encoding to the system’s default. Successful text annotation using Unicode may require fonts designed to support Unicode.

Seems like you need ghostscript or the DPS library for text handling, so it may no be available for you.
See also:

- 4.13.99 Text(x as Double, y as Double, text as string) 176

4.13.101 TextAntialias(flag as boolean)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Antialias while drawing text (default true).
Notes: The main reason to disable text antialiasing is to avoid adding new colors to the image.

4.13.102 TextDecoration(DecorationType as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify decoration (e.g. UnderlineDecoration) to apply to text.

4.13.103 TextUnderColor(c as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a box under rendered text using the specified color.

4.13.104 Translation(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply coordinate translation (set new coordinate origin).
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

draw.StrokeColor new GM16ColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Translation 5,5
draw.StrokeColor new GM16ColorRGBMBS("blue")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.13.105 Viewbox(x1 as Integer, y1 as Integer, x2 as Integer, y2 as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Dimensions of the output viewbox.
Notes: If the image is to be written to a vector format (e.g. MVG or SVG), then a PushGraphicContext() object should be pushed to the head of the list, followed by a Viewbox() statement to establish the output canvas size. A matching PopGraphicContext() object should be pushed to the tail of the list.

4.13.106 Properties

4.13.107 Image as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image this graphics object belongs to.
Notes: (Read only property)

4.14 class GM16ImageArrayMBS

4.14.1 class GM16ImageArrayMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an array of images in GraphicsMagick.
Example:

```
// extract all layers of photoshop file
dim file as FolderItem = SpecialFolder.Desktop.Child("test.psd")
dim images as new GM16ImageArrayMBS

images.readImages(file.NativePath)

dim c as Integer = images.size
for i as Integer = 0 to c-1
dim image as GM16ImageMBS = images.Image(i)
file = SpecialFolder.Desktop.Child(image.FileName+"."+str(i)+".png")
image.write(file)
next
```

Notes: Can be used to assemble/disassemble gif images.

4.14.2 Methods

4.14.3 animateImages

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Animate a sequence of image frames.
Notes: Image frames are displayed in succession, creating an animated effect. The animation options are taken from the first image frame. This feature is only supported under X11 at the moment.

4.14.4 append(image as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds an image to the end of the array.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child(”test.gif”)
g.readImages(file.NativePath)

// put copy of first image on the back
dim n as GM16ImageMBS = g.FirstImage
g.append n

// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child(”output.gif”)

g.writeImages(output.NativePath)
```

Notes: Instead of gif, you can also use tif files.

4.14.5 appendImages(stack as boolean = false) as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Append a sequence of image frames, writing the result to new image.
Notes: All the input image frames must have the same width or height. Image frames of the same width are stacked top-to-bottom. Image frames of the same height are stacked left-to-right. If the stack parameter is false, rectangular image frames are stacked left-to-right otherwise top-to-bottom.

4.14.6 averageImages as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Average a sequence of image frames, writing the result to averagedImage.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child(”test.gif”)
g.readImages(file.NativePath)

// averageImages
dim n as GM16ImageMBS = g.averageImages
Backdrop = n.CopyPicture
```

Notes: All the input image frames must be the same size in pixels.

4.14.7 coalesceImages as GM16ImageArrayMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create a coalesced image sequence obtained by "playing" the image sequence (observing page offsets and disposal methods) to create a new image sequence in which all frames are full size and completely rendered.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// deconstruct
g = g.coalesceImages

// write gif
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")
g.writeImages output.NativePath
```

Notes: Note that if the original image sequence relied on page offsets and disposal methods that the resulting sequence will be larger (perhaps much larger) then the original. This is useful for GIF animation sequences that have page offsets and disposal methods. The resuting image sequence is returned.

4.14.8 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty image array.

4.14.9 deconstructImages as GM16ImageArrayMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Break down an image sequence into constituent parts.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// deconstruct
```

```
g = g.deconstructImages

// write gif
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")
g.writeImages output.NativePath
```

Notes: This is useful for creating GIF or MNG animation sequences.

4.14.10 displayImages

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Display a sequence of image frames.
Notes: Through use of a pop-up menu, image frames may be selected in succession. This feature is fully supported under X11 but may have only limited support in other environments.
Caution: if an image format is is not compatable with the display visual (e.g. JPEG on a colormapped display) then the original image will be altered. Use a copy of the original if this is a problem.

display methods are not supported currently.

4.14.11 FirstImage as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns first image in array.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// show first image
dim img as GM16ImageMBS = g.FirstImage

// convert to true color for CopyPicture to work
const TrueColorType=6
img.type=TrueColorType

Backdrop = img.CopyPicture
```

4.14.12 flattenImages as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Merge a sequence of image frames which represent image layers into a single composited representation.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// put copy of first image on the back
dim n as GM16ImageMBS = g.flattenImages

Backdrop = n.CopyPicture
```

Notes: Returns the flattened image. This function is useful for combining Photoshop layers into a single image.

4.14.13 Image(index as Integer) as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries image with given index.

4.14.14 insert(image as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Inserts an image on the front.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// put copy of first image on the front
dim n as GM16ImageMBS = g.FirstImage
g.insert n

// write to file
```

```
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

4.14.15 LastImage as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns last image in array.

4.14.16 mapImages(map as GM16ImageMBS, dither as boolean = true, measureError as boolean = false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replace the colors of a sequence of images with the closest color from a reference image.
Notes: Set dither to true to enable dithering. Set measureError to true in order to evaluate quantization error.

4.14.17 montageImages(options as GM16MontageMBS) as GM16ImageArrayMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create a composite image by combining several separate image frames.
Example:

```
// build montage
dim StackingMontage as New GM16MontageMBS
StackingMontage.backgroundColor = New GM16ColorMBS(&cE7E7E7)
StackingMontage.fillColor = New GM16ColorMBS(&c000000)
StackingMontage.tile = New GM16GeometryMBS("1x20")
StackingMontage.geometry = New GM16GeometryMBS("160x120+5+5")
StackingMontage.font = "Helvetica"
StackingMontage.pointSize = 12
StackingMontage.title = "Title goes here"

// make picture
dim logo as Picture = LogoMBS(500)
dim image as New GM16ImageMBS(logo)

image.label("Sample label")
```

```
// Put the current image into the array
Dim StackingFrames As new GM16ImageArrayMBS
StackingFrames.insert(image)

// show result
dim resultImages as GM16ImageArrayMBS = StackingFrames.montageImages(StackingMontage)
Backdrop = resultImages.Image(0).CopyPicture
```

Notes: Multiple frames may be generated in the output array depending on the tile setting and the number of image frames montaged. Montage options are provided via the parameter options. Options set in the first image frame (backgroundColor, borderColor, matteColor, fillColor, strokeColor, font and fontPointsize) are also used as options by montageImages().

4.14.18 morphImages(frames as Integer) as GM16ImageArrayMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Morph a seqence of image frames.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// coalesce to make sure we have full images
g = g.coalesceImages
// morph to 10 pictures
g = g.morphImages(10)

// write gif
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")
g.writeImages output.NativePath
```

Notes: This algorithm expands the number of image frames (output to the new image array) by adding the number of intervening frames specified by frames such that the original frames morph (blend) into each other when played as an animation.

4.14.19 mosaicImages as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Inlay a number of images to form a single coherent picture.
Notes: The result image argument is updated with a mosaic constructed from the image sequence.

4.14.20 quantizeImages(measureError as boolean = false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Quantize colors in images using current quantization settings.
Notes: Set measureError to true in order to measure quantization error.

4.14.21 readImages(blob as GM16BlobMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read a sequence of image frames into existing container (appending to array) from blob.
See also:

- 4.14.22 readImages(imageSpec as string) 187

4.14.22 readImages(imageSpec as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read a sequence of image frames into existing container (appending to array) with image names specified in the string imageSpec.
See also:

- 4.14.21 readImages(blob as GM16BlobMBS) 187

4.14.23 remove(index as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes the image with the given index.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)
```

```
// remove first
g.remove 0

// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

Notes: Index should be between 0 and size-1.

4.14.24 reverse

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reverses the order of images in the array.

4.14.25 writeImages(blob as GM16BlobMBS, adjoin as boolean = true)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Writes images to the given blob object.
Notes: Write images in container to in-memory BLOB specified by Blob blob. Set adjoin to false to write a set of image frames via a wildcard imageSpec (e.g. image%02d.miff).
Caution: if an image format is selected which is capable of supporting fewer colors than the original image or quantization has been requested, the original image will be quantized to fewer colors. Use a copy of the original if this is a problem.
See also:

- 4.14.26 writeImages(imageSpec as string, adjoin as boolean = true) 188

4.14.26 writeImages(imageSpec as string, adjoin as boolean = true)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Writes images to the given path.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)
```

```
// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

Notes: Write images in container to file specified by string imageSpec. Set adjoin_ to false to write a set of image frames via a wildcard imageSpec (e.g. image%02d.miff).
The wildcard must be one of %0Nd, %0No, or %0Nx.
Caution: if an image format is selected which is capable of supporting fewer colors than the original image or quantization has been requested, the original image will be quantized to fewer colors. Use a copy of the original if this is a problem.
See also:

- 4.14.25 writeImages(blob as GM16BlobMBS, adjoin as boolean = true) 188

4.14.27 Properties

4.14.28 empty as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Checks whether image array is empty.
Notes: Returns true if array is empty or false if not.
(Read only property)

4.14.29 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal handle of the image array.
Notes: Should always be non zero.
(Read and Write property)

4.14.30 size as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns number of images in this array.
Example:

```
// read gif
dim g as new GM16ImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// display number of images
MsgBox str(g.size)
```

Notes: (Read only property)

4.15 class GM16ImageChannelStatisticsMBS

4.15.1 class GM16ImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The statistics for image channel.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim stat as GM16ImageStatisticsMBS = g.statistics
dim gs as GM16ImageChannelStatisticsMBS = stat.blue

MsgBox "blue channel: "+str(gs.minimum)+"-"+str(Gs.maximum)+", mean "+str(gs.mean)
```

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.15.2 Methods

4.15.3 Constructor

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The private constructor.

4.15.4 Properties

4.15.5 maximum as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Maximum value observed.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim stat as GM16ImageStatisticsMBS = g.statistics
dim gs as GM16ImageChannelStatisticsMBS = stat.green

MsgBox "maximum green color: "+str(gs.maximum)
```

Notes: (Read only property)

4.15.6 mean as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Average (mean) value observed.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim stat as GM16ImageStatisticsMBS = g.statistics
dim r as GM16ImageChannelStatisticsMBS = stat.red

MsgBox "mean red color: "+str(R.mean)
```

Notes: (Read only property)

4.15.7 minimum as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Minimum value observed.
Notes: (Read only property)

4.15.8 standardDeviation as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Standard deviation, sqrt(variance).
Notes: (Read only property)

4.15.9 variance as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Variance.

Notes: (Read only property)

4.16 class GM16ImageMBS

4.16.1 class GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image is the primary object in Magick++ and represents a single image frame (see image design).
Example:

```
dim c as new GM16ColorMBS("white")
dim g as new GM16GeometryMBS(100,100)
dim image as new GM16ImageMBS(g, c)
```

Notes: With MBS Plugin 14.0 we offer this classes in 8bit (GM prefix) or 16bit (GM16 prefix).

The GM16ImageArrayMBS class must be used to operate on image sequences or images (e.g. of format GIF, TIFF, MIFF, Postscript, & MNG) which are comprized of multiple image frames. Individual frames of a multi-frame image may be requested by adding array-style notation to the end of the file name (e.g. "animation.gif [3] " retrieves the fourth frame of a GIF animation. Various image manipulation operations may be applied to the image. Attributes may be set on the image to influence the operation of the manipulation operations. The GM16PixelsMBS class provides low-level access to image pixels.
Blog Entries

- News from the MBS Xojo Plugins Version 23.3
- News from the MBS Xojo Plugins Version 23.2
- Change brightness saturation and hue by Magick
- News from the MBS Xojo Plugins Version 20.4
- News from the MBS Xojo Plugins Version 20.1
- MBS Xojo Plugins, version 20.1pr5
- MBS Xojo / Real Studio Plugins, version 15.1pr4
- MBS Xojo / Real Studio Plugins, version 15.0pr9

Xojo Developer Magazine

- 20.2, pages 82 to 83: Wifi QR Code, Embedding your Wifi password in a QR code by Stefanie Juchmes

4.16.2 Methods

4.16.3 adaptiveThreshold(width as UInt32, height as UInt32, offset as double = 0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply adaptive thresholding to the image.
Notes: see:
http://www.dai.ed.ac.uk/HIPR2/adpthrsh.htm

Adaptive thresholding is useful if the ideal threshold level is not known in advance, or if the illumination gradient is not constant across the image. Adaptive thresholding works by evaulating the mean (average) of a pixel region (size specified by width and height) and using the mean as the thresholding value. In order to remove residual noise from the background, the threshold may be adjusted by subtracting a constant offset (default zero) from the mean to compute the threshold.

4.16.4 addNoise(noise as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add noise to image with the specified noise type.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.addNoise(image.GaussianNoise)

Backdrop=image.CopyPicture
```

Notes: Use one of this constants: GaussianNoise, ImpulseNoise, LaplacianNoise, MultiplicativeGaussianNoise, PoissonNoise, UniformNoise.

4.16.5 addNoiseChannel(channel as Integer, noise as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add noise to an image channel with the specified noise type. The channel parameter specifies the channel to add noise to.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.addNoiseChannel(image.BlueChannel, image.ImpulseNoise)

Backdrop=image.CopyPicture
```

Notes: The noiseType parameter specifies the type of noise.
Use one of this constants: GaussianNoise, ImpulseNoise, LaplacianNoise, MultiplicativeGaussianNoise, PoissonNoise, UniformNoise.

4.16.6 affineTransform(sx as Double, sy as Double, rx as Double, ry as Double, tx as Double, ty as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies an affine transformation to the drawing matrix.
Notes: Specify a transformation matrix to adjust scaling, rotation, and translation (coordinate transformation) for subsequently drawn objects in the same or decendent drawing context. The sx & sy parameters represent the x & y scale factors, the rx & ry parameters represent the x & y rotation, and the tx & ty parameters represent the x & y translation.

4.16.7 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate using specified text, bounding area, and placement gravity.
Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.16.8 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer, degrees as Double) 197
- 4.16.9 annotate(text as string, gravity as Integer) 198
- 4.16.10 annotate(text as string, location as GM16GeometryMBS) 199

4.16.8 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer, degrees as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate with text using specified text, bounding area, placement gravity, and rotation.
Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.16.7 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer) 196
- 4.16.9 annotate(text as string, gravity as Integer) 198
- 4.16.10 annotate(text as string, location as GM16GeometryMBS) 199

4.16.9 annotate(text as string, gravity as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate with text (bounding area is entire image) and placement gravity.
Example:

```
dim White as new GM16ColorGrayMBS(1)
dim Black as new GM16ColorGrayMBS(0)
dim geo as new GM16GeometryMBS("300x200")

dim g as new GM16ImageMBS(geo, White)

g.antiAlias = False
g.fillColor = Black
g.lineWidth = 1
g.strokeColor = Black
g.font = "@/Library/Fonts/Tahoma.ttf"
g.fontPointsize = 15

g.annotate("Hello World", g.SouthGravity)

Backdrop = g.CopyPicture
```

Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.16.7 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer) 196
- 4.16.8 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer, degrees as Double) 197
- 4.16.10 annotate(text as string, location as GM16GeometryMBS) 199

4.16.10 annotate(text as string, location as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate using specified text, and placement location.
Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.16.7 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer) 196
- 4.16.8 annotate(text as string, boundingArea as GM16GeometryMBS, gravity as Integer, degrees as Double) 197
- 4.16.9 annotate(text as string, gravity as Integer) 198

4.16.11 attributeValues as dictionary

Plugin Version: 17.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A dictionary with all attributes.
Notes: As attributes are created on demand, this will only return all so far generated attributes.

4.16.12 autoOrient

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Orient image to be right-side up based on its current orientation attribute.
Notes: This allows the image to be viewed correctly when the orientation attribute is not available, or is not respected.

4.16.13 blur(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur an image with the specified blur factor.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.blur(30,10)

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.16.14 blurChannel(channel as Integer, radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur an image channel with the specified blur factor.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.blurChannel(image.BlueChannel, 30,10)

Backdrop=image.CopyPicture
```

Notes: The channel parameter specifies the channel to modify. The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.16.15 border

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Border image (add border to image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.border

Backdrop=image.CopyPicture
```

Notes: The color of the border is specified by the borderColor attribute.
See also:

- 4.16.16 border(geometry as GM16GeometryMBS) 202

4.16.16 border(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Border image (add border to image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.border GM16GeometryMBS.Make(10,10)

Backdrop=image.CopyPicture
```

Notes: The color of the border is specified by the borderColor attribute.
See also:

- 4.16.15 border 202

4.16.17 borderGeometryDefault as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The default geometry description for border.

4.16.18 cacheThreshold(threshold as UInt32)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pixel cache threshold in megabytes.
Notes: Once this memory threshold is exceeded, all subsequent pixels cache operations are to/from disk. This setting is shared by all Image objects.

4.16.19 cdl(cdl as string)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Bake in the ASC-CDL.
Notes: Bake in the ASC-CDL, which is a convention for the for the exchange of basic primary color grading information between for the exchange of basic primary color grading information between equipment and software from different manufacturers. It is a useful transform for other purposes as well.

4.16.20 channel(channel as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extract channel from image.
Notes: Use this option to extract a particular channel from the image. MatteChannel for example, is useful for extracting the opacity values from an image.

4.16.21 charcoal(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Charcoal effect image (looks like charcoal sketch).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.charcoal

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.16.22 chop(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chop image (remove vertical or horizontal subregion of image).

4.16.23 colorHistogram as dictionary

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Calculates histogram.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim d as Dictionary = g.colorHistogram

MsgBox str(d.Count)+" color"

// check first color
dim c as GM16ColorMBS = d.key(0)

MsgBox "Color "+str(c.colorValue)+": "+str(d.Value(c))
```

Notes: The dictionary has a GM16ColorMBS/GMColor16MBS object as key for each color and an unsigned integer as value.

4.16.24 colorize(opacity as UInt32, penColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorize image with pen color, using specified percent opacity.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.colorize(10, new GM16ColorMBS("red"))

Backdrop=image.CopyPicture
```

See also:

- 4.16.25 colorize(opacityRed as UInt32, opacityGreen as UInt32, opacityBlue as UInt32, penColor as GM16ColorMBS) 205

4.16.25 colorize(opacityRed as UInt32, opacityGreen as UInt32, opacityBlue as UInt32, penColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorize image with pen color, using specified percent opacity for red, green, and blue quantums.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.colorize(10, 0, 5, new GM16ColorMBS("red"))

Backdrop=image.CopyPicture
```

See also:

- 4.16.24 colorize(opacity as UInt32, penColor as GM16ColorMBS) 204

4.16.26 colorMap as GM16ColorMBS()

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries array with all colors in color map.
See also:

- 4.16.309 colorMap(index as UInt32) as GM16ColorMBS 305

4.16.27 colorMatrix(order as Integer, ColorMatrix() as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply a color matrix to the image channels.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)

dim m(8) as Double

m(0) = 0.25
m(1) = 0
```

```
m(2) = 0.25

m(3) = 0
m(4) = 0
m(5) = 0

m(6) = 0.25
m(7) = 0
m(8) = 0.25

g.colorMatrix 3, m

Backdrop = g.CopyPicture
```

Notes: The user supplied matrix may be of order 1 to 5 (1x1 through 5x5).

4.16.28 CombinePictureWithMask as picture

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image with mask.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

dim c as new GM16ColorMBS("white")
image.transparent(c)

Backdrop=image.CombinePictureWithMask
```

Notes: Internally this calls Width and Height, CopyPicture and CopyMask.

4.16.29 compare(image as GM16ImageMBS) as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compare current image with another image.
Notes: Sets meanErrorPerPixel, normalizedMaxError, and normalizedMeanError in the current image. False is returned if the images are identical. An ErrorOption exception is thrown if the reference image columns, rows, colorspace, or matte differ from the current image:

4.16.30 composite(compositeImage as GM16ImageMBS, gravity as Integer, CompositeOperator as Integer = 2)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose an image onto another at specified x and y offset and using a specified algorithm.

4.16.31 compositeAt(compositeImage as GM16ImageMBS, offset as GM16GeometryMBS, CompositeOperator as Integer = 2)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose an image onto another at specified x and y offset and using a specified algorithm.

4.16.32 compositeXY(compositeImage as GM16ImageMBS, xOffset as Integer, yOffset as Integer, CompositeOperator as Integer = 2)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose an image onto another at specified x and y offset and using a specified algorithm.

4.16.33 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Default constructor.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GM16ImageMBS
dim blob as new GM16BlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

See also:

- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.34 Constructor(blob as GM16BlobMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image from in-memory Blob.
See also:

- 4.16.33 Constructor 207
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size from in-memory Blob.
See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size and depth from in-memory Blob.
See also:

- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213

- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size, depth, and format from in-memory Blob.
See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size, depth, and format from in-memory Blob.
Example:

```
// first open file and read data in blob
SVG_File = FolderItem.ShowOpenFileDialog("")
If SVG_File = Nil Then Return

dim stream as BinaryStream = BinaryStream.Open(SVG_File)
dim data as string = stream.Read(stream.Length)
dim blob as new GM16BlobMBS(data)
```

```
// 400 width and 400 height
Dim geo As New GM16GeometryMBS(400, 400, 0, 0)

// pass type here to have GraphicsMagick know it since there is no file name in blob:
dim g as New GM16ImageMBS(blob, geo, ”svg”)
```

See also:

- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.39 Constructor(file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct from image file.
See also:

- 4.16.33 Constructor 207
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212

- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.40 Constructor(other as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an image by making a copy of the existing one.
See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.41 Constructor(Path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct from image file or image specification.
See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210

- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.42 Constructor(pic as picture)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new GMImage with the given picture.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)
Backdrop=image.CopyPicture
```

Notes: Pixels from both the picture and picture's mask.
See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS) 214
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.43 Constructor(size as GM16GeometryMBS, ColorValue as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct a blank image canvas of specified size and color.
Example:

```
dim g as new GM16GeometryMBS(600,600)
dim c as new GM16ColorRGBMBS(1.0,0.0,0.0) // red
dim image as new GM16ImageMBS(g, c)

const TrueColorType=6

// Ensure that there are no other references to this image.
image.modifyImage
// Set the image type to TrueColor DirectClass representation.
image.type=TrueColorType

Backdrop=image.CopyPicture(0,0,600,600)
```

See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.37 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212
- 4.16.41 Constructor(Path as string) 212
- 4.16.42 Constructor(pic as picture) 213
- 4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 214

4.16.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes single image frame from an array of raw pixels, with specified storage type (ConstituteImage).
Example:

```
dim data as new memoryblock(2048*2048) // your data
dim image as new GM16ImageMBS(2048, 2048, "I", GM16ImageMBS.StorageTypeCharPixel, data)
```

Notes: Returns an Image corresponding to an image stored in a raw memory array format. The pixel data must be in scanline order top-to-bottom. The data can be unsigned char, unsigned short int, unsigned int, unsigned long, float, or double. Float and double require the pixels to be normalized to the range [0..1] , otherwise the range is [0..MaxVal] where MaxVal is the maximum possible value for that type.

Note that for most 32-bit architectures the size of an unsigned long is the same as unsigned int, but for 64-bit architectures observing the LP64 standard, an unsigned long is 64 bits, while an unsigned int remains 32 bits. This should be considered when deciding if the data should be described as "Integer" or "Long".

For example, to create a 640x480 image from unsigned red-green-blue character data, use

image = new GM16ImageMBS(640, 480, "RGB", GM16ImageMBS.StorageTypeCharPixel, pixels);

width: width in pixels of the image.
height: height in pixels of the image.
map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (same as Transparency), O = Opacity, T = Transparency, C = cyan, Y = yellow, M = magenta, K = black, or I = intensity (for grayscale). Specify "P" = pad, to skip over a quantum which is intentionally ignored. Creation of an alpha channel for CMYK images is currently not supported.
type: Define the data type of the pixels. Float and double types are expected to be normalized [0..1] otherwise [0..MaxRGB] . Choose from these types: StorageTypeCharPixel, StorageTypeShortPixel, StorageTypeIntegerPixel, StorageTypeLongPixel, StorageTypeFloatPixel, or StorageTypeDoublePixel.
pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.
See also:

- 4.16.33 Constructor 207
- 4.16.34 Constructor(blob as GM16BlobMBS) 208
- 4.16.35 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS) 209
- 4.16.36 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, depth as UInt32) 209
- 4.16.38 Constructor(blob as GM16BlobMBS, geometry as GM16GeometryMBS, Magick as string) 210
- 4.16.39 Constructor(file as folderitem) 211
- 4.16.40 Constructor(other as GM16ImageMBS) 212

4.16.45 contrast(sharpen as UInt32)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Contrast image (enhance intensity differences in image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.contrast(10)

Backdrop=image.CopyPicture
```

4.16.46 convolve(order as Integer, ColorMatrix() as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Convolve image.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)

dim m(8) as Double

m(0) = 0.25
m(1) = 0
m(2) = 0.25

m(3) = 0
m(4) = 0
m(5) = 0

m(6) = 0.25
m(7) = 0
m(8) = 0.25

g.convolve 3, m
```

```
Backdrop = g.CopyPicture
```

Notes: Applies a user-specified convolution to the image.
order represents the number of columns and rows in the filter kernel.
kernel is an array of doubles representing the convolution kernel.

4.16.47 CopyPicture as picture

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image and returns it as a new picture.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle
draw.Rectangle(250, 250, 100, 100)

Backdrop=image.CopyPicture
```

Notes: You may need to set image type to RGB to get it working.
See also:

- 4.16.48 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture 217

4.16.48 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image and returns it as a new picture.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)
```

```
image.threshold 127

// convert to RGB so CopyPicture works
image.type = image.TrueColorType
Backdrop=image.CopyPicture(0,0,250,250)
```

Notes: You may need to set image type to RGB to get it working.
See also:

- 4.16.47 CopyPicture as picture 217

4.16.49 CopyPictureMask as picture

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image's mask and returns it as a new picture.
Example:

```
Dim f As FolderItem = SpecialFolder.Desktop.Child("test.png")
Dim g As New GM16ImageMBS(f)

// get image with mask
Dim p As picture = g.CopyPicture
p.mask = g.CopyPictureMask

window1.Backdrop = p
```

See also:

- 4.16.50 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture 218

4.16.50 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image's mask and returns it as a new picture.
See also:

- 4.16.49 CopyPictureMask as picture 218

4.16.51 CopyPixelsMemory as Memoryblock

Plugin Version: 15.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copy the pixels as they are into a memoryblock.
Notes: Optional specify rectangle.
Returns nil on low memory or bad parameter. Image must be of type class direct (not palette picture).
Order of pixel data is normally Red, Green, Blue, Opacity. Or Cyan, Magenta, Yellow, Black for CMYK images.
For GM16ImageMBS, the data is 8bit per channel. For GMImage16MBS, the data is 16bit per channel.
See also:

- 4.16.52 CopyPixelsMemory(x as Integer, y as Integer, width as Integer, height as Integer) as Memoryblock 219

4.16.52 CopyPixelsMemory(x as Integer, y as Integer, width as Integer, height as Integer) as Memoryblock

Plugin Version: 15.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copy the pixels as they are into a memoryblock.
Notes: Optional specify rectangle.
Returns nil on low memory or bad parameter. Image must be of type class direct (not palette picture).
Order of pixel data is normally Red, Green, Blue, Opacity. Or Cyan, Magenta, Yellow, Black for CMYK images.
For GM16ImageMBS, the data is 8bit per channel. For GMImage16MBS, the data is 16bit per channel.
See also:

- 4.16.51 CopyPixelsMemory as Memoryblock 219

4.16.53 CreateHBITMAP as Ptr

Plugin Version: 15.1, Platform: Windows, Targets: All.

Function: Creates a HBITMAP for the image for use with Windows Declares.
Example:

```
// get test image
dim logo as Picture = LogoMBS(500)

// create GraphicsMagick image
dim g as new GM16ImageMBS(logo)

// make a HBitmap
dim hBitmap as ptr = g.CreateHBITMAP

// convert back to Xojo picture
```

```
dim pic as Picture = WindowsBitmapMBS.HBitmapToPicture(hBitmap, true)

// show in window
Backdrop = pic

// and cleanup memory
WindowsBitmapMBS.DeleteBitmap(hBitmap)
```

Notes: The HBITMAP returned needs to be freed when you are done with it or you risk having a memory leak.

4.16.54 crop(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crop image (return subregion of original image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.crop GM16GeometryMBS.Make(100,200)

Backdrop=image.CopyPicture
```

4.16.55 cycleColormap(amount as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Cycle (rotate) image colormap.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.cycleColormap(5)

image.type = image.TrueColorType

Backdrop=image.CopyPicture
```

4.16.56 Describe(verbose as Integer = 1) as String

Plugin Version: 23.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Describes an image by printing its attributes.
Example:

```
Dim f As FolderItem = FindFile("test.jpg")

Dim g As New GM16ImageMBS(f)
Dim s As String = g.Describe(2)
Break
```

Notes: Attributes include the image width, height, size, and others.

verbose: Whether output should be verbose.
Default is 1. Pass 0 to get a shorted output.
Pass 2 to count the number of colors in the image.

4.16.57 despeckle

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Despeckle image (reduce speckle noise).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.despeckle

Backdrop=image.CopyPicture
```

4.16.58 display

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Display image on screen.
Notes: Caution: if an image format is is not compatible with the display visual (e.g. JPEG on a colormapped display) then the original image will be altered. Use a copy of the original if this is a problem.

The plugin is not compiled with X11 so this call may not be useful.

4.16.59 edge(radius as Double=0.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Edge image (hilight edges in image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.edge

Backdrop=image.CopyPicture
```

Notes: The radius is the radius of the pixel neighborhood. Specify a radius of zero for automatic radius selection.

4.16.60 emboss(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Emboss image (hilight edges with 3D effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.emboss

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.16.61 enhance

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhance image (minimize noise).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)
```

```
image.enhance

Backdrop=image.CopyPicture
```

4.16.62 erase

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set all image pixels to the current background color.

4.16.63 extent(geo as GM16GeometryMBS)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// extend image to fit
dim geo as new GM16GeometryMBS(500,500)
image.extent geo

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.16.64 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS) 224
- 4.16.65 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS, gravity as Integer) 224
- 4.16.66 extent(geo as GM16GeometryMBS, gravity as Integer) 225

4.16.64 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// extend image to fit
dim geo as new GM16GeometryMBS(500,500)
dim col as GM16ColorMBS = GM16ColorMBS.Black
image.extent geo, col

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.16.63 extent(geo as GM16GeometryMBS) 223
- 4.16.65 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS, gravity as Integer) 224
- 4.16.66 extent(geo as GM16GeometryMBS, gravity as Integer) 225

4.16.65 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS, gravity as Integer)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// extend image to fit
dim geo as new GM16GeometryMBS(500,500)
dim col as GM16ColorMBS = GM16ColorMBS.Black
image.extent geo, col
```

```
window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.16.63 extent(geo as GM16GeometryMBS) 223
- 4.16.64 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS) 224
- 4.16.66 extent(geo as GM16GeometryMBS, gravity as Integer) 225

4.16.66 extent(geo as GM16GeometryMBS, gravity as Integer)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// resize proportionally to fit
dim geo as new GM16GeometryMBS(500,500)
image.extent geo, image.CenterGravity

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.16.63 extent(geo as GM16GeometryMBS) 223
- 4.16.64 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS) 224
- 4.16.65 extent(geo as GM16GeometryMBS, backgroundColor as GM16ColorMBS, gravity as Integer) 224

4.16.67 flip

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flip image (reflect each scanline in the vertical direction).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.flip

Backdrop=image.CopyPicture
```

4.16.68 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.16.69 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 226
- 4.16.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS) 227
- 4.16.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 227

4.16.69 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match:
See also:

- 4.16.68 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS) 226
- 4.16.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS) 227

- 4.16.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 227

4.16.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.16.68 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS) 226
- 4.16.69 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 226
- 4.16.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 227

4.16.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match:
See also:

- 4.16.68 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS) 226
- 4.16.69 floodFillColor(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 226
- 4.16.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GM16ColorMBS) 227

4.16.72 floodFillOpacity(x as UInt32, y as UInt32, opacity as UInt32, PaintMethod as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill pixels matching color (within fuzz factor) of target pixel(x,y) with replacement opacity value using method.

4.16.73 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.16.74 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 228
- 4.16.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS) 228
- 4.16.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 229

4.16.74 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.16.73 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS) 228
- 4.16.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS) 228
- 4.16.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 229

4.16.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.16.73 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS) 228
- 4.16.74 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 228

- 4.16.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 229

4.16.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.16.73 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS) 228
- 4.16.74 floodFillTexture(point as GM16GeometryMBS, fillColor as GM16ColorMBS, borderColor as GM16ColorMBS) 228
- 4.16.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GM16ColorMBS) 228

4.16.77 flop

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flop image (reflect each scanline in the horizontal direction).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.flop

Backdrop=image.CopyPicture
```

4.16.78 FontMap as string

Plugin Version: 20.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries current font map in use.
Notes: The MBS Plugin provides to GraphicsMagick the font map to use.
This is a XML defining which fonts are available.
Use this function to learn what fonts may be available or debug to see why a font doesn't load.

4.16.79 fontTypeMetrics(name as string) as GM16TypeMetricMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain font metrics for text string given current font, pointsize, and density settings.

4.16.80 formatExpression(expression as string) as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format the specified expression similar to command line '-format'.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: For example "%wx%h" is converted to a string containing image WIDTHxHEIGHT like "640x480".

4.16.81 frame

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a decorative frame around the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.frame

Backdrop=image.CopyPicture
```

See also:

- 4.16.82 frame(geometry as GM16GeometryMBS) 231
- 4.16.83 frame(width as UInt32, height as UInt32, innerBevel as Integer=6, outerBevel as Integer=6) 231

4.16.82 frame(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a decorative frame around the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.frame(GM16GeometryMBS.Make("10x10"))

Backdrop=image.CopyPicture
```

See also:

- 4.16.81 frame 230
- 4.16.83 frame(width as UInt32, height as UInt32, innerBevel as Integer=6, outerBevel as Integer=6) 231

4.16.83 frame(width as UInt32, height as UInt32, innerBevel as Integer=6, outerBevel as Integer=6)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a decorative frame around the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.frame(15,15)

Backdrop=image.CopyPicture
```

See also:

- 4.16.81 frame 230
- 4.16.82 frame(geometry as GM16GeometryMBS) 231

4.16.84 frameGeometryDefault as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The default geometry description for frame.

4.16.85 gamma(gammaRed as Double, gammaGreen as Double, gammaBlue as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma correct the image or individual image channels.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.gamma(1,2,3)

Backdrop=image.CopyPicture
```

See also:

- 4.16.252 gamma as Double 288

4.16.86 gaussianBlur(width as Double, sigma as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gaussian blur image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.gaussianBlur(30, 10)

Backdrop=image.CopyPicture
```

Notes: The number of neighbor pixels to be included in the convolution mask is specified by width. The standard deviation of the gaussian bell curve is specified by sigma

4.16.87 gaussianBlurChannel(channel as Integer, width as Double, sigma as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gaussian blur image channel.
Notes: The number of neighbor pixels to be included in the convolution mask is specified by width. The standard deviation of the gaussian bell curve is specified by sigma.

4.16.88 getChromaBluePrimary(byref x as Double, byref y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity blue primary point.

4.16.89 getchromaGreenPrimary(byref x as Double, byref y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity green primary point.
Notes: e.g. x=0.3, y=0.6

4.16.90 getchromaRedPrimary(byref x as Double, byref y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity red primary point
Notes: e.g. x=0.64, y=0.33

4.16.91 getchromaWhitePoint(byref x as Double, byref y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity white point
Notes: e.g. x=0.3127, y=0.329

4.16.92 getConstPixels(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers read-only pixels from the image to the pixel cache as defined by the specified region

4.16.93 GetEXIFOrientation(byref orientation as integer) as boolean

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries orientation from EXIF.
Notes: Orientation is set to number from 0 to 8 depending on rotation. -1 if unknown.
This function can only read orientation, if there is an EXIF block in image.
Returns true for success and false for failure.

For new development, please use ExifTagsMBS class instead.

4.16.94 getPixels(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers pixels from the image to the pixel cache as defined by the specified region.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)

// get pointer to some pixels to write
dim x as ptr = g.getPixels(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
g.syncPixels

// show
me.Backdrop = g.CopyPicture
```

Notes: Modified pixels may be subsequently transferred back to the image via syncPixels. This method is valid for DirectClass images.

4.16.95 Graphics as GM16GraphicsMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a graphics object for this image.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: Using the graphics object you can draw on the image.

4.16.96 haldClut(image as GM16ImageMBS)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply a color lookup table (Hald CLUT) to the image.

4.16.97 Hash(Size as Integer = 8) as String

Plugin Version: 19.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Calculates a hash of the image.
Example:

```
Dim p As Picture = LogoMBS(500)
Dim g As New GM16ImageMBS(p)
msgbox g.Hash
```

Notes: Hash is returned as 64 characters being 1 or 0.
We convert image to 8x8, turn grayscale and check if pixels are above or below mean value.
This hash is quite immune against resizing, compression artifacts and hue changes.
You can use LevenshteinDistanceMBS or JaroWinklerDistanceMBS to compare two hashes.

Added size parameter for version 22.4:
The size of the bitmap. Value from 8 to 1024. Default is 8.

4.16.98 implode(factor as Double=0.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Implode image (special effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.implode(0.3)

Backdrop=image.CopyPicture
```

4.16.99 IsLoggingEnabled as Boolean

Plugin Version: 21.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Checks whether logging is enabled.
Notes: Returns true if we log GraphicsMagick usage.

4.16.100 JasperLibVersion as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries version string for jasper library.

4.16.101 label(text as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Assign a label to an image.
Notes: Use this option to assign a specific label to the image. Optionally you can include the image filename, type, width, height, or scene number in the label by embedding special format characters. If the first character of string is @, the image label is read from a file titled by the remaining characters in the string. When converting to Postscript, use this option to specify a header string to print above the image.
See also:

- 4.16.263 label as string 291

4.16.102 level(black_point as Double, white_point as Double, mid_point as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Level image to increase image contrast, and/or adjust image gamma.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.level(0, 127.0)

Backdrop=image.CopyPicture
```

Notes: Adjust the levels of the image by scaling the colors falling between specified white and black points to the full available quantum range. The parameters provided represent the black, mid (gamma), and white points. The black point specifies the darkest color in the image. Colors darker than the black point are set to zero. Mid point (gamma) specifies a gamma correction to apply to the image. White point specifies the lightest color in the image. Colors brighter than the white point are set to the maximum quantum value. The black and white point have the valid range 0 to MaxRGB while mid (gamma) has a useful range of 0 to ten:

4.16.103 levelChannel(channel as Integer, black_point as Double, white_point as Double, mid_point as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Level image channel to increase image contrast, and/or adjust image gamma.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.levelChannel(image.BlueChannel, 0, 127.0)

Backdrop=image.CopyPicture
```

Notes: Adjust the levels of the image channel by scaling the colors falling between specified white and black points to the full available quantum range. The parameters provided represent the black, mid (gamma), and white points. The black point specifies the darkest color in the image. Colors darker than the black point are set to zero. Mid point (gamma) specifies a gamma correction to apply to the image. White point specifies the lightest color in the image. Colors brighter than the white point are set to the maximum quantum value. The black and white point have the valid range 0 to MaxRGB while mid (gamma) has a useful range of 0 to ten.

4.16.104 LibVersion as String

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the version string of the GraphicsMagick library.

4.16.105 LoadIconvLibrary(path as String, byref Error as String) as boolean

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the iconv library.
Notes: The GraphicsMagick classes may use libiconv for text encoding conversion.
If you explicitly need, you can load the library on start of solution.

MBS Plugin may try to load iconv.dll/dylib/so automatically when first iconv function is called.

4.16.106 MagickVersion as string

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries the version text of the GraphicsMagick library.

4.16.107 magnify

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Magnify image by integral size (double the dimensions)
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.magnify

Backdrop=image.CopyPicture
```

4.16.108 map(mapImage as GM16ImageMBS, dither as boolean=false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Remap image colors with closest color from a reference image.
Example:

```
// some picture we want to map colors
dim pic as Picture = LogoMBS(500)

// build a picture with palette
dim backgroundColor as new GM16ColorMBS(255,255,255) // white
dim size as new GM16GeometryMBS(10,10)

dim i as new GM16ImageMBS(pic)
dim x as new GM16ImageMBS(size, backgroundColor)

x.pixelColor(0,0) = new GM16ColorMBS(0 ,0 ,0 ) // black
x.pixelColor(0,1) = new GM16ColorMBS(255,0 ,0 ) // red
x.pixelColor(0,2) = new GM16ColorMBS(0 ,255,0 ) // green
x.pixelColor(0,3) = new GM16ColorMBS(0 ,0 ,255) // blue
x.pixelColor(0,4) = new GM16ColorMBS(255,255,0 ) // yellow
x.pixelColor(0,5) = new GM16ColorMBS(0 ,255,255) // cyan
x.pixelColor(0,6) = new GM16ColorMBS(255,0 ,255) // magenta
```

```
// do the map
i.map(x, false)

// convert result from palette picture to bitmap picture
i.type = i.TrueColorType

// and copy picture to backdrop
Backdrop = i.CopyPicture
```

Notes: Set dither to true in to apply Floyd/Steinberg error diffusion to the image. By default, color reduction chooses an optimal set of colors that best represent the original image. Alternatively, you can choose a particular set of colors from an image file with this option.

4.16.109 matteFloodfill(target as GM16ColorMBS, opacity as UInt32, x as Integer, y as Integer, PaintMethod as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Floodfill designated area with a replacement opacity value.

4.16.110 medianFilter(radius as Double=0.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Filter image by replacing each pixel component with the median color in a circular neighborhood.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.medianFilter(10)

Backdrop=image.CopyPicture
```

4.16.111 minify

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduce image by integral (half) size.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.minify

Backdrop=image.CopyPicture
```

4.16.112 modequalizeifyImage

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Not documented.

4.16.113 modifyImage

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Prepare to update image (copy if reference >1).
Notes: Normally Magick++'s implicit reference counting takes care of all instance management. In the rare case that the automatic instance management does not work, use this method to assure that there is only one reference to the image to be modified. It should be used in the cases where a GraphicsMagick C function is used directly on an image which may have multiple references:

4.16.114 modulate(brightness as Double, saturation as Double, hue as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Modulate percent hue, saturation, and brightness of an image.
Example:

```
dim logo as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(logo)

image.type = image.TrueColorType

// brightness 150%
image.modulate(150,100,100)
backdrop = image.CopyPicture
```

Notes: Modulation of saturation and brightness is as a ratio of the current value (100 for no change). Modulation of hue is an absolute rotation of -180 degrees to +180 degrees from the current position corresponding to an argument range of 0 to 200 (100 for no change).

4.16.115 montageGeometry as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile size and offset within an image montage.
Notes: Only valid for montage images.

4.16.116 motionBlur(radius as Double, sigma as Double, angle as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Motion blur image with specified blur factor.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.motionBlur(30,10,90)

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels. The angle parameter specifies the angle the object appears to be comming from (zero degrees is from the right).

4.16.117 negate(grayscale as boolean=false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negate colors in image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.negate

Backdrop=image.CopyPicture
```

Notes: Set grayscale to only negate grayscale values in image.

4.16.118 normalize

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Normalize image (increase contrast by normalizing the pixel values to span the full range of color values).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.normalize

Backdrop=image.CopyPicture
```

4.16.119 oilPaint(radius as Double=3.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Oilpaint image (image looks like an oil painting).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.oilPaint

Backdrop=image.CopyPicture
```

4.16.120 opacity(opacity as UInt32)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or attenuate the opacity channel in the image.
Notes: If the image pixels are opaque then they are set to the specified opacity value, otherwise they are blended with the supplied opacity value. The value of opacity ranges from 0 (completely opaque) to MaxRGB. The defines OpaqueOpacity and TransparentOpacity are available to specify completely opaque or completely transparent, respectively.

4.16.121 opaque(opaqueColor as GM16ColorMBS, penColor as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Change color of specified opaque pixel to specified pen color.

4.16.122 ping(data as GM16BlobMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads information for an image from the blob.
Notes: Ping is similar to read except only enough of the image is read to determine the image columns, rows, and filesize. Access the columns, rows, and fileSize attributes after invoking ping. The image pixels are not valid after calling ping.
See also:

- 4.16.123 ping(file as folderitem) 244
- 4.16.124 ping(Path as string) 245

4.16.123 ping(file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads information for an image from the file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.jpg”)

// try with Constructor (same as read)
dim t1 as Double = Microseconds
dim g1 as new GM16ImageMBS(f)

// now just ping
dim t2 as Double = Microseconds
dim g2 as new GM16ImageMBS
g2.ping(f)

// or read
dim t3 as Double = Microseconds
dim g3 as new GM16ImageMBS
g3.read(f)

dim t4 as Double = Microseconds

// show speeds
MsgBox str(T4-t3)+” ¬µs for read”+EndOfLine+_
```

```
str(T3-t2)+" ¬µs for ping"+EndOfLine+_
str(T2-t1)+" ¬µs for Constructor"
```

Notes: Ping is similar to read except only enough of the image is read to determine the image columns, rows, and filesize. Access the columns, rows, and fileSize attributes after invoking ping. The image pixels are not valid after calling ping.
See also:

- 4.16.122 ping(data as GM16BlobMBS) 244
- 4.16.124 ping(Path as string) 245

4.16.124 ping(Path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads information for an image from the image specification.
Notes: Ping is similar to read except only enough of the image is read to determine the image columns, rows, and filesize. Access the columns, rows, and fileSize attributes after invoking ping. The image pixels are not valid after calling ping.
See also:

- 4.16.122 ping(data as GM16BlobMBS) 244
- 4.16.123 ping(file as folderitem) 244

4.16.125 PNGLibVersion as string

Plugin Version: 17.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries PNG library version string.

4.16.126 quantize(measureError as boolean=false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Quantize image (reduce number of colors).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.quantize
image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: Set measureError to true in order to calculate error attributes.

4.16.127 QuantumDepth as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the quantum depth.

4.16.128 quantumOperator(channel as Integer, Operator as Integer, rvalue as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply an arithmetic or bitwise operator to the image pixel quantums.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)

const AddQuantumOp = 1
const ThresholdQuantumOp = 10

g.quantumOperator( g.AllChannels, AddQuantumOp, 100)

// show
me.Backdrop = g.CopyPicture
```

See also:

- 4.16.129 quantumOperator(x as Integer, y as Integer, columns as Integer, rows as Integer, channel as Integer, Operator as Integer, rvalue as Double) 246

4.16.129 quantumOperator(x as Integer, y as Integer, columns as Integer, rows as Integer, channel as Integer, Operator as Integer, rvalue as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply an arithmetic or bitwise operator to the image pixel quantums.
See also:

- 4.16.128 quantumOperator(channel as Integer, Operator as Integer, rvalue as Double) 246

4.16.130 raiseGeometryDefault as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The default geometry description for raise.

4.16.131 raiseImage

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Raise image (lighten or darken the edges of an image to give a 3-D raised or lowered effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.raiseImage

Backdrop=image.CopyPicture
```

See also:

- 4.16.132 raiseImage(geometry as GM16GeometryMBS, raisedFlag as boolean=false) 247

4.16.132 raiseImage(geometry as GM16GeometryMBS, raisedFlag as boolean=false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Raise image (lighten or darken the edges of an image to give a 3-D raised or lowered effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.raiseImage(GM16GeometryMBS.Make(5,8))

Backdrop=image.CopyPicture
```

See also:

- 4.16.131 raiseImage 247

4.16.133 randomThreshold(thresholds as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Random threshold image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.randomThreshold(GM16GeometryMBS.make("50x200"))

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: Changes the value of individual pixels based on the intensity of each pixel compared to a random threshold. The result is a low-contrast, two color image. The thresholds argument is a geometry containing LOWxHIGH thresholds. If the string contains 2x2, 3x3, or 4x4, then an ordered dither of order 2, 3, or 4 will be performed instead. If a channel argument is specified then only the specified channel is altered. This is a very fast alternative to 'quantize' based dithering.

4.16.134 randomThresholdChannel(thresholds as GM16GeometryMBS, channel as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Random threshold image channel.
Notes: Changes the value of individual pixels based on the intensity of each pixel compared to a random threshold. The result is a low-contrast, two color image. The thresholds argument is a geometry containing LOWxHIGH thresholds. If the string contains 2x2, 3x3, or 4x4, then an ordered dither of order 2, 3, or 4 will be performed instead. If a channel argument is specified then only the specified channel is altered. This is a very fast alternative to 'quantize' based dithering.

4.16.135 read(blob as GM16BlobMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame from in-memory Blob.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GM16ImageMBS
dim blob as new GM16BlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

See also:

- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size from in-memory Blob.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250

- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size and depth from in-memory Blob.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size, depth, and format from in-memory Blob.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size, and format from in-memory Blob.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.140 read(file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame into current object.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.141 read(path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame into current object.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.142 read(size as GM16GeometryMBS, file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size into current object.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.143 read(size as GM16GeometryMBS, Path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size into current object.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252
- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 254

4.16.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame from an array of raw pixels, with specified storage type (ConstituteImage).
Notes: Returns an Image corresponding to an image stored in a raw memory array format. The pixel data must be in scanline order top-to-bottom. The data can be unsigned char, unsigned short int, unsigned int, unsigned long, float, or double. Float and double require the pixels to be normalized to the range [0..1] , otherwise the range is [0..MaxVal] where MaxVal is the maximum possible value for that type.

Note that for most 32-bit architectures the size of an unsigned long is the same as unsigned int, but for 64-bit architectures observing the LP64 standard, an unsigned long is 64 bits, while an unsigned int remains 32 bits. This should be considered when deciding if the data should be described as "Integer" or "Long".

For example, to create a 640x480 image from unsigned red-green-blue character data, use

image = new GM16ImageMBS(640, 480, "RGB", GM16ImageMBS.StorageTypeCharPixel, pixels);

width: width in pixels of the image.
height: height in pixels of the image.
map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (same as Transparency), O = Opacity, T = Transparency, C = cyan, Y = yellow, M = magenta, K = black, or I = intensity (for grayscale). Specify "P" = pad, to skip over a quantum which is intentionally ignored. Creation of an alpha channel for CMYK images is currently not supported.
type: Define the data type of the pixels. Float and double types are expected to be normalized [0..1] otherwise [0..MaxRGB] . Choose from these types: StorageTypeCharPixel, StorageTypeShortPixel, StorageTypeIntegerPixel, StorageTypeLongPixel, StorageTypeFloatPixel, or StorageTypeDoublePixel.
pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.
See also:

- 4.16.135 read(blob as GM16BlobMBS) 248
- 4.16.136 read(blob as GM16BlobMBS, size as GM16GeometryMBS) 249
- 4.16.137 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer) 250
- 4.16.138 read(blob as GM16BlobMBS, size as GM16GeometryMBS, depth as Integer, magick as string) 250
- 4.16.139 read(blob as GM16BlobMBS, size as GM16GeometryMBS, magick as string) 251
- 4.16.140 read(file as folderitem) 252
- 4.16.141 read(path as string) 252

- 4.16.142 read(size as GM16GeometryMBS, file as folderitem) 253
- 4.16.143 read(size as GM16GeometryMBS, Path as string) 253

4.16.145 reduceNoise

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduce noise in image using a noise peak elimination filter.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.reduceNoise

Backdrop=image.CopyPicture
```

See also:

- 4.16.146 reduceNoise(order as Double) 255

4.16.146 reduceNoise(order as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduce noise in image using a noise peak elimination filter.
See also:

- 4.16.145 reduceNoise 255

4.16.147 ReleaseDate as String

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the release date of the used graphics magick library.
Notes: We update the library only when someone needs an update, so if you need, please contact us.

4.16.148 repage

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image page canvas and position.

4.16.149 resize(geo as GM16GeometryMBS)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image, specifying only geometry, with filter and blur obtained from Image default.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// resize proportionally to fit
dim geo as new GM16GeometryMBS(500,500)
image.resize geo

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Same result as 'zoom' method.
See also:

- 4.16.150 resize(geo as GM16GeometryMBS, filterType as Integer) 256
- 4.16.151 resize(geo as GM16GeometryMBS, filterType as Integer, blur as double) 257

4.16.150 resize(geo as GM16GeometryMBS, filterType as Integer)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image, specifying geometry and filter, with blur using Image default.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// resize proportionally to fit
dim geo as new GM16GeometryMBS(500,500)
image.resize geo, image.CubicFilter

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

See also:

- 4.16.149 resize(geo as GM16GeometryMBS) 256
- 4.16.151 resize(geo as GM16GeometryMBS, filterType as Integer, blur as double) 257

4.16.151 resize(geo as GM16GeometryMBS, filterType as Integer, blur as double)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image, specifying geometry, filter, and blur.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GM16ImageMBS(f)

// resize proportionally to fit
dim geo as new GM16GeometryMBS(500,500)
image.resize geo, image.CubicFilter, 3

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

See also:

- 4.16.149 resize(geo as GM16GeometryMBS) 256
- 4.16.150 resize(geo as GM16GeometryMBS, filterType as Integer) 256

4.16.152 roll(columns as UInt32, rows as UInt32)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll image (rolls image vertically and horizontally) by specified number of columnms and rows).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.roll(30,30)

Backdrop=image.CopyPicture
```

See also:

- 4.16.153 roll(roll as GM16GeometryMBS) 257

4.16.153 roll(roll as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll image (rolls image vertically and horizontally) by specified number of columnms and rows).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.roll(GM16GeometryMBS.Make(0,0,30,30))

Backdrop=image.CopyPicture
```

See also:

- 4.16.152 roll(columns as UInt32, rows as UInt32) 257

4.16.154 rotate(degree as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotate image counter-clockwise by specified number of degrees.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.rotate(30)

Backdrop=image.CopyPicture
```

4.16.155 sample(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image by using pixel sampling algorithm.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.sample GM16GeometryMBS.make(100,100)

Backdrop=image.CopyPicture
```

4.16.156 scale(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image by using simple ratio algorithm which provides good quality.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.scale new GM16GeometryMBS(100,100)

Backdrop=image.CopyPicture
```

4.16.157 segment(clusterThreshold as Double=1.0, smoothingThreshold as Double=1.5)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Segment (coalesce similar image components) by analyzing the histograms of the color components and identifying units that are homogeneous with the fuzzy c-means technique.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.segment

image.type = image.TrueColorType

Backdrop=image.CopyPicture
```

Notes: A histogram is built for the image. This histogram is filtered to reduce noise and a second derivative of the histogram plot is built and used to identify potential cluster colors (peaks in the histogram). The cluster colors are then validated by scanning through all of the pixels to see how many pixels fall within each cluster. Some candidate cluster colors may not match any of the image pixels at all and should be discarded. Specify clusterThreshold, as the number of pixels matching a cluster color in order for the cluster to be considered valid. SmoothingThreshold eliminates noise in the second derivative of the histogram. As the value is increased, you can expect a smoother second derivative. The default is 1.5.

4.16.158 setChromaBluePrimary(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity blue primary point.
Notes: e.g. x=0.15, y=0.06

4.16.159 setchromaGreenPrimary(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity green primary point.
Notes: e.g. x=0.3, y=0.6

4.16.160 setchromaRedPrimary(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity red primary point
Notes: e.g. x=0.64, y=0.33

4.16.161 setchromaWhitePoint(x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity white point
Notes: e.g. x=0.3127, y=0.329

4.16.162 SetEXIFOrientation(orientation as integer) as boolean

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets orientation for EXIF.
Notes: Changing orientation may need to set orientation via SetOrientation and SetEXIFOrientation. With a JPEG you have orientation both in JPEG header and in EXIF metadata.
Returns true for success and false for failure.

For new development, please use ExifTagsMBS class instead.

4.16.163 SetLogEventMask(events as String)

Plugin Version: 21.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set log event mask.
Example:

```
GM16ImageMBS.SetLogEventMask(”coder,annotate”)
```

Notes: Defines which events are logged.
By default logging goes to stderr, so on macOS you may need to run your app via Terminal to see logs there.

List of events includes: none, Configure, Annotate, Render, Transform, Locale, Coder, X11, Cache, Blob, Deprecate, User, Resource, TemporaryFile, Exception, Option, Information, Warning, Error, FatalError and All.

4.16.164 SetPicture(pic as picture, x as Integer, y as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the picture into the Image at the given position.

4.16.165 SetPictureMask(maskpic as picture, x as Integer, y as Integer)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the picture into the Image’s mask at the given position.
Example:

```
// this converts 32 bit PNG with alpha channel to BMP

dim f as FolderItem = SpecialFolder.Desktop.Child(”test.png”)
dim p as Picture = Picture.Open(f)

dim g as new GM16ImageMBS( new GM16GeometryMBS(p.Width, p.Height), new GM16ColorGrayMBS(1.0))

g.type = g.TrueColorMatteType
g.matte = True
g.magick = ”BMP”

g.SetPicture(p, 0, 0)
g.SetPictureMask(p.mask.invertMBS, 0, 0)

f = SpecialFolder.Desktop.Child(”test.bmp”)
g.write(f)
```

4.16.166 setPixels(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Allocates a pixel cache region to store image pixels as defined by the region rectangle.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)

// get pointer to some pixels to write
dim x as ptr = g.setPixels(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
g.syncPixels

// show
me.Backdrop = g.CopyPicture
```

Notes: This area is subsequently transferred from the pixel cache to the image via syncPixels.

4.16.167 setStrokeDashArray(values() as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets stroke dash pattern.
Notes: Specify the pattern of dashes and gaps used to stroke paths. The strokeDashArray represents a zero-terminated array of numbers that specify the lengths of alternating dashes and gaps in pixels. If an odd number of values is provided, then the list of values is repeated to yield an even number of values. A typical strokeDashArray array might contain the members 5 3 2 0, where the zero value indicates the end of the pattern array.

4.16.168 shade(azimuth as Double=30.0, elevation as Double=30.0, colorShading as boolean=false)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shade image using distant light source.
Notes: Specify azimuth and elevation as the position of the light source. By default, the shading results as a grayscale image.. Set colorShading to true to shade the red, green, and blue components of the image.

4.16.169 sharpen(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpen pixels in image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.sharpen

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.16.170 sharpenChannel(channel as Integer, radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpen pixels in image channel.
Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.16.171 shave(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shave pixels from image edges.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.shave(new GM16GeometryMBS(200,200))

Backdrop=image.CopyPicture
```

4.16.172 shear(xShearAngle as Double, yShearAngle as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shear image (create parallelogram by sliding image by X or Y axis).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.shear(10,20)

Backdrop=image.CopyPicture
```

Notes: Shearing slides one edge of an image along the X or Y axis, creating a parallelogram. An X direction shear slides an edge along the X axis, while a Y direction shear slides an edge along the Y axis. The amount of the shear is controlled by a shear angle. For X direction shears, x degrees is measured relative to the Y axis, and similarly, for Y direction shears y degrees is measured relative to the X axis. Empty triangles left over from shearing the image are filled with the color defined as borderColor.

4.16.173 signature(force as boolean=false) as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image textual signature.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

MsgBox image.signature

Backdrop=image.CopyPicture
```

Notes: Set force to true in order to re-calculate the signature regardless of whether the image data has been modified.

4.16.174 solarize(factor as Double=50.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Solarize image (similar to effect seen when exposing a photographic film to light during the development process)
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.solarize

Backdrop=image.CopyPicture
```

4.16.175 spread(amount as UInt32=3)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Spread pixels randomly within image by specified ammount
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.spread 5

Backdrop=image.CopyPicture
```

4.16.176 statistics as GM16ImageStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain image statistics.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim stat as GM16ImageStatisticsMBS = g.statistics
dim gs as GM16ImageChannelStatisticsMBS = stat.blue

MsgBox ”blue channel: ”+str(gs.minimum)+”-”+str(Gs.maximum)+”, mean ”+str(gs.mean)
```

Notes: Statistics are normalized to the range of 0.0 to 1.0 and are output to the specified ImageStatistics structure.

4.16.177 stegano(watermark as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add a digital watermark to the image (based on second image).
Example:

```
dim p as Picture = LogoMBS(500)
dim p1 as Picture = New Picture(550,500,32)
dim p2 as Picture = New Picture(550,500,32)

p1.Graphics.DrawPicture p, 0,0
p2.Graphics.DrawPicture p,50,0

dim image1 as new GM16ImageMBS(p1)
dim image2 as new GM16ImageMBS(p2)

image2.zoom(new GM16GeometryMBS(100,100)) // scale down

// add watermark
image1.stegano(image2)

// now make a threshold so you see the difference
image1.threshold 254

image1.type = image1.TrueColorType
Backdrop=image1.CopyPicture
```

4.16.178 stereo(rightImage as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image which appears in stereo when viewed with red-blue glasses (Red image on left, blue on right)
Example:

```
dim p as Picture = LogoMBS(500)
dim p1 as Picture = New Picture(550,500,32)
dim p2 as Picture = New Picture(550,500,32)

p1.Graphics.DrawPicture p, 0,0
p2.Graphics.DrawPicture p,50,0

dim image1 as new GM16ImageMBS(p1)
dim image2 as new GM16ImageMBS(p2)

image1.stereo(IMAGE2)

Backdrop=image1.CopyPicture
```

4.16.179 strip

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Remove all profiles and text attributes from the image.

4.16.180 strokeDashArray as Double()

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries stroke dash pattern.
Notes: Specify the pattern of dashes and gaps used to stroke paths. The strokeDashArray represents a zero-terminated array of numbers that specify the lengths of alternating dashes and gaps in pixels. If an odd number of values is provided, then the list of values is repeated to yield an even number of values. A typical strokeDashArray array might contain the members 5 3 2 0, where the zero value indicates the end of the pattern array.

4.16.181 swirl(degree as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Swirl image (image pixels are rotated by degrees).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.swirl 200
```

```
Backdrop=image.CopyPicture
```

4.16.182 syncPixels

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers the image cache pixels to the image.

4.16.183 texture(texture as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Channel a texture on pixels matching image background color.

4.16.184 threshold(degree as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Threshold image channels (below threshold becomes black, above threshold becomes white).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.threshold 127

// convert to RGB so CopyPicture works
image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: The range of the threshold parameter is 0 to MaxRGB.

4.16.185 thumbnail(geometry as GM16GeometryMBS)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image using several algorithms to make smaller images very quickly.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)

// make thumbnail
dim geo as new GM16GeometryMBS(100, 100)
g.thumbnail(geo)

// show
me.Backdrop = g.CopyPicture
```

4.16.186 TIFFLibVersion as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries version string for tiff library.

4.16.187 transform(imageGeometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform image based on image and crop geometries.
Notes: Crop geometry is optional.
See also:

- 4.16.188 transform(imageGeometry as GM16GeometryMBS, cropGeometry as GM16GeometryMBS) 269

4.16.188 transform(imageGeometry as GM16GeometryMBS, cropGeometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform image based on image and crop geometries.
Notes: Crop geometry is optional.
See also:

- 4.16.187 transform(imageGeometry as GM16GeometryMBS) 269

4.16.189 transformOrigin(tx as Double, ty as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Origin of coordinate system to use when annotating with text or drawing.

4.16.190 transformReset

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reset transformation parameters to default.

4.16.191 transformRotation(angle as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotation to use when annotating with text or drawing.

4.16.192 transformScale(tx as Double, ty as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scale to use when annotating with text or drawing.

4.16.193 transformSkewX(x as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Skew to use in X axis when annotating with text or drawing.

4.16.194 transformSkewY(y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Skew to use in Y axis when annotating with text or drawing.

4.16.195 transparent(color as GM16ColorMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add matte channel to image, setting pixels matching color to transparent.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

dim c as new GM16ColorMBS(”white”)
image.transparent(c)

Backdrop=image.CombinePictureWithMask
```

4.16.196 trim

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Trim edges that are the background color from the image.
Example:

```
dim p as Picture = LogoMBS(500)
// make the logo picture bigger
dim q as Picture = New Picture(700,700,32)

q.Graphics.DrawPicture p,100,100

dim image as new GM16ImageMBS(q)

// now trim the white border away
image.trim

Backdrop=image.CopyPicture
```

Notes: See ColorFuzz property for how far the pixel value can differentiate.

4.16.197 unregisterId

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Not documented.

4.16.198 unsharpmask(radius as Double, sigma as Double, amount as Double, threshold as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replace image with a sharpened version of the original image using the unsharp mask algorithm.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.unsharpmask(10,1,0.5,50)

Backdrop=image.CopyPicture
```

Notes: radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
amount: the percentage of the difference between the original and the blur image that is added back into the original.
threshold: the threshold in pixels needed to apply the diffence amount.

4.16.199 unsharpmaskChannel(channel as Integer, radius as Double, sigma as Double, amount as Double, threshold as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replace image channel with a sharpened version of the original image using the unsharp mask algorithm.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.unsharpmaskChannel(Image.RedChannel, 10,1,0.5,50)

Backdrop=image.CopyPicture
```

Notes:

4.16.200 wave(amplitude as Double=25.0, wavelength as Double=150.0)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

channel: image channel to modify.
radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
amount: the percentage of the difference between the original and the blur image that is added back into the original.
threshold: the threshold in pixels needed to apply the diffence amount.

Function: Map image pixels to a sine wave.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.wave

Backdrop=image.CopyPicture
```

4.16.201 WebPVersion as String

Plugin Version: 23.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns version of WebP library.
Notes: Should be a version string like "1.3.0".

4.16.202 write(blob as GM16BlobMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to in-memory Blob, with optional format and adjoin parameters.
See also:

- 4.16.203 write(blob as GM16BlobMBS, magick as string) 274
- 4.16.204 write(blob as GM16BlobMBS, magick as string, depth as UInt32) 274
- 4.16.205 write(file as folderitem) 274
- 4.16.206 write(Path as string) 275
- 4.16.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 275

4.16.203 write(blob as GM16BlobMBS, magick as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to in-memory Blob, with optional format and adjoin parameters.
See also:

- 4.16.202 write(blob as GM16BlobMBS) 273
- 4.16.204 write(blob as GM16BlobMBS, magick as string, depth as UInt32) 274
- 4.16.205 write(file as folderitem) 274
- 4.16.206 write(Path as string) 275
- 4.16.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 275

4.16.204 write(blob as GM16BlobMBS, magick as string, depth as UInt32)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to in-memory Blob, with optional format and adjoin parameters.
See also:

- 4.16.202 write(blob as GM16BlobMBS) 273
- 4.16.203 write(blob as GM16BlobMBS, magick as string) 274
- 4.16.205 write(file as folderitem) 274
- 4.16.206 write(Path as string) 275
- 4.16.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 275

4.16.205 write(file as folderitem)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to a file.
Example:

```
// this converts 32 bit PNG with alpha channel to BMP

dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim p as Picture = Picture.Open(f)

dim g as new GM16ImageMBS( new GM16GeometryMBS(p.Width, p.Height), new GM16ColorGrayMBS(1.0))
```

```
g.type = g.TrueColorMatteType
g.matte = True
g.magick = ”BMP”

g.SetPicture(p, 0, 0)
g.SetPictureMask(p.mask.invertMBS, 0, 0)

f = SpecialFolder.Desktop.Child(”test.bmp”)
g.write(f)
```

See also:

- 4.16.202 write(blob as GM16BlobMBS) 273
- 4.16.203 write(blob as GM16BlobMBS, magick as string) 274
- 4.16.204 write(blob as GM16BlobMBS, magick as string, depth as UInt32) 274
- 4.16.206 write(Path as string) 275
- 4.16.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 275

4.16.206 write(Path as string)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to a file.
See also:

- 4.16.202 write(blob as GM16BlobMBS) 273
- 4.16.203 write(blob as GM16BlobMBS, magick as string) 274
- 4.16.204 write(blob as GM16BlobMBS, magick as string, depth as UInt32) 274
- 4.16.205 write(file as folderitem) 274
- 4.16.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 275

4.16.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to an array of pixels with storage type specified by user (DispatchImage).
Notes: e.g. image.write(0, 0, 640, 1, ”RGB”, 0, pixels)
See also:

4.16.208 ZLibVersion as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries version string for zlib library.

4.16.209 zoom(geometry as GM16GeometryMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Zoom (resize) image to specified size.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.zoom(new GM16GeometryMBS(200,200))

Backdrop=image.CopyPicture
```

4.16.210 ZPL(Header as boolean = true) as String

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries image as ZPL hex image.
Notes: This is for sending data to a receipt printer. The picture is taken as black & white image and we build the hex string, you can send to the printer.
Picture needs to have a width dividable by 8.

Set Header to false for skipping header and footer.

4.16.211 Properties

4.16.212 adjoin as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Join images into a single multi-image file.
Notes: (Read and Write property)

4.16.213 animationDelay as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Time in 1/100ths of a second (0 to 65535) which must expire before displaying the next image in an animated sequence.
Notes: This option is useful for regulating the animation of a sequence of GIF images within Netscape.
(Read and Write property)

4.16.214 animationIterations as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of iterations to loop an animation (e.g. Netscape loop extension) for.
Notes: (Read and Write property)

4.16.215 antiAlias as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Control antialiasing of rendered Postscript and Postscript or TrueType fonts.
Notes: Enabled by default.
(Read and Write property)

4.16.216 backgroundColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Example:

```
// make a red and turn it to 100% transparent
Dim red As New GM16ColorRGBMBS("red")
```

```
red.alpha = 1

// now make image and make it RGBA with the transparnet background
dim RastoredVectorImage as New GM16ImageMBS
RastoredVectorImage.type = GM16ImageMBS.TrueColorMatteType
RastoredVectorImage.backgroundColor = red

// now read SVG, so we get a transparent background
dim SVG_File as FolderItem = SpecialFolders.desktop.Child("test.svg")
RastoredVectorImage.read(SVG_File)
```

Notes: (Read and Write property)

4.16.217 backgroundTexture as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image file name to use as the background texture.
Notes: Does not modify image pixels.
(Read and Write property)

4.16.218 baseColumns as UInt32

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base image width (before transformations)
Notes: (Read only property)

4.16.219 baseFilename as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base image filename (before transformations)
Notes: (Read only property)

4.16.220 baseRows as Uint32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base image height (before transformations).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

Title = str(image.baseRows)+" x "+str(image.baseColumns)
```

Notes: (Read only property)

4.16.221 borderColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

4.16.222 boundingBox as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Return smallest bounding box enclosing non-border pixels.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.fillColor = new GM16ColorRGBMBS("red") // set color
image.strokeColor = new GM16ColorRGBMBS("green") // set color

image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle

draw.Circle(250, 250, 120, 150)
draw.Draw

draw = nil
image.type = image.TrueColorType

Backdrop = image.CopyPicture
```

MsgBox image.boundingBox.StringValue

Notes: The current fuzz value is used when discriminating between pixels. This is the crop bounding box used by crop(Geometry(0,0)).
(Read only property)

4.16.223 boxColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base color that annotation text is rendered on (default none).
Notes: (Read and Write property)

4.16.224 classType as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image class (DirectClass or PseudoClass).
Notes: NOTE: setting a DirectClass image to PseudoClass will result in the loss of color information if the number of colors in the image is greater than the maximum palette size (either 256 or 65536 entries depending on the value of QuantumDepth when ImageMagick was built):
(Read and Write property)

4.16.225 clipMask as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Associate a clip mask image with the current image.
Notes: The clip mask image must have the same dimensions as the current image or an exception is thrown. Clipping occurs wherever pixels are transparent in the clip mask image. Clipping Pass an invalid image to unset an existing clip mask.
(Read and Write property)

4.16.226 colorFuzz as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this

option to match colors that are close to the target color in RGB space.

e.g. set to 50 for 8 bit class and 50 * 257 for the 16 bit class to allow 20% divagation in pixel values.
(Read and Write property)

4.16.227 colorMapSize as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of entries in the colormap.
Notes: Setting the colormap size may extend or truncate the colormap. The maximum number of supported entries is specified by the MaxColormapSize constant, and is dependent on the value of QuantumDepth when GraphicsMagick is compiled. An exception is thrown if more entries are requested than may be supported. Care should be taken when truncating the colormap to ensure that the image colormap indexes reference valid colormap entries.
(Read and Write property)

4.16.228 colorSpace as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The colorspace (e.g. CMYK) used to represent the image pixel colors.
Notes:

UndefinedColorspace	= 0
RGBColorspace	= 1 (Plain old RGB colorspace)
GRAYColorspace	= 2 (Plain old full-range grayscale)
TransparentColorspace	= 3 (RGB but preserve matte channel during quantize)
OHTAColorspace	= 4
XYZColorspace	= 5 (CIE XYZ)
YCCColorspace	= 6 (Kodak PhotoCD PhotoYCC)
YIQColorspace	= 7
YPbPrColorspace	= 8
YUVColorspace	= 9
CMYKColorspace	= 10 (Cyan, magenta, yellow, black, alpha)
sRGBColorspace	= 11 (Kodak PhotoCD sRGB)
HSLColorspace	= 12 (Hue, saturation, luminosity)
HWBColorspace	= 13 (Hue, whiteness, blackness)
LABColorspace	= 14 (LAB colorspace not supported yet other than via lcms)
CineonLogRGBColorspace	= 15 (RGB data with Cineon Log scaling, 2.048 density range)
Rec601LumaColorspace	= 16 (Luma (Y) according to ITU-R 601)
Rec601YCbCrColorspace	= 17 (YCbCr according to ITU-R 601)
Rec709LumaColorspace	= 18 (Luma (Y) according to ITU-R 709)
Rec709YCbCrColorspace	= 19 (YCbCr according to ITU-R 709)

(Read and Write property)

4.16.229 columns as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image width.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

Title = str(image.columns)+" x "+str(image.rows)
Backdrop=image.CopyPicture
```

Notes: (Read only property)

4.16.230 comment as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image comment.
Notes: When you set this property, you add comment string to image.
By default, each image is commented with its file name. Use this method to assign a specific comment to the image. Optionally you can include the image filename, type, width, height, or other image attributes by embedding special format characters:
(Read and Write property)

4.16.231 compose as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composition operator to be used when composition is implicitly used (such as for image flattening).
Notes: (Read and Write property)

4.16.232 compressType as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.
Notes: The default is the compression type of the input image file.
(Read and Write property)

4.16.233 debug as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enable printing of debug messages from GraphicsMagick as it executes.
Notes: (Read and Write property)

4.16.234 density as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Vertical and horizontal resolution in pixels of the image.
Example:

```
dim p as new GM16ImageMBS

dim item as FolderItem = SpecialFolder.Desktop.Child("input.png")
p.read(item)
p.scale new GM16GeometryMBS(3750,3750)
p.quality = 95
p.resolutionUnits = p.PixelsPerInchResolution
p.density = new GM16GeometryMBS(300, 300)
dim out as FolderItem = SpecialFolder.Desktop.Child("output.png")
p.write out
```

Notes: This option specifies an image density when decoding a Postscript or Portable Document page. Often used with psPageSize.
(Read and Write property)

4.16.235 depth as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (bits allocated to red/green/blue components).
Notes: Used to specify the bit depth when reading or writing raw images or when the output format supports multiple depths. Defaults to the quantum depth that GraphicsMagick is compiled with.
(Read and Write property)

4.16.236 directory as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile names from within an image montage.
Notes: (Read only property)

4.16.237 endian as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian mode.
Notes: Endianness (LSBEndian like Intel, MSBEndian like SPARC, or NativeEndian for what this computer uses) for image formats which support endian-specific options.
(Read and Write property)

4.16.238 ExifThumbnail as String

Plugin Version: 20.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extracts an embedded thumbnail in EXIF data.
Example:

```
Dim g As New GM16ImageMBS

// not load, but just read header & metadata
g.ping("/Users/cs/Desktop/test.JPG")

// get thumbnail
Dim Thumbnail As String = g.ExifThumbnail

// show it
window1.Backdrop = picture.FromData(Thumbnail)
```

Notes: Returns string containing JPEG compressed image data.

For new development, please use ExifTagsMBS class instead.
(Read only property)

4.16.239 fileName as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image file name.
Notes: (Read and Write property)

4.16.240 fileSize as Int64

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of bytes of the image on disk.
Notes: (Read only property)

4.16.241 fillColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color to use when filling drawn objects.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.16.242 fillPattern as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pattern to use while filling drawn objects.
Notes: (Read and Write property)

4.16.243 fillRule as Integer

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rule to use when filling drawn objects
Notes: (Read and Write property)

4.16.244 filterType as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The reduction filter employed has a sigificant effect on the time required to resize an image and the resulting quality. The default filter is Lanczos which has been shown to produce high quality results when reducing most images.
Notes: Filter to use when resizing image.
(Read and Write property)

4.16.245 font as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font.
Notes: If the font is a fully qualified X server font name, the font is obtained from an X server. To use a TrueType font, precede the TrueType filename with an @. Otherwise, specify a Postscript font name (e.g. ”helvetica”).
(Read and Write property)

4.16.246 FontFamily as String

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font family to use.
Notes: The plugin will look through the list to find best match for combination of family, style, stretch and weight.
You can use either FontFamily or Font property, but not both.
Setting font family clears font.
(Read and Write property)

4.16.247 fontPointsize as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font point size.
Notes: (Read and Write property)

4.16.248 FontStretch as Integer

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font stretch.
Notes: Can be Normal, UltraCondensed, ExtraCondensed, Condensed, SemiCondensed, SemiExpanded, Expanded, ExtraExpanded, UltraExpanded or Any.
See stretch constants.
(Read and Write property)

4.16.249 FontStyle as Integer

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font style to use.
Notes: Can be Normal, Italic, Oblique or Any.
See font style constants.
(Read and Write property)

4.16.250 FontWeight as Integer

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font weight.
Notes: The font weight in range from 0 to 1000.
400 is normal and 800 bold.
(Read and Write property)

4.16.251 format as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Long image format description.
Notes: (Read only property)

4.16.252 gamma as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma correct the image or individual image channels.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.gamma = 3.0

Backdrop=image.CopyPicture
```

Notes: If you get the value, it is the gamma level of the image. Gamma is a pow() function which converts between the linear light representation and the representation for the computer display. Most computer images are gamma corrected to 2.2 (1/0.4545) so that each step results in a visually linear step on a computer or video display:
(Read and Write property)
See also:

- 4.16.85 gamma(gammaRed as Double, gammaGreen as Double, gammaBlue as Double) 232

4.16.253 geometry as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size of the image when encoding.
Notes: (Read only property)

4.16.254 getConstIndexes as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain immutable image pixel indexes (valid for PseudoClass images)
Notes: (Read only property)

4.16.255 getIndexes as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain mutable image pixel indexes (valid for PseudoClass images)
Notes: (Read only property)

4.16.256 gifDisposeMethod as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GIF disposal method.
Notes: This option (specific to the GIF file format) is used to control how successive frames are rendered (how the preceding frame is disposed of) when creating a GIF animation.

Constant	Disposal	Description
UndefinedDispose	0	No disposal specified.
NoneDispose	1	Do not dispose between frames.
BackgroundDispose	2	Overwrite frame with background color from header.
PreviousDispose	3	Overwrite with previous frame.

(Read and Write property)

4.16.257 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Example:

```
dim c as new GM16ColorMBS(”white”)
dim g as new GM16GeometryMBS(100,100)
dim image as new GM16ImageMBS(g, c)
MsgBox hex(Image.handle) // valid if not zero
```

Notes: (Read and Write property)

4.16.258 height as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the image.
Example:

```
dim c as new GM16ColorRGBMBS(1.0,0.0,0.0)
dim size as new GM16GeometryMBS(100,100)
dim g as new GM16ImageMBS(size, c)
```

```
MsgBox str(g.width)+" "+str(g.height)
```

Notes: This is a convenience function for you which calls size.height.
(Read only property)

4.16.259 iccColorProfile as GM16BlobMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ICC color profile.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("IMG_0793.tif")
dim Image as new GM16ImageMBS(f)
dim ProfileBlob as GM16BlobMBS = Image.iccColorProfile
dim ProfileData as string = ProfileBlob.CopyString
dim cm as LCMS2ProfileMBS = LCMS2ProfileMBS.OpenProfileFromString(ProfileData)
dim name as string = cm.Name

Break // check data in debugger
```

Notes: Supplied via a Blob since Magick++/ and GraphicsMagick do not currently support formating this data structure directly. Specifications are available from the International Color Consortium for the format of ICC color profiles.
(Read and Write property)

4.16.260 interlaceType as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.
(Read and Write property)

4.16.261 iptcProfile as GM16BlobMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: IPTC profile.
Notes: Supplied via a Blob since Magick++ and GraphicsMagick do not currently support formating this data structure directly. Specifications are available from the International Press Telecommunications Council for IPTC profiles.
(Read and Write property)

4.16.262 isValid as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Does object contain valid image?
Notes: Set to false in order to invalidate the image. Images constructed via the default constructor are invalid images and isValid() will return false.
(Read and Write property)

4.16.263 label as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image label.
Notes: (Read only property)
See also:

- 4.16.101 label(text as string) 237

4.16.264 lineWidth as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Stroke width for drawing vector objects (default one)
Notes: This method is now deprecated. Please use strokeWidth instead.
(Read and Write property)

4.16.265 magick as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The name of the codec to use for compression.
Example:

```
// this converts 32 bit PNG with alpha channel to BMP

dim f as FolderItem = SpecialFolder.Desktop.Child(”test.png”)
dim p as Picture = Picture.Open(f)

dim g as new GM16ImageMBS( new GM16GeometryMBS(p.Width, p.Height), new GM16ColorGrayMBS(1.0))

g.type = g.TrueColorMatteType
g.matte = True
g.magick = ”BMP”

g.SetPicture(p, 0, 0)
g.SetPictureMask(p.mask.invertMBS, 0, 0)

f = SpecialFolder.Desktop.Child(”test.bmp”)
g.write(f)
```

Notes: (Read and Write property)

4.16.266 matte as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image supports transparency (matte channel)
Notes: (Read and Write property)

4.16.267 matteColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (frame) color.
Notes: (Read and Write property)

4.16.268 meanErrorPerPixel as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mean error per pixel computed when an image is color reduced.
Notes: This parameter is only valid if verbose is set to true and the image has just been quantized.
(Read only property)

4.16.269 modulusDepth as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image modulus depth (minimum number of bits required to support red/green/blue components without loss of accuracy).
Notes: The pixel modulus depth may be decreased by supplying a value which is less than the current value, updating the pixels (reducing accuracy) to the new depth. The pixel modulus depth can not be increased over the current value using this method.
(Read and Write property)

4.16.270 monochrome as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform image to black and white while color reducing (quantizing).
Notes: (Read and Write property)

4.16.271 normalizedMaxError as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The normalized max error per pixel computed when an image is color reduced.
Notes: This parameter is only valid if verbose is set to true and the image has just been quantized.
(Read only property)

4.16.272 normalizedMeanError as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The normalized mean error per pixel computed when an image is color reduced.
Notes: This parameter is only valid if verbose is set to true and the image has just been quantized.
(Read only property)

4.16.273 orientation as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image orientation. Supported by some file formats such as DPX and TIFF. Useful for turning the right way up.
Notes: (Read and Write property)

4.16.274 page as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size and location of an image canvas.
Notes: Use this option to specify the dimensions and position of the Postscript page in dots per inch or a TEXT page in pixels. This option is typically used in concert with density .

Page may also be used to position a GIF image (such as for a scene in an animation).
(Read and Write property)

4.16.275 penColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The pen color.
Notes: (Read and Write property)

4.16.276 quality as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level (default 75).
Notes: (Read and Write property)

4.16.277 quantizeColors as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Maximum number of colors to quantize to.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.quantizeColors = 10
image.quantize

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.16.278 quantizeColorSpace as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorspace to quantize in (default RGB).
Example:

```
// load a picture
dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim pic as Picture = Picture.Open(f)

const GrayColorSpace = 2

Dim Converter As New GM16ImageMBS(Pic)

// quantize with dither
Converter.type = GM16ImageMBS.BilevelType
Converter.quantizeColorSpace = GrayColorSpace
Converter.quantizeColors = 2
Converter.quantizeDither = True
Converter.quantize

// convert back to Xojo
Converter.type = GM16ImageMBS.TrueColorType
Backdrop = Converter.CopyPicture
```

Notes: Empirical evidence suggests that distances in color spaces such as YUV or YIQ correspond to perceptual color differences more closely than do distances in RGB space. These color spaces may give better results when color reducing an image.
(Read and Write property)

4.16.279 quantizeDither as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply Floyd/Steinberg error diffusion to the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.quantizeColors = 10
image.quantizeDither = true
image.quantize

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: The basic strategy of dithering is to trade intensity resolution for spatial resolution by averaging the intensities of several neighboring pixels. Images which suffer from severe contouring when reducing colors can be improved with this option. The quantizeColors or monochrome option must be set for this option to take effect.
(Read and Write property)

4.16.280 quantizeTreeDepth as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Depth of the quantization color classification tree.
Notes: Values of 0 or 1 allow selection of the optimal tree depth for the color reduction algorithm. Values between 2 and 8 may be used to manually adjust the tree depth.
(Read and Write property)

4.16.281 Quiet as Boolean

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Suppress all warning messages.
Notes: Error messages are still reported.
(Read and Write property)

4.16.282 renderingIntent as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of rendering intent (used when applying an ICC color profile).
Notes: (Read and Write property)

4.16.283 resolutionUnits as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Example:

```
Dim item As FolderItem = SpecialFolder.Desktop.Child("test.jpeg")
Dim p As New GM16ImageMBS(item)

// scale image
p.quality = 95
p.scale New GM16GeometryMBS(1000,1000)

// change resolution
p.density = New GM16GeometryMBS(300, 300)
p.resolutionUnits = p.PixelsPerInchResolution

// remove metadata
Dim empty As New GM16BlobMBS
p.profile("EXIF") = empty
p.profile("IPTC") = empty
p.profile("XMP") = empty

Dim out As FolderItem = SpecialFolder.Desktop.Child("output.jpeg")
p.write out
```

Notes: (Read and Write property)

4.16.284 rows as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of pixel rows in the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

Title = str(image.columns)+" x "+str(image.rows)
Backdrop=image.CopyPicture
```

Notes: (Read only property)

4.16.285 scene as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image scene number.
Notes: (Read and Write property)

4.16.286 size as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height of a raw image (an image which does not support width and height information).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

MsgBox image.size.StringValue

image.size = new GM16GeometryMBS(200,200)

Backdrop=image.CopyPicture
```

Notes: Size may also be used to affect the image size read from a multi-resolution format (e.g. Photo CD, JBIG, or JPEG.
(Read and Write property)

4.16.287 strokeAntiAlias as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enable/disable stroke anti-aliasing.
Notes: (Read and Write property)

4.16.288 strokeColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color to use when drawing object outlines.
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS(”white”) // white
dim image as new GM16ImageMBS(g, c)
```

```
image.strokeColor = new GM16ColorRGBMBS(”red”) // Outline color
image.fillColor = new GM16ColorRGBMBS(”green”) // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.16.289 strokeDashOffset as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: While drawing using a dash pattern, specify distance into the dash pattern to start the dash (default 0).
Notes: (Read and Write property)

4.16.290 strokeLineCap as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the end of open subpaths when they are stroked. #
Notes: Values of LineCap are UndefinedCap, ButtCap, RoundCap, and SquareCap.
(Read and Write property)

4.16.291 strokeLineJoin as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the corners of paths (or other vector shapes) when they are stroked. Values of LineJoin are UndefinedJoin, MiterJoin, RoundJoin, and BevelJoin.
Notes: (Read and Write property)

4.16.292 strokeMiterLimit as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify miter limit.
Notes: When two line segments meet at a sharp angle and miter joins have been specified for 'lineJoin', it is possible for the miter to extend far beyond the thickness of the line stroking the path. The miterLimit' imposes a limit on the ratio of the miter length to the 'lineWidth'. The default value of this parameter is 4. (Read and Write property)

4.16.293 strokePattern as GM16ImageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pattern image to use while stroking object outlines.
Notes: (Read and Write property)

4.16.294 strokeWidth as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Stroke width for drawing vector objects (default one).
Example:

```
dim g as new GM16GeometryMBS(500,500)
dim c as new GM16ColorRGBMBS("white") // white
dim image as new GM16ImageMBS(g, c)

image.strokeColor = new GM16ColorRGBMBS("red") // Outline color
image.fillColor = new GM16ColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GM16GraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.16.295 subImage as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Subimage of an image sequence.
Notes: (Read and Write property)

4.16.296 subRange as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of images relative to the base image.
Notes: (Read and Write property)

4.16.297 textEncoding as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotation text encoding (e.g. "UTF-16").
Notes: (Read and Write property)

4.16.298 tileName as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile name.
Notes: (Read and Write property)

4.16.299 totalColors as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of colors in the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

Title = str(image.totalColors) // shows 5284
Backdrop=image.CombinePictureWithMask
```

Notes: (Read only property)

4.16.300 type as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of this image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)

image.type = image.GrayscaleType

Backdrop=image.CopyPicture
```

Notes: You can set this value to convert the image to the type.

Convert the image representation to the specified type or retrieve the current image type. If the image is reduced to an inferior type, then image information may be lost (e.g. color changed to grayscale).

Available enumerations for the type parameter:

BilevelType	1	black/white
GrayscaleType	2	grayscale
GrayscaleMatteType	3	grayscale with alpha (opacity) channel
PaletteType	4	colormapped
PaletteMatteType	5	colormapped with transparency
TrueColorType	6	true (full) color
TrueColorMatteType	7	true (full) color with alpha (opacity) channel
ColorSeparationType	8	Cyan, magenta, yellow, and black
ColorSeparationMatteType	9	Cyan, magenta, yellow, and black with alpha (opacity) channel
OptimizeType	10	Optimize the image type to best represent the existing pixels

(Read and Write property)

4.16.301 verbose as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Print detailed information about the image.

Notes: (Read and Write property)

4.16.302 view as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: FlashPix viewing parameters.
Notes: (Read and Write property)

4.16.303 width as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the image.
Example:

```
dim c as new GM16ColorRGBMBS(1.0,0.0,0.0)
dim size as new GM16GeometryMBS(100,100)
dim g as new GM16ImageMBS(size, c)

MsgBox str(g.width)+" "+str(g.height)
```

Notes: This is a convenience function for you which calls size.width.
(Read only property)

4.16.304 x11Display as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: X11 display to display to, obtain fonts from, or to capture image from.
Notes: (Read and Write property)

4.16.305 XResolution as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: x resolution of the image.
Notes: See also density functions.
Settable with version 19.0.
(Read and Write property)

4.16.306 YResolution as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: y resolution of the image.
Notes: Settable with version 19.0.
(Read and Write property)

4.16.307 attributeValue(name as string) as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Access an arbitrary named image attribute.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("IMG_4048.jpg")
dim g as new GM16ImageMBS(f)
dim a as string = g.attributeValue("EXIF:DateTime")
MsgBox a
```

Notes: Any number of named attributes may be attached to the image. For example, the image comment is a named image attribute with the name "comment". EXIF tags are attached to the image as named attributes. Use the syntax "EXIF:<tag>" to request an EXIF tag similar to "EXIF:DateTime":
(Read and Write computed property)

4.16.308 channelDepth(channel as Integer) as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or obtain modulus channel depth.
Notes: (Read and Write computed property)

4.16.309 colorMap(index as UInt32) as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color at colormap position index.
Notes: (Read and Write computed property)
See also:

- 4.16.26 colorMap as GM16ColorMBS() 205

4.16.310 defineSet(magick as string, key as string) as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or obtain a definition flag to applied when encoding or decoding the specified format.
Notes: Similar to the defineValue() method except that passing the flag value 'true' creates a value-less define with that format and key. Passing the flag value 'false' removes any existing matching definition. The method returns 'true' if a matching key exists, and 'false' if no matching key exists.
(Read and Write computed property)

4.16.311 defineValue(magick as string, key as string) as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or obtain a definition string to applied when encoding or decoding the specified format.
Notes: The meanings of the definitions are format specific. The format is designated by the magick argument, the format-specific key is designated by key, and the associated value is specified by value. See the defineSet() method if the key must be removed entirely.
(Read and Write computed property)

4.16.312 pixelColor(x as UInt32, y as UInt32) as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Get/set pixel color at location x & y.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GM16ImageMBS(p)
dim c as new GM16ColorMBS("red")

for x as Integer = 240 to 260
image.pixelColor(x,250)=c
next

for y as Integer = 240 to 260
image.pixelColor(250,y)=c
next

Backdrop=image.CopyPicture
```

Notes: (Read and Write computed property)

4.16.313 profile(name as string) as GM16BlobMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Get or set a named profile.
Example:

```
Dim item As FolderItem = SpecialFolder.Desktop.Child("test.jpeg")
Dim p As New GM16ImageMBS(item)

// scale image
p.quality = 95
p.scale New GM16GeometryMBS(1000,1000)

// change resolution
p.density = New GM16GeometryMBS(300, 300)
p.resolutionUnits = p.PixelsPerInchResolution

// remove metadata
Dim empty As New GM16BlobMBS
p.profile("EXIF") = empty
p.profile("IPTC") = empty
p.profile("XMP") = empty

Dim out As FolderItem = SpecialFolder.Desktop.Child("output.jpeg")
p.write out
```

Notes: Add or remove a named profile to/from the image. Remove the profile by passing an empty Blob (e.g. Blob()). Valid names are "*", "8BIM", "ICM", "IPTC", or a user/format-defined profile name.

Retrieve a named profile from the image. Valid names are: "8BIM", "8BIMTEXT", "APP1", "APP1JPEG", "ICC", "ICM", & "IPTC" or an existing user/format-defined profile name
(Read and Write computed property)

4.16.314 Constants

Constants

Constant	Value	Description
AbsoluteIntent	3	One of the intent type constants.
AddCompositeOp	8	One of the composite type constants.
AllChannels	10	One of the possible channel constants.
AllCompliance	&hffff	One of the Compliance type constants.
AssociatedAlpha	1	One of the possible alpha type constants.
AtopCompositeOp	4	One of the composite type constants.
BackgroundDispose	2	One of the gif dispose type constants.
BesselFilter	14	One of the filter type constants.
BilevelType	1	One of the image type constants.
BlackChannel	8	One of the possible channel constants.
BlackmanFilter	7	One of the filter type constants.
BlueChannel	5	One of the possible channel constants.
BottomLeftOrientation	4	One of the orientation type constants. Line direction: Left to right Frame Direction: Bottom to top
BottomRightOrientation	3	One of the orientation type constants. Line direction: Right to left Frame Direction: Bottom to top
BoxFilter	2	One of the filter type constants.
BumpmapCompositeOp	12	One of the composite type constants.
BZipCompression	2	One of the compression type constants.
CatromFilter	11	One of the filter type constants.
CenterGravity	5	One of the possible gravity constants.
ClearCompositeOp	18	One of the composite type constants.
ColorizeCompositeOp	28	One of the composite type constants.
ColorSeparationMatteType	9	One of the image type constants.
ColorSeparationType	8	One of the image type constants.
ConcatenateMode	3	One of the image type constants.
CopyBlackCompositeOp	35	One of the composite type constants.
CopyBlueCompositeOp	16	One of the composite type constants.
CopyCompositeOp	13	One of the composite type constants.
CopyCyanCompositeOp	32	One of the composite type constants.
CopyGreenCompositeOp	15	One of the composite type constants.
CopyMagentaCompositeOp	33	One of the composite type constants.
CopyOpacityCompositeOp	17	One of the composite type constants.
CopyRedCompositeOp	14	One of the composite type constants.
CopyYellowCompositeOp	34	One of the composite type constants.
CubicFilter	10	One of the filter type constants.
CyanChannel	2	One of the possible channel constants.
DarkenCompositeOp	24	One of the composite type constants.
DifferenceCompositeOp	10	One of the composite type constants.
DirectClass	1	One of the class type constants.
DisplaceCompositeOp	20	One of the composite type constants.
DissolveCompositeOp	19	One of the composite type constants.
DivideCompositeOp	36	One of the composite type constants.
EastGravity	6	One of the possible gravity constants.
FaxCompression	3	One of the compression type constants.
ForgetGravity	0	One of the possible gravity constants.
FrameMode	1	One of the mode type constants.
GaussianFilter	8	One of the filter type constants.
GaussianNoise	1	One of the possible noise constants.
GrayChannel	11	One of the possible channel constants.
GrayscaleMatteType	3	One of the image type constants.
GrayscaleType	2	One of the image type constants.
GreenChannel	3	One of the possible channel constants.
Group4Compression	4	One of the compression type constants.
HammingFilter	6	One of the filter type constants.
HanningFilter	5	One of the filter type constants.

Font Stretch

Constant	Value	Description
AnyStretch	9	Don't care.
CondensedStretch	3	Condensed
ExpandedStretch	6	Expanded
ExtraCondensedStretch	2	Extra Condensed
ExtraExpandedStretch	7	Extra Expanded
NormalStretch	0	Normal (Default)
SemiCondensedStretch	4	Semi Condensed
SemiExpandedStretch	5	Semi Expanded
UltraCondensedStretch	1	Ultra Condensed
UltraExpandedStretch	8	Ultra Expanded

Font Style

Constant	Value	Description
AnyStyle	3	Don't care.
ItalicStyle	1	Italic font.
NormalStyle	0	Normal (Default)
ObliqueStyle	2	Oblique font.

Storage Types

Constant	Value	Description
StorageTypeCharPixel	0	8bit numbers.
StorageTypeDoublePixel	5	64bit floating numbers.
StorageTypeFloatPixel	4	32bit floating numbers.
StorageTypeIntegerPixel	2	32bit numbers.
StorageTypeLongPixel	3	64bit numbers.
StorageTypeShortPixel	1	16bit numbers.

4.17 class GM16ImageStatisticsMBS

4.17.1 class GM16ImageStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for image statistics.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.17.2 Methods

4.17.3 Constructor

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The private constructor.

4.17.4 Properties

4.17.5 blue as GM16ImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue channel statistics.
Notes: (Read only property)

4.17.6 green as GM16ImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green channel statistics.
Notes: (Read only property)

4.17.7 opacity as GM16ImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity channel statistics.
Notes: (Read only property)

4.17.8 red as GM16ImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red channel statistics.
Notes: (Read only property)

4.18 class GM16LockMBS

4.18.1 class GM16LockMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for locking a certain resource.
Notes: The idea is to pass the constructor a mutexlock and keep the only reference to this new lock object on the stack. On the end of the method, the destructor is called by Xojo and releases the mutexlock automatically.

4.18.2 Methods

4.18.3 Constructor(mutexlock as GM16MutexLockMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new Lock based on the given mutexlock.

4.18.4 Properties

4.18.5 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.18.6 target as GM16MutexLockMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mutexlock this lock is referencing to.
Notes: (Read and Write property)

4.19 class GM16MontageFramedMBS

4.19.1 class GM16MontageFramedMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: MontageFramed provides the means to specify montage options when it is desired to have decorative frames around the image thumbnails.
Notes: MontageFramed inherits from Montage and therefore provides all the methods of Montage as well as those shown in the table ”MontageFramed Methods”.

Framed thumbnails consist of four components: the thumbnail image, the thumbnail frame, the thumbnail border, an optional thumbnail shadow, and an optional thumbnail label area.
Subclass of the GM16MontageMBS class.

4.19.2 Methods

4.19.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.19.4 Properties

4.19.5 borderColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the background color within the thumbnail frame.
Notes: (Read and Write computed property)

4.19.6 borderWidth as Uint32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the border (in pixels) to place between a thumbnail and its surrounding frame.
Notes: This option only takes effect if thumbnail frames are enabled (via frameGeometry) and the thumbnail geometry specification doesn’t also specify the thumbnail border width.
(Read and Write computed property)

4.19.7 frameGeometry as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the geometry specification for frame to place around thumbnail.
Notes: If this parameter is not specified, then the montage is unframed.
(Read and Write computed property)

4.19.8 matteColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail frame color.
Notes: (Read and Write computed property)

4.20 class GM16MontageMBS

4.20.1 class GM16MontageMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Montage is the base class to provide montage options and provides methods to set all options required to render simple (unframed) montages.
Example:

```
// build montage
dim StackingMontage as New GM16MontageMBS
StackingMontage.backgroundColor = New GM16ColorMBS(&cE7E7E7)
StackingMontage.fillColor = New GM16ColorMBS(&c000000)
StackingMontage.tile = New GM16GeometryMBS("1x20")
StackingMontage.geometry = New GM16GeometryMBS("160x120+5+5")
StackingMontage.font = "Helvetica"
StackingMontage.pointSize = 12
StackingMontage.title = "Title goes here"

// make picture
dim logo as Picture = LogoMBS(500)
dim image as New GM16ImageMBS(logo)

image.label("Sample label")

// Put the current image into the array
Dim StackingFrames As new GM16ImageArrayMBS
StackingFrames.insert(image)

// show result
dim resultImages as GM16ImageArrayMBS = StackingFrames.montageImages(StackingMontage)
Backdrop = resultImages.Image(0).CopyPicture
```

Notes: See GM16MontageFramedMBS if you would like to create a framed montage.

Unframed thumbnails consist of four components: the thumbnail image, the thumbnail border, an optional thumbnail shadow, and an optional thumbnail label area.

4.20.2 Methods

4.20.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.20.4 Properties

4.20.5 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.20.6 backgroundColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the background color that thumbnails are imaged upon.
Notes: (Read and Write computed property)

4.20.7 compose as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the image composition algorithm for thumbnails.
Notes: This controls the algorithm by which the thumbnail image is placed on the background. Use of OverCompositeOp is recommended for use with images that have transparency. This option may have negative side-effects for images without transparency.
(Read and Write computed property)

4.20.8 fileName as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the image filename to be used for the generated montage images.
Notes: To handle the case were multiple montage images are generated, a printf-style format may be embedded within the filename. For example, a filename specification of image%02d.miff names the montage

images as image00.miff, image01.miff, etc.
(Read and Write computed property)

4.20.9 fillColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the fill color to use for the label text.
Notes: (Read and Write computed property)

4.20.10 font as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail label font.
Notes: (Read and Write computed property)

4.20.11 geometry as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the size of the generated thumbnail.
Notes: (Read and Write computed property)

4.20.12 gravity as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail positioning within the specified geometry area.
Notes: If the thumbnail is smaller in any dimension than the geometry, then it is placed according to this specification.
See Gravity constants in GM16ImageMBS class.
(Read and Write computed property)

4.20.13 label as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the format used for the image label.
Notes: Special format characters may be embedded in the format string to include information about the

image.
(Read and Write computed property)

4.20.14 penColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the pen color to use for the label text (same as fill).
Notes: (Read and Write computed property)

4.20.15 pointSize as UInt32

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail label font size.
Notes: (Read and Write computed property)

4.20.16 shadow as boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enable/disable drop-shadow on thumbnails.
Notes: (Read and Write computed property)

4.20.17 strokeColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the stroke color to use for the label text.
Notes: (Read and Write computed property)

4.20.18 texture as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies a texture image to use as montage background.
Notes: The built-in textures "granite:" and "plasma:" are available. A texture is the same as a background image.
(Read and Write computed property)

4.20.19 tile as GM16GeometryMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the maximum number of montage columns and rows in the montage.
Notes: The montage is built by filling out all cells in a row before advancing to the next row. Once the montage has reached the maximum number of columns and rows, a new montage image is started.
(Read and Write computed property)

4.20.20 title as string

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the montage title.
Notes: (Read and Write computed property)

4.20.21 transparentColor as GM16ColorMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies a montage color to set transparent.
Notes: This option can be set the same as the background color in order for the thumbnails to appear without a background when rendered on an HTML page. For best effect, ensure that the transparent color selected does not occur in the rendered thumbnail colors.
(Read and Write computed property)

4.21 class GM16MutexLockMBS

4.21.1 class GM16MutexLockMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mutex class for GraphicsMagick.

4.21.2 Methods

4.21.3 lock

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Locks the lock.
Notes: Only one thread at a time can get the lock. The other threads will wait when lock is called.

4.21.4 unlock

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unlocks the lock.

4.21.5 Properties

4.21.6 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.22 class GM16NotInitializedExceptionMBS

4.22.1 class GM16NotInitializedExceptionMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception raised if you access a method/property in an object and the object was not initialized propertly.
Notes: Check the message property for details.
Subclass of the GM16ErrorExceptionMBS class.

4.23 class GM16PathArgsMBS

4.23.1 class GM16PathArgsMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: This is a class for arguments to the path arc/curve methods in GM16GraphicsMBS.
Example:

```
dim g as new GM16PathArgsMBS(1,2,3,4) // for a QuadraticCurveto

MsgBox str(g.x1)+EndOfLine+str(g.y1)+EndOfLine+str(g.x)+EndOfLine+str(g.y)
```

Notes: Due we use this class for three different ways, we have three constructors to fill in the value you need for the calls.

4.23.2 Methods

4.23.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor for creating an empty object.
See also:

- 4.23.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 322
- 4.23.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double) 323
- 4.23.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 323

4.23.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create the arguments object for the PathArc methods in GM16GraphicsMBS.
See also:

- 4.23.3 Constructor 322

- 4.23.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double) 323
- 4.23.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 323

4.23.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create the arguments object for the QuadraticCurveto methods in GM16GraphicsMBS.
Example:

```
dim g as new GM16PathArgsMBS(1,2,3,4)

MsgBox str(g.x1)+EndOfLine+str(g.y1)+EndOfLine+str(g.x)+EndOfLine+str(g.y)
```

See also:

- 4.23.3 Constructor 322
- 4.23.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 322
- 4.23.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 323

4.23.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create the arguments object for the Curveto methods in GM16GraphicsMBS.
Example:

```
dim g as new GM16PathArgsMBS(1,2,3,4,5,6)

MsgBox str(g.x1)+EndOfLine+str(g.y1)+EndOfLine+str(g.x2)+EndOfLine+str(g.y2)+EndOfLine+str(g.x)+EndOfLine+str(g.y)
```

See also:

- 4.23.3 Constructor 322
- 4.23.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 322

- 4.23.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double) 323

4.23.7 Properties

4.23.8 largeArcFlag as Boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The large arc flag.
Notes: Draw longer of the two matching arcs
(Read and Write property)

4.23.9 radiusX as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The radius x value.
Notes: (Read and Write property)

4.23.10 radiusY as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The radius y value.
Notes: (Read and Write property)

4.23.11 sweepFlag as Boolean

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The sweep flag value.
Notes: Draw arc matching clock-wise rotation.
(Read and Write property)

4.23.12 x as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x value.
Notes: For an arc: End-point X

(Read and Write property)

4.23.13 x1 as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x1 value.
Notes: (Read and Write property)

4.23.14 x2 as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x2 value.
Notes: (Read and Write property)

4.23.15 xAxisRotation as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x Axis Rotation value.
Notes: Rotation relative to X axis.
(Read and Write property)

4.23.16 y as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y value.
Notes: for an arc: End-point Y
(Read and Write property)

4.23.17 y1 as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y1 value.
Notes: (Read and Write property)

4.23.18 y2 as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y2 value.
Notes: (Read and Write property)

4.24 class GM16PixelsMBS

4.24.1 class GM16PixelsMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty pixels object.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.jpg”)
dim g as new GM16ImageMBS(f)
dim p as new GM16PixelsMBS(g)

// get pointer to some pixels to read/write
dim x as ptr = p.get(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
p.sync

// show
window1.Backdrop = g.CopyPicture
```

4.24.2 Methods

4.24.3 Constructor(Image as GM16ImageMBS)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new Pixels object with the pixels from an image.

4.24.4 get(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfer pixels from the image to the pixel view as defined by the specified region.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim p as new GM16PixelsMBS(g)

// get pointer to some pixels
dim x as ptr = p.get(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
p.sync

// show
window1.Backdrop = g.CopyPicture
```

Notes: Modified pixels may be subsequently transferred back to the image via sync.

4.24.5 getConst(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfer read-only pixels from the image to the pixel view as defined by the specified region.

4.24.6 indexes as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Return pixel colormap index array.

4.24.7 set(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Allocate a pixel view region to store image pixels as defined by the region rectangle.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim p as new GM16PixelsMBS(g)

// get pointer to some pixels to write
dim x as ptr = p.set(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
p.sync

// show
window1.Backdrop = g.CopyPicture
```

Notes: This area is subsequently transferred from the pixel view to the image via sync.

4.24.8 sync

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers the image cache pixels to the image.

4.24.9 Properties

4.24.10 columns as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width of view.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GM16ImageMBS(f)
dim p as new GM16PixelsMBS(g)
```

```
// get pointer to some pixels
dim x as ptr = p.get(0, 0, 100, 100)

// and show size
MsgBox str(p.columns)+" x "+str(p.rows)
```

Notes: (Read only property)

4.24.11 handle as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.24.12 rows as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Height of view.
Notes: (Read only property)

4.24.13 x as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Left ordinate of view.
Notes: (Read only property)

4.24.14 y as Integer

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Top ordinate of view.
Notes: (Read only property)

4.25 class GM16TypeMetricMBS

4.25.1 class GM16TypeMetricMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The TypeMetric class provides the means to pass data from the Image class's TypeMetric method to the user.
Notes: It provides information regarding font metrics such as ascent, descent, text width, text height, and maximum horizontal advance. The units of these font metrics are in pixels, and that the metrics are dependent on the current Image font (default Ghostscript's "Helvetica"), pointsize (default 12 points), and x/y resolution (default 72 DPI) settings.

The pixel units may be converted to points (the standard resolution-independent measure used by the typesetting industry) via the following equation:

size_points = (size_pixels * 72)/resolution
where resolution is in dots-per-inch (DPI). This means that at the default image resolution, there is one pixel per point.

Note that a font's pointsize is only a first-order approximation of the font height (ascender + descender) in points. The relationship between the specified pointsize and the rendered font height is determined by the font designer.

See FreeType Glyph Conventions for a detailed description of font metrics related issues.

4.25.2 Methods

4.25.3 Constructor

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.25.4 Properties

4.25.5 ascent as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the distance in pixels from the text baseline to the highest/upper grid coordinate used

to place an outline point.
Notes: Always a positive value.
(Read only property)

4.25.6 descent as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the the distance in pixels from the baseline to the lowest grid coordinate used to place an outline point.
Notes: Always a negative value.
(Read only property)

4.25.7 maxHorizontalAdvance as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the maximum horizontal advance (advance from the beginning of a character to the beginning of the next character) in pixels.
Notes: (Read only property)

4.25.8 textHeight as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns text height in pixels.
Notes: (Read only property)

4.25.9 textWidth as Double

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns text width in pixels.
Notes: (Read only property)

4.26 class GM16UnsupportedExceptionMBS

4.26.1 class GM16UnsupportedExceptionMBS

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: An exception raised if you call the GM functions on an unsupported platform.
Notes: Check the message property for details.
This exception is currently only used on Windows.
(Windows support may come later)
Subclass of the GM16ErrorExceptionMBS class.

4.27 class GMBlobMBS

4.27.1 class GMBlobMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for binary large objects.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GMImageMBS
dim blob as new GMBlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

Notes: Blob provides the means to contain any opaque data. It is named after the term "Binary Large OBject" commonly used to describe unstructured data (such as encoded images) which is stored in a database. While the function of Blob is very simple (store a pointer and and size associated with allocated data), the Blob class provides some very useful capabilities. In particular, it is fully reference counted just like the Image class.

The Blob class supports value assignment while preserving any outstanding earlier versions of the object. Since assignment is via a pointer internally, Blob is efficient enough to be stored directly in an STL container or any other data structure which requires assignment. In particular, by storing a Blob in an associative container (such as STL's 'map') it is possible to create simple indexed in-memory "database" of Blobs.

Magick++ currently uses Blob to contain encoded images (e.g. JPEG) as well as ICC and IPTC profiles. Since Blob is a general-purpose class, it may be used for other purposes as well.
Blog Entries

- Tip of the day: Render SVG with GraphicsMagick Plugin

4.27.2 Methods

4.27.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Default constructor creating an empty blob object.
See also:

- 4.27.4 Constructor(data as memoryblock, offset as Integer, size as Integer) 335
- 4.27.5 Constructor(data as string) 335
- 4.27.6 Constructor(other as GMBlobMBS) 336

4.27.4 Constructor(data as memoryblock, offset as Integer, size as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct object with data, making a copy of the supplied data.
See also:

- 4.27.3 Constructor 335
- 4.27.5 Constructor(data as string) 335
- 4.27.6 Constructor(other as GMBlobMBS) 336

4.27.5 Constructor(data as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct object with data, making a copy of the supplied data.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GMImageMBS
dim blob as new GMBlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
```

Backdrop=mp.CombinePictureWithMask

See also:

- 4.27.3 Constructor 335
- 4.27.4 Constructor(data as memoryblock, offset as Integer, size as Integer) 335
- 4.27.6 Constructor(other as GMBlobMBS) 336

4.27.6 Constructor(other as GMBlobMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copy constructor (reference counted).
See also:

- 4.27.3 Constructor 335
- 4.27.4 Constructor(data as memoryblock, offset as Integer, size as Integer) 335
- 4.27.5 Constructor(data as string) 335

4.27.7 CopyMemory as memoryblock

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a copy of the data as a memoryblock.
Notes: Returns nil on any error like low memory.

4.27.8 CopyString as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a copy of the data as a string.

4.27.9 Data as Ptr

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: A memoryblock with the data from this blob.
Example:

```
dim b as new GMBlobMBS("Hello")

dim m as memoryblock = b.Data
MsgBox m.StringValue(0,5) // shows "Hello"
```

Notes: This is a memoryblock referencing the data of the blob. It has no size set. The memoryblock can only be used as long as the blob object exists. if you use it after you destroyed the blob object, you can crash you application.

4.27.10 Update(data as memoryblock, offset as Integer, size as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces the content of this blob with a copy of the bytes in the memoryblock.
See also:

- 4.27.11 Update(data as string) 337

4.27.11 Update(data as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces the content of this blob with a copy of the bytes in the string.
Notes: Offset is zero based.
See also:

- 4.27.10 Update(data as memoryblock, offset as Integer, size as Integer) 337

4.27.12 Properties

4.27.13 handle as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal handle of the blob object.
Notes: (Read and Write property)

4.27.14 length as UInt64

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain data length in bytes.
Example:

```
dim b as new GMBlobMBS("Hello")

MsgBox str(B.length) // shows 5
```

Notes: (Read only property)

4.27.15 base64 as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blob content as a string in Base64 format.
Example:

```
dim b as new GMBlobMBS("Hello")

MsgBox b.base64 // shows "SGVsbG8="
```

Notes: (Read and Write computed property)

4.28 class GMCoderInfoMBS

4.28.1 class GMCoderInfoMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class used to get information about all registered coders.
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GMCoderInfoMBS in coders
names.Append coder.name
next

MsgBox Join(names,EndOfLine)
```

Notes: The CoderInfo class provides the means to provide information regarding GraphicsMagick support for an image format (designated by a magick string). It may be used to provide support for a specific named format (provided as an argument to the constructor), or as an element of a container when format support is queried using the coderInfoList() templated function.
Blog Entries

- MBS Xojo Plugins, version 18.4pr1

4.28.2 Methods

4.28.3 CoderInfoList as GMCoderInfoMBS()

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a list of all coders.
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
```

4.28.4 Properties

4.28.5 description as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format description (e.g. ”CompuServe graphics interchange format”).
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GMCoderInfoMBS in coders
names.Append coder.name+” ”+coder.description
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.28.6 isMultiFrame as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format supports multiple frames.
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GMCoderInfoMBS in coders
names.Append coder.name+” ”+str(coder.isMultiFrame)
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.28.7 isReadable as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format is readable.
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GMCoderInfoMBS in coders
```

```
names.Append coder.name+" "+str(coder.isReadable)
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.28.8 isWritable as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format is writeable.
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
dim names(-1) as string

for each coder as GMCoderInfoMBS in coders
names.Append coder.name+" "+str(coder.isWritable)
next

MsgBox Join(names,EndOfLine)
```

Notes: (Read and Write property)

4.28.9 ModuleName as String

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Name of loadable module.
Notes: (Read and Write property)

4.28.10 name as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format name (e.g. "GIF").
Example:

```
dim coders(-1) as GMCoderInfoMBS = GMCoderInfoMBS.CoderInfoList
dim coder as GMCoderInfoMBS = coders(0) // pick first one
```

MsgBox coder.name

Notes: (Read and Write property)

4.28.11 Note as String

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Usage note for user.
Notes: (Read and Write property)

4.28.12 Version as String

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Support library version.
Notes: (Read and Write property)

4.29 class GMColorGrayMBS

4.29.1 class GMColorGrayMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color subclass for a grayscale color.
Example:

```
dim g as new GMColorGrayMBS(0.5)
MsgBox str(g.shade)
```

Notes: Representation of grayscale RGB color.
Equal parts red, green, and blue specified as a ratio (0 to 1).
Subclass of the GMColorMBS class.

4.29.2 Methods

4.29.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GMColorGrayMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.29.4 Constructor(other as GMColorMBS) 343
- 4.29.5 Constructor(shade as Double) 344

4.29.4 Constructor(other as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GMColorGrayMBS(0.5)
dim o as new GMColorGrayMBS(g)
```

```
MsgBox str(o.shade)
```

See also:

- 4.29.3 Constructor 343
- 4.29.5 Constructor(shade as Double) 344

4.29.5 Constructor(shade as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given value.
Example:

```
dim g as new GMColorGrayMBS(1.0)
MsgBox str(g.colorValue)
```

Notes: Range is 0.0 to 1.0.
See also:

- 4.29.3 Constructor 343
- 4.29.4 Constructor(other as GMColorMBS) 343

4.29.6 Properties

4.29.7 shade as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The gray value for this color.
Example:

```
dim g as new GMColorGrayMBS(1.0)
MsgBox str(g.shade)
```

Notes: Range is 0.0 to 1.0
(Read and Write property)

4.30 class GMColorHSLMBS

4.30.1 class GMColorHSLMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for a HSL color.
Example:

```
dim g as new GMColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.colorValue)
```

Notes: Subclass of the GMColorMBS class.

4.30.2 Methods

4.30.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GMColorHSLMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.30.4 Constructor(hue as Double, saturation as Double, luminosity as Double) 345
- 4.30.5 Constructor(other as GMColorMBS) 346

4.30.4 Constructor(hue as Double, saturation as Double, luminosity as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim g as new GMColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.hue)+" "+str(g.saturation)+" "+str(g.luminosity)
```

See also:

- 4.30.3 Constructor 345
- 4.30.5 Constructor(other as GMColorMBS) 346

4.30.5 Constructor(other as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GMColorHSLMBS(0.1,0.2,0.3)
dim o as new GMColorHSLMBS(g)

MsgBox str(o.colorValue)
```

See also:

- 4.30.3 Constructor 345
- 4.30.4 Constructor(hue as Double, saturation as Double, luminosity as Double) 345

4.30.6 Properties

4.30.7 hue as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The hue value.
Example:

```
dim g as new GMColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.hue)
```

Notes: (Read and Write property)

4.30.8 luminosity as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The luminosity value.
Example:

```
dim g as new GMColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.luminosity)
```

Notes: (Read and Write property)

4.30.9 saturation as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The saturation value.
Example:

```
dim g as new GMColorHSLMBS(0.1,0.2,0.3)
MsgBox str(g.saturation)
```

Notes: (Read and Write property)

4.31 class GMColorMBS

4.31.1 class GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color is the base color class.
Example:

```
dim c as new GMColorMBS(127,255,127) // light green
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

Notes: It is a simple container class for the pixel red, green, blue, and alpha values scaled to fit GraphicsMagick's Quantum size. Normally users will instantiate a class derived from Color which supports the color model that fits the needs of the application. The Color class may be constructed directly from an X11-style color string. As a perhaps odd design decision, the value transparent black is considered to represent an unset value (invalid color) in many cases. This choice was made since it avoided using more memory. The default Color constructor constructs an invalid color (i.e. transparent black) and may be used as a parameter in order to remove a color setting.
Blog Entries

- MBS Xojo Plugins 18.3
- MBS Xojo Plugins, version 18.3pr1
- Gradients in GraphicsMagick

4.31.2 Methods

4.31.3 Black as GMColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries a black color.
Example:

```
dim black as GMColorMBS = GMColorMBS.Black

MsgBox str(black.colorValue)
```

4.31.4 Color(ColorValue as Color) as GMColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts color from Xojo to GMColorMBS.
Example:

```
dim c as GMColorMBS = GMColorMBS.Color(&cFF0000)

MsgBox str(c.colorValue)
```

See also:

- 4.31.5 Color(ColorValue as Color, alpha as Integer) as GMColorMBS 349
- 4.31.6 Color(red as integer, green as integer, blue as integer) as GMColorMBS 349
- 4.31.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GMColorMBS 350

4.31.5 Color(ColorValue as Color, alpha as Integer) as GMColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts color from Xojo to GMColorMBS with separate alpha value.
Example:

```
dim c as GMColorMBS = GMColorMBS.Color(&cFF0000, 128)

MsgBox str(c.colorValue)+" "+str(c.alpha)
```

Notes: Alpha in range from 0 to 255.
See also:

- 4.31.4 Color(ColorValue as Color) as GMColorMBS 348
- 4.31.6 Color(red as integer, green as integer, blue as integer) as GMColorMBS 349
- 4.31.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GMColorMBS 350

4.31.6 Color(red as integer, green as integer, blue as integer) as GMColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates color with RGB values.
Example:

```
dim c as GMColorMBS = GMColorMBS.Color(127, 191, 255)

MsgBox str(c.colorValue)
```

Notes: Range in 0 to 255 for 8bit and 0 to 65535 for 16bit class.
See also:

- 4.31.4 Color(ColorValue as Color) as GMColorMBS 348
- 4.31.5 Color(ColorValue as Color, alpha as Integer) as GMColorMBS 349
- 4.31.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GMColorMBS 350

4.31.7 Color(red as integer, green as integer, blue as integer, alpha as Integer) as GMColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates color with RGB values.
Example:

```
dim c as GMColorMBS = GMColorMBS.Color(127, 191, 255, 127)

MsgBox str(c.colorValue)+" "+str(c.alphaQuantum)
```

Notes: Range in 0 to 255 for 8bit and 0 to 65535 for 16bit class.
See also:

- 4.31.4 Color(ColorValue as Color) as GMColorMBS 348
- 4.31.5 Color(ColorValue as Color, alpha as Integer) as GMColorMBS 349
- 4.31.6 Color(red as integer, green as integer, blue as integer) as GMColorMBS 349

4.31.8 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GMColorMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.31.9 Constructor(ColorName as string) 351

- 4.31.10 Constructor(ColorValue as color) 351
- 4.31.11 Constructor(ColorValue as color, alpha as Integer) 352
- 4.31.12 Constructor(other as GMColorMBS) 352
- 4.31.13 Constructor(red as Integer, green as Integer, blue as Integer) 353
- 4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 354

4.31.9 Constructor(ColorName as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color based on the X11 color name.
Example:

```
dim c as new GMColorMBS("red")

MsgBox str(c.redQuantum)+"-"+str(c.greenQuantum)+"-"+str(c.blueQuantum) // shows "255-0-0"

dim d as new GMColorMBS("#77FF00")

MsgBox str(d.redQuantum)+"-"+str(d.greenQuantum)+"-"+str(d.blueQuantum) // shows "119-255-0"
```

Notes: An alternate way to construct the class is via an X11-compatible color specification string (e.g. Color("red") or Color ("#FF0000")). Since the class may be constructed from a string, convenient strings may be passed in place of an explicit Color object in methods which accept a reference to Color. Color may also be converted to a std::string for convenience in user interfaces, and for saving settings to a text file.
See also:

- 4.31.8 Constructor 350
- 4.31.10 Constructor(ColorValue as color) 351
- 4.31.11 Constructor(ColorValue as color, alpha as Integer) 352
- 4.31.12 Constructor(other as GMColorMBS) 352
- 4.31.13 Constructor(red as Integer, green as Integer, blue as Integer) 353
- 4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 354

4.31.10 Constructor(ColorValue as color)

Plugin Version: 10.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GMColorMBS(&cFF0000)
MsgBox str(c.redQuantum)+” ”+str(c.greenQuantum)+” ”+str(c.blueQuantum)
```

See also:

- 4.31.8 Constructor 350
- 4.31.9 Constructor(ColorName as string) 351
- 4.31.11 Constructor(ColorValue as color, alpha as Integer) 352
- 4.31.12 Constructor(other as GMColorMBS) 352
- 4.31.13 Constructor(red as Integer, green as Integer, blue as Integer) 353
- 4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 354

4.31.11 Constructor(ColorValue as color, alpha as Integer)

Plugin Version: 10.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GMColorMBS(&cFF0102, 127)
MsgBox str(c.redQuantum)+” ”+str(c.greenQuantum)+” ”+str(c.blueQuantum)+” ”+str(c.alpha)
```

See also:

- 4.31.8 Constructor 350
- 4.31.9 Constructor(ColorName as string) 351
- 4.31.10 Constructor(ColorValue as color) 351
- 4.31.12 Constructor(other as GMColorMBS) 352
- 4.31.13 Constructor(red as Integer, green as Integer, blue as Integer) 353
- 4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 354

4.31.12 Constructor(other as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim r as new GMColorMBS(1,2,3)
dim c as new GMColorMBS(r)
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.31.8 Constructor 350
- 4.31.9 Constructor(ColorName as string) 351
- 4.31.10 Constructor(ColorValue as color) 351
- 4.31.11 Constructor(ColorValue as color, alpha as Integer) 352
- 4.31.13 Constructor(red as Integer, green as Integer, blue as Integer) 353
- 4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 354

4.31.13 Constructor(red as Integer, green as Integer, blue as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GMColorMBS(1,2,3)
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
See also:

- 4.31.8 Constructor 350
- 4.31.9 Constructor(ColorName as string) 351
- 4.31.10 Constructor(ColorValue as color) 351
- 4.31.11 Constructor(ColorValue as color, alpha as Integer) 352
- 4.31.12 Constructor(other as GMColorMBS) 352
- 4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer) 354

4.31.14 Constructor(red as Integer, green as Integer, blue as Integer, alpha as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GMColorMBS(1,2,3,4)

// display color, alpha is double...
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)+" "+str(c.alpha)
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
See also:

- 4.31.8 Constructor 350
- 4.31.9 Constructor(ColorName as string) 351
- 4.31.10 Constructor(ColorValue as color) 351
- 4.31.11 Constructor(ColorValue as color, alpha as Integer) 352
- 4.31.12 Constructor(other as GMColorMBS) 352
- 4.31.13 Constructor(red as Integer, green as Integer, blue as Integer) 353

4.31.15 QuantumByteSize as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum byte size.
Example:

```
MsgBox str(GMColorMBS.QuantumByteSize)
```

Notes: As the plugin uses 8 bit this value should be 1.

4.31.16 scaleDoubleToQuantum(value as Double) as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales a double value to a value in the range of a quantum.
Example:

```
dim d as Double = 1.0
dim v as Integer = GMColorMBS.scaleDoubleToQuantum(d)
MsgBox str(v)
```

Notes: As the plugin uses 8 bit quantums, this is basicly a multiplication by 255.0

4.31.17 scaleQuantumToDouble(value as Integer) as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales a quantum to a double value.
Example:

```
dim v as Integer = 255
dim d as Double = GMColorMBS.scaleQuantumToDouble(v)
MsgBox str(d)
```

Notes: The plugin uses 8bit quantums, so this is basicly the division of value by 255.0

4.31.18 White as GMColorMBS

Plugin Version: 18.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries a white color.
Example:

```
dim White as GMColorMBS = GMColorMBS.White

MsgBox str(White.colorValue)
```

4.31.19 Properties

4.31.20 alpha as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha value of this color.
Example:

```
dim c as new GMColorMBS(1,2,3,1.0)
MsgBox str(c.alpha)
```

Notes: Range is 0.0 to 1.0. If you pass values higher, they are divided by 255.
(Read and Write property)

4.31.21 alphaQuantum as Integer

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha color value.
Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.31.22 blueQuantum as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color value.
Example:

```
dim c as new GMColorMBS(1,2,3)
MsgBox str(c.redQuantum) // 3
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.31.23 colorValue as color

Plugin Version: 10.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Xojo color for the GraphicsMagick color.
Example:

```
dim c as new GMColorMBS(&cFF0102)
MsgBox str(c.ColorValue)
```

Notes: (Read and Write property)

4.31.24 greenQuantum as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color value.
Example:

```
dim r as new GMColorMBS(1,2,3)
MsgBox str(r.greenQuantum) // shows 2
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.31.25 handle as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal color reference.
Example:

```
dim r as new GMColorMBS(1,2,3)
MsgBox str(r.handle)
```

Notes: (Read and Write property)

4.31.26 intensity as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The intensity of this color.
Example:

```
dim c as new GMColorMBS(1,2,3)
MsgBox str(c.intensity)
```

Notes: (Read only property)

4.31.27 isValid as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Does object contain valid color?
Example:

```
dim c as new GMColorMBS(1,2,3)
MsgBox str(c.isValid)
```

Notes: (Read and Write property)

4.31.28 redQuantum as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color value.
Example:

```
dim c as new GMColorMBS(1,2,3)
MsgBox str(c.redQuantum) // 1
```

Notes: For 8-bit range is 0 to 255.
For 16-bit range is 0 to 65535.
(Read and Write property)

4.32 class GMColorMonoMBS

4.32.1 class GMColorMonoMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Representation of a black/white color (true/false)
Example:

```
dim g as new GMColorMonoMBS(false)
MsgBox str(g.colorValue)
```

Notes: Subclass of the GMColorMBS class.

4.32.2 Methods

4.32.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GMColorMonoMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.32.4 Constructor(mono as boolean) 359
- 4.32.5 Constructor(other as GMColorMBS) 360

4.32.4 Constructor(mono as boolean)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim g as new GMColorMonoMBS(false)
MsgBox str(g.mono)
```

See also:

- 4.32.3 Constructor 359
- 4.32.5 Constructor(other as GMColorMBS) 360

4.32.5 Constructor(other as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GMColorMonoMBS(false)
dim o as new GMColorMonoMBS(g)
MsgBox str(o.mono)
```

See also:

- 4.32.3 Constructor 359
- 4.32.4 Constructor(mono as boolean) 359

4.32.6 Properties

4.32.7 mono as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color value.
Example:

```
dim g as new GMColorMonoMBS(true)
MsgBox str(g.mono)
```

Notes: (Read and Write property)

4.33 class GMColorRGBMBS

4.33.1 class GMColorRGBMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color class for RGB colors.
Example:

```
dim c as new GMColorRGBMBS(1.0,0.0,0.0) // red
MsgBox str(C.red)+" "+str(c.green)+" "+str(c.blue)
MsgBox str(C.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

Notes: Representation of RGB color with red, green, and blue specified as ratios (0 to 1)
Subclass of the GMColorMBS class.
Blog Entries

- Crop a two side page document to a single page document

4.33.2 Methods

4.33.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GMColorRGBMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.33.4 Constructor(other as GMColorMBS) 361
- 4.33.5 Constructor(red as Double, green as Double, blue as Double) 362

4.33.4 Constructor(other as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GMColorRGBMBS(1,2,3)
dim o as new GMColorRGBMBS(g)
MsgBox str(o.colorValue)
```

See also:

- 4.33.3 Constructor 361
- 4.33.5 Constructor(red as Double, green as Double, blue as Double) 362

4.33.5 Constructor(red as Double, green as Double, blue as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim c as new GMColorRGBMBS(0.1,0.2,0.3)
```

Notes: Range is 0.0 to 1.0.
See also:

- 4.33.3 Constructor 361
- 4.33.4 Constructor(other as GMColorMBS) 361

4.33.6 Properties

4.33.7 blue as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color component.
Example:

```
dim c as new GMColorRGBMBS(0.0,0.0,1.0)
MsgBox str(c.blue)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.33.8 green as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color component.
Example:

```
dim c as new GMColorRGBMBS(0.0,1.0,0.0)
MsgBox str(c.green)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.33.9 red as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color component.
Example:

```
dim c as new GMColorRGBMBS(1.0,0.0,0.0) // red
MsgBox str(C.red)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.34 class GMColorYUVMBS

4.34.1 class GMColorYUVMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Representation of a color in the YUV colorspace
Example:

```
dim g as new GMColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.y)+" "+str(g.u)+" "+str(g.v)
```

Notes: Subclass of the GMColorMBS class.

4.34.2 Methods

4.34.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with transparent black.
Example:

```
dim c as new GMColorYUVMBS
MsgBox str(c.redQuantum)+" "+str(c.greenQuantum)+" "+str(c.blueQuantum)
```

See also:

- 4.34.4 Constructor(other as GMColorMBS) 364
- 4.34.5 Constructor(y as Double, u as Double, v as Double) 365

4.34.4 Constructor(other as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color copying the existing color.
Example:

```
dim g as new GMColorYUVMBS(0.1, 0.2, 0.3)
dim o as new GMColorYUVMBS(g)
MsgBox str(o.colorValue)
```

See also:

- 4.34.3 Constructor 364
- 4.34.5 Constructor(y as Double, u as Double, v as Double) 365

4.34.5 Constructor(y as Double, u as Double, v as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color with the given values.
Example:

```
dim g as new GMColorYUVMBS(0.1, 0.2, 0.3)
```

See also:

- 4.34.3 Constructor 364
- 4.34.4 Constructor(other as GMColorMBS) 364

4.34.6 Properties

4.34.7 u as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The u color component.
Example:

```
dim g as new GMColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.u)
```

Notes: Range is -0.5 to +0.5.
(Read and Write property)

4.34.8 v as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The v color component.
Example:

```
dim g as new GMColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.v)
```

Notes: Range is -0.5 to +0.5.
(Read and Write property)

4.34.9 y as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y color component.
Example:

```
dim g as new GMColorYUVMBS(0.1, 0.2, 0.3)
MsgBox str(g.y)
```

Notes: Range is 0.0 to 1.0.
(Read and Write property)

4.35 class GMConvertMBS

4.35.1 class GMConvertMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class to convert images thread friendly.
Notes: This class is intended to process lots of images in several instances in several threads.
If you for example use 8 Xojo threads, to process thousands of images to scale them down for thumbnails, you can easily keep 8 CPU cores busy.

Please make a new instance, set options and call run method. When run is done, please read output properties.
Do not modify properties while thread is running.
Blog Entries

- MBS Xojo Plugins, version 23.5pr7
- Multithreaded plugin functions can increase speed of Xojo application
- MonkeyBread Software Releases the MBS Xojo Plugins in version 19.1
- MBS Xojo Plugins, version 19.1pr2
- CGImageSourceMBS CreateThumbnailMT method for Xojo

Videos

- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 17.5, page 39: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes
- 17.3, page 11: News

4.35.2 Methods

4.35.3 Constructor

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.35.4 Run

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Starts converter.
Notes: The work is performed on a preemptive thread, so this function does not block the application and can yield time to other Xojo threads. Must be called in a Xojo thread to enjoy benefits. If called in main thread will block, but keep other background threads running.
If you run several threads calling MT methods, you can get all CPU cores busy while main thread shows GUI with progress window.

4.35.5 Properties

4.35.6 AutoOrient as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to call autoOrient method to change orientation of image data to 0¬∞.
Notes: (Read and Write property)

4.35.7 Enhance as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to run enhance command.
Notes: If true, we call enhance on the image to minimize noise.
(Read and Write property)

4.35.8 Equalize as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to run equalize command.
Notes: If set to true, we call equalize command on image (histogram equalization).
(Read and Write property)

4.35.9 ImageType as Integer

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the image type.

Notes: If value is >0, the image type is change to the given type.
(Read and Write property)

4.35.10 InputData as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input data string.
Notes: If set, we read image from this data.
(Read and Write property)

4.35.11 InputFile as FolderItem

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input folderitem.
Notes: If set input file to read.
(Read and Write property)

4.35.12 InputGeometry as GMGeometryMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input geometry.
Notes: Some formats can be loaded with different scales, so this geometry is passed to read method to define the format requested.
(Read and Write property)

4.35.13 InputImage as GMImageMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input image.
Notes: (Read and Write property)

4.35.14 InputMagick as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image magick type.
Notes: If empty, the type of file is automatically determined.
(Read and Write property)

4.35.15 InputMemory as MemoryBlock

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input data memoryblock.
Notes: If set, we read image from this data.
(Read and Write property)

4.35.16 InputPath as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The input file path.
Notes: If set input file to read.
(Read and Write property)

4.35.17 OutputData as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output data as string.
Notes: (Read and Write property)

4.35.18 OutputFile as FolderItem

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The folderitem where to write file.
Notes: If set, the image will be written to this file.
(Read and Write property)

4.35.19 OutputImage as GMImageMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output image object.
Notes: (Read and Write property)

4.35.20 OutputMagick as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output magick.
Notes: You can set this to a magick codec type to define output format, e.g. ”jpeg”.
(Read and Write property)

4.35.21 OutputMemory as MemoryBlock

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Output data as memory block.
Notes: (Read and Write property)

4.35.22 OutputPath as String

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The native file path for output.
Notes: If set, the image will be written to this path.
(Read and Write property)

4.35.23 Quality as Integer

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Quality of images in range 1 to 100.
Notes: If value is >0, we assign it to the image for setting image quality.
(Read and Write property)

4.35.24 Running as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether this converter is running.
Notes: Set to true while Run method runs.

(Read only property)

4.35.25 ScaleGeometry as GMGeometryMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The geometry for call to scale method.
Notes: scale method is called with this geometry (if set) to reduce image size.
(Read and Write property)

4.35.26 Strip as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to remove all profiles and text attributes from the image.
Notes: (Read and Write property)

4.35.27 ThumbnailGeometry as GMGeometryMBS

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The geometry for call to thumbnail method.
Notes: thumbnail method is called with this geometry (if set) to reduce image size.
(Read and Write property)

4.35.28 Trim as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether trim command is called on image to remove extra blank space around image.
Notes: (Read and Write property)

4.35.29 WantOutputData as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether you want image compressed and stored in OutputData property.
Notes: (Read and Write property)

4.35.30 WantOutputMemory as Boolean

Plugin Version: 19.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether you want image compressed and stored in OutputMemory property.
Notes: (Read and Write property)

4.36 class GMCoordinateMBS

4.36.1 class GMCoordinateMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Graphics Magick class for a coordinate.
Example:

```
dim c as new GMCoordinateMBS(5,6)
MsgBox str(c.x)+" "+str(c.y)
```

4.36.2 Methods

4.36.3 Constructor

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create a new coordinate.
See also:

- 4.36.4 Constructor(x as Double, y as Double) 374

4.36.4 Constructor(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create a new coordinate.
Example:

```
dim c as new GMCoordinateMBS(5,6)
MsgBox str(c.x)+" "+str(c.y)
```

See also:

- 4.36.3 Constructor 374

4.36.5 Properties

4.36.6 x as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x value.
Example:

```
dim c as new GMCoordinateMBS
c.x = 5
MsgBox str(c.x)
```

Notes: (Read and Write property)

4.36.7 y as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y value.
Example:

```
dim c as new GMCoordinateMBS
c.y = 5
MsgBox str(c.y)
```

Notes: (Read and Write property)

4.37 class GMErrorExceptionMBS

4.37.1 class GMErrorExceptionMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception to report errors in the GraphicMagick plugin.
Notes: Check the message property for details.
Subclass of the RuntimeException class.

4.38 class GMGeometryMBS

4.38.1 class GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Geometry provides a convenient means to specify a geometry argument.
Example:

```
dim g as new GMGeometryMBS(300,400)
MsgBox str(G.width)+" "+str(G.height)
```

Notes: The object may be initialized from a string containing a geometry specification. It may also be initialized by more efficient parameterized constructors.
Blog Entries

- Auto crop a signature picture
- Crop a two side page document to a single page document
- MBS Xojo Plugins, version 19.0pr6
- Gradients in GraphicsMagick
- Tip of the day: Render SVG with GraphicsMagick Plugin

Xojo Developer Magazine

- 19.6, page 74: Fun with GraphicsMagick, Cool methods from the GMImageMBS class by Stefanie Juchmes

4.38.2 Methods

4.38.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates empty geometry.
Example:

```
dim g as new GMGeometryMBS
MsgBox str(G.width)+" "+str(G.height)
```

See also:

- 4.38.4 Constructor(geometry as string) 378
- 4.38.5 Constructor(other as GMGeometryMBS) 378
- 4.38.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) 379

4.38.4 Constructor(geometry as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct geometry from string.
Example:

```
dim g as new GMGeometryMBS("600x600")

MsgBox str(G.width)+" "+str(G.height)
```

Notes: See the GraphicsMagick website for details.
http://www.graphicsmagick.org/Magick++/Geometry.html
See also:

- 4.38.3 Constructor 377
- 4.38.5 Constructor(other as GMGeometryMBS) 378
- 4.38.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) 379

4.38.5 Constructor(other as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new geometry object by copying an existing one.
Example:

```
dim g as new GMGeometryMBS(600,600)
dim h as new GMGeometryMBS(g)
MsgBox str(h.width)
```

See also:

- 4.38.3 Constructor 377
- 4.38.4 Constructor(geometry as string) 378

- 4.38.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) 379

4.38.6 Constructor(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates geometry with the given values.
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.width)
```

See also:

- 4.38.3 Constructor 377
- 4.38.4 Constructor(geometry as string) 378
- 4.38.5 Constructor(other as GMGeometryMBS) 378

4.38.7 Make(geometry as string) as GMGeometryMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct geometry from string.
Example:

```
dim g as GMGeometryMBS = GMGeometryMBS.Make(”600x600”)

MsgBox str(G.width)+” ”+str(G.height)
```

Notes: See the GraphicsMagick website for more details:
http://www.graphicsmagick.org/Magick++/Geometry.html
See also:

- 4.38.8 Make(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) as GMGeometryMBS 380

4.38.8 Make(Width as UInt32, Height as UInt32, XOffset as UInt32=0, YOffset as UInt32=0, xNegative as boolean=false, yNegative as boolean=false) as GMGeometryMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates geometry with the given values.
Example:

```
dim g as GMGeometryMBS = GMGeometryMBS.Make(600,600)
MsgBox str(g.width)
```

See also:

- 4.38.7 Make(geometry as string) as GMGeometryMBS 379

4.38.9 Properties

4.38.10 aspect as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize without preserving aspect ratio (!).
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.aspect)
```

Notes: (Read and Write property)

4.38.11 fillArea as Boolean

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image to fit total pixel area specified by dimensions.
Notes: Same as @ in the geometry specification.
(Read and Write property)

4.38.12 greater as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize if image is greater than size (>).
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.greater)
```

Notes: (Read and Write property)

4.38.13 height as Uint32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height value.
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.height)
```

Notes: (Read and Write property)

4.38.14 isValid as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Does object contain a valid geometry?
Example:

```
dim g as new GMGeometryMBS(100,200)
MsgBox str(G.isValid)
```

Notes: May be set to false in order to invalidate an existing geometry object.
(Read and Write property)

4.38.15 less as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize if image is less than size (<).
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.less)
```

Notes: (Read and Write property)

4.38.16 limitPixels as Boolean

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Dimensions are treated as minimum rather than maximum values.
Notes: Same as ^in the geometry specification.
(Read and Write property)

4.38.17 percent as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height are expressed as percentages.
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.percent)
```

Notes: (Read and Write property)

4.38.18 StringValue as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The string representation of the geometry object.
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.StringValue)
```

Notes: (Read and Write property)

4.38.19 width as Uint32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width value.
Example:

```
dim g as new GMGeometryMBS(600,600)
MsgBox str(g.width)
```

Notes: (Read and Write property)

4.38.20 xNegative as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sign of X offset negative? (X origin at right)
Example:

```
dim g as new GMGeometryMBS(100,200,30,40,true,false)
MsgBox str(G.xNegative)
```

Notes: (Read and Write property)

4.38.21 xOff as Uint32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: X offset from origin.
Example:

```
dim g as new GMGeometryMBS(100,200,30,40,true,true)
MsgBox str(G.xOff)+" "+str(G.yOff)
```

Notes: (Read and Write property)

4.38.22 yNegative as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sign of Y offset negative? (Y origin at bottom)
Example:

```
dim g as new GMGeometryMBS(100,200,30,40,false,true)
MsgBox str(G.yNegative)
```

Notes: (Read and Write property)

4.38.23 yOff as Uint32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Y offset from origin
Example:

```
dim g as new GMGeometryMBS(100,200,30,40,true,true)
MsgBox str(G.xOff)+" "+str(G.yOff)
```

Notes: (Read and Write property)

4.39 class GMGraphicsMBS

4.39.1 class GMGraphicsMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for drawing commands targeting a GMImageMBS.
Notes: Please remember that all commands are collected till you call the Draw method.
Blog Entries

- Crop a two side page document to a single page document
- MBS Real Studio Plugins, version 12.5pr13

4.39.2 Methods

4.39.3 Arc(startX as Double, startY as Double, endX as Double, endY as Double, startDegrees as Double, endDegrees as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an arc using the stroke color and based on the circle starting at coordinates startX,startY, and ending with coordinates endX,endY, and bounded by the rotational arc startDegrees,endDegrees.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.type = image.TrueColorType
image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.arc(250, 250, 100, 100,50,300)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.4 Bezier(values() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a bezier curve using the stroke color and based on the coordinates specified by the coordinates array.

4.39.5 Circle(originX as Double, originY as Double, perimX as Double, perimY as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a circle using the stroke color and thickness using specified origin and perimeter coordinates.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.type = image.TrueColorType
image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.39.6 ClipPath(id as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Select a drawing clip path matching id.

4.39.7 ColorPixel(x as Double, y as Double, paintMethod as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color image according to paintMethod.
Notes: The point method recolors the target pixel. The replace method recolors any pixel that matches

the color of the target pixel. Floodfill recolors any pixel that matches the color of the target pixel and is a neighbor, whereas filltoborder recolors any neighbor pixel that is not the border color. Finally, reset recolors all pixels.

4.39.8 CompositeImage(x as Double, y as Double, file as folderitem)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through.
See also:

- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388

- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.10 CompositeImage(x as Double, y as Double, path as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, at specified coordinates.

Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, using specified composition algorithm, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390

- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, using specified composition algorithm, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer) 392

4.39.16 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string, CompositeOperator as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composite current image with contents of specified image, rendered with specified width and height, using specified composition algorithm, at specified coordinates.
Notes: If the matte attribute is set to true, then the image composition will consider an alpha channel, or transparency, present in the image file so that non-opaque portions allow part (or all) of the composite image to show through. If the specified width or height is zero, then the image is composited at its natural size, without enlargement or reduction.
See also:

- 4.39.8 CompositeImage(x as Double, y as Double, file as folderitem) 387
- 4.39.9 CompositeImage(x as Double, y as Double, image as GMImageMBS) 387
- 4.39.10 CompositeImage(x as Double, y as Double, path as string) 388
- 4.39.11 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem) 388
- 4.39.12 CompositeImage(x as Double, y as Double, w as Double, h as Double, file as folderitem, CompositeOperator as Integer) 389
- 4.39.13 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS) 390
- 4.39.14 CompositeImage(x as Double, y as Double, w as Double, h as Double, image as GMImageMBS, CompositeOperator as Integer) 390
- 4.39.15 CompositeImage(x as Double, y as Double, w as Double, h as Double, path as string) 391

4.39.17 Constructor(image as GMImageMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new object referencing the given image.

4.39.18 DashArray(values() as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the pattern of dashes and gaps used to stroke paths.
Notes: The strokeDashArray represents a zero-terminated array of numbers that specify the lengths of alternating dashes and gaps in pixels. If an odd number of values is provided, then the list of values is repeated to yield an even number of values. A typical strokeDashArray array might contain the members 5 3 2 0, where the zero value indicates the end of the pattern array.

4.39.19 DashOffset(offset as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the distance into the dash pattern to start the dash.
Notes: See documentation on SVG's stroke-dashoffset property for usage details.

4.39.20 Draw

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws all draw commands collected.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.type = image.TrueColorType
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.StrokeColor new GMColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.21 DrawPath

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw on image using vector path.
Example:

```
// new picture, 500x500 and filled with white
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)


dim draw as GMGraphicsMBS = image.Graphics

// Draw path
```

```
dim cr as new GMColorRGBMBS("red")
dim gr as new GMColorRGBMBS("green")
draw.StrokeColor cr
draw.FillColor gr
draw.PathMovetoAbs(30,10)
draw.PathLinetoAbs(20,55)
draw.PathLinetoAbs(70,50)
draw.PathLinetoAbs(80,5)
draw.DrawPath

draw.Draw

// show picture
image.type = image.TrueColorType // make sure it's a bitmap
Backdrop=image.CopyPicture
```

4.39.22 Ellipse(originX as Double, originY as Double, perimX as Double, perimY as Double, arcStart as Double, arcEnd as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an ellipse using the stroke color and thickness, specified origin, x & y radius, as well as specified start and end of arc in degrees.
Notes: If a fill color is specified, then the object is filled.

4.39.23 FillColor(c as GMColorMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify drawing object fill color.

4.39.24 FillOpacity(opacity as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify opacity to use when drawing using fill color.

4.39.25 FillRule(fillRule as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the algorithm which is to be used to determine what parts of the canvas are included inside the shape.
Notes: See documentation on SVG's fill-rule property for usage details.

4.39.26 Font(fontname as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify font name to use when drawing text.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS(”white”) // white
dim image as new GMImageMBS(g, c)

dim draw as GMGraphicsMBS = image.Graphics

// draw red text
draw.strokeColor(new GMColorRGBMBS(”red”)) // Outline color
draw.strokeWidth(1)
draw.Font(”/Library/Fonts/Verdana.ttf”)
draw.Text(50, 50, ”Hello”, ””)
draw.Draw

Backdrop=image.CopyPicture
```

See also:

- 4.39.27 Font(fontname as string, StyleType as Integer, weight as Integer, StretchType as Integer) 395

4.39.27 Font(fontname as string, StyleType as Integer, weight as Integer, StretchType as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the font.
Notes: Specify font family, style, weight (one of the set { 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 } with 400 being the normal size), and stretch to be used to select the font used when drawing text. Wildcard matches may be applied to style via the AnyStyle enumeration, applied to weight if weight is zero, and applied to stretch via the AnyStretch enumeration.
See also:

- 4.39.26 Font(fontname as string) 395

4.39.28 Gravity(GravityType as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify text positioning gravity.

4.39.29 Line(startX as Double, startY as Double, endX as Double, endY as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a line using stroke color and thickness using starting and ending coordinates
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a line
draw.Line(100,100,400,400)
draw.Draw

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

4.39.30 Matte(x as Double, y as Double, paintMethod as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Change the pixel matte value to transparent.
Notes: The point method changes the matte value of the target pixel. The replace method changes the matte value of any pixel that matches the color of the target pixel. Floodfill changes the matte value of any pixel that matches the color of the target pixel and is a neighbor, whereas filltoborder changes the matte value of any neighbor pixel that is not the border color, Finally reset changes the matte value of all pixels.

4.39.31 MiterLimit(miterlimit as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify miter limit.
Notes: When two line segments meet at a sharp angle and miter joins have been specified for 'lineJoin', it is possible for the miter to extend far beyond the thickness of the line stroking the path. The miterLimit' imposes a limit on the ratio of the miter length to the 'lineWidth'. The default value of this parameter is 4.

4.39.32 PathArcAbs(c as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GMPathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.39.33 PathArcAbs(c() as GMPathArgsMBS) 397
- 4.39.34 PathArcAbs(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 398

4.39.33 PathArcAbs(c() as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GMPathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.39.32 PathArcAbs(c as GMPathArgsMBS) 397
- 4.39.34 PathArcAbs(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 398

4.39.34 PathArcAbs(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw an arc

draw.PathMovetoAbs 100,100
draw.PathArcAbs(100,100, 0, false, false, 200,200)
draw.DrawPath
draw.Draw

Backdrop=image.CopyPicture
```

Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.
See also:

- 4.39.32 PathArcAbs(c as GMPathArgsMBS) 397
- 4.39.33 PathArcAbs(c() as GMPathArgsMBS) 397

4.39.35 PathArcRel(c as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GMPathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.39.36 PathArcRel(c() as GMPathArgsMBS) 399
- 4.39.37 PathArcRel(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 400

4.39.36 PathArcRel(c() as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.

In the GMPathArgsMBS, set the following properties: radiusX, radiusY, xAxisRotation, bool largeArcFlag, sweepFlag, x and y.
See also:

- 4.39.35 PathArcRel(c as GMPathArgsMBS) 399
- 4.39.37 PathArcRel(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 400

4.39.37 PathArcRel(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws an elliptical arc from the current point to (x, y).
Notes: The size and orientation of the ellipse are defined by two radii (radiusX, radiusY) and an xAxisRotation, which indicates how the ellipse as a whole is rotated relative to the current coordinate system. The center (cx, cy) of the ellipse is calculated automatically to satisfy the constraints imposed by the other parameters. largeArcFlag and sweepFlag contribute to the automatic calculations and help determine how the arc is drawn. If largeArcFlag is true then draw the larger of the available arcs. If sweepFlag is true, then draw the arc matching a clock-wise rotation.
See also:

- 4.39.35 PathArcRel(c as GMPathArgsMBS) 399
- 4.39.36 PathArcRel(c() as GMPathArgsMBS) 399

4.39.38 PathClosePath

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Close the current subpath by drawing a straight line from the current point to current subpath's most recent starting point (usually, the most recent moveto point).

4.39.39 PathCurvetoAbs(c as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.40 PathCurvetoAbs(c() as GMPathArgsMBS) 401
- 4.39.41 PathCurvetoAbs(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 401

4.39.40 PathCurvetoAbs(c() as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.39 PathCurvetoAbs(c as GMPathArgsMBS) 400
- 4.39.41 PathCurvetoAbs(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 401

4.39.41 PathCurvetoAbs(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.39 PathCurvetoAbs(c as GMPathArgsMBS) 400
- 4.39.40 PathCurvetoAbs(c() as GMPathArgsMBS) 401

4.39.42 PathCurvetoRel(c as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.43 PathCurvetoRel(c() as GMPathArgsMBS) 402
- 4.39.44 PathCurvetoRel(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 402

4.39.43 PathCurvetoRel(c() as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.42 PathCurvetoRel(c as GMPathArgsMBS) 401
- 4.39.44 PathCurvetoRel(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 402

4.39.44 PathCurvetoRel(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Notes: PathCurvetoAbs indicates that absolutecoordinates will follow; PathCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.42 PathCurvetoRel(c as GMPathArgsMBS) 401
- 4.39.43 PathCurvetoRel(c() as GMPathArgsMBS) 402

4.39.45 PathLinetoAbs(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point.

PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.39.46 PathLinetoAbs(c() as GMCoordinateMBS) 403
- 4.39.47 PathLinetoAbs(x as Double, y as Double) 403

4.39.46 PathLinetoAbs(c() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.39.45 PathLinetoAbs(c as GMCoordinateMBS) 402
- 4.39.47 PathLinetoAbs(x as Double, y as Double) 403

4.39.47 PathLinetoAbs(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.39.45 PathLinetoAbs(c as GMCoordinateMBS) 402
- 4.39.46 PathLinetoAbs(c() as GMCoordinateMBS) 403

4.39.48 PathLinetoHorizontalAbs(v as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draws a horizontal line from the current point (cpx, cpy) to (x, cpy). PathLinetoHorizontalAbs indicates that absolute coordinates are supplied; PathLinetoHorizontalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (x, cpy) for the final value of x.

4.39.49 PathLinetoHorizontalRel(v as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draws a horizontal line from the current point (cpx, cpy) to (x, cpy). PathLinetoHorizontalAbs indicates that absolute coordinates are supplied; PathLinetoHorizontalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (x, cpy) for the final value of x.

4.39.50 PathLinetoRel(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.39.51 PathLinetoRel(c() as GMCoordinateMBS) 404
- 4.39.52 PathLinetoRel(x as Double, y as Double) 404

4.39.51 PathLinetoRel(c() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.39.50 PathLinetoRel(c as GMCoordinateMBS) 404
- 4.39.52 PathLinetoRel(x as Double, y as Double) 404

4.39.52 PathLinetoRel(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Example:

```
// new picture, 500x500 and filled with white
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)


dim draw as GMGraphicsMBS = image.Graphics

// Draw path

dim cr as new GMColorRGBMBS("red")
dim gr as new GMColorRGBMBS("green")
draw.StrokeColor cr
draw.FillColor gr
draw.PathMovetoAbs(30,10)
draw.PathLinetoAbs(20,55)
draw.PathLinetoAbs(70,50)
draw.PathLinetoAbs(80,5)
draw.DrawPath

draw.Draw

// show picture
image.type = image.TrueColorType // make sure it's a bitmap
Backdrop=image.CopyPicture
```

Notes: Draw a line from the current point to the given coordinate which becomes the new current point. PathLinetoAbs indicates that absolute coordinates are used; PathLinetoRel indicates that relative coordinates are used. A number of coordinates pairs may be specified in a list to draw a polyline. At the end of the command, the new current point is set to the final set of coordinates provided.
See also:

- 4.39.50 PathLinetoRel(c as GMCoordinateMBS) 404
- 4.39.51 PathLinetoRel(c() as GMCoordinateMBS) 404

4.39.53 PathLinetoVerticalAbs(v as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various "lineto" commands draw straight lines from the current point to a new point.
Notes: Draws a vertical line from the current point (cpx, cpy) to (cpx, y). PathLinetoVerticalAbs indicates that absolute coordinates are supplied; PathLinetoVerticalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (cpx, y) for the final value of y.

4.39.54 PathLinetoVerticalRel(v as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The various ”lineto” commands draw straight lines from the current point to a new point.
Notes: Draws a vertical line from the current point (cpx, cpy) to (cpx, y). PathLinetoVerticalAbs indicates that absolute coordinates are supplied; PathLinetoVerticalRel indicates that relative coordinates are supplied. At the end of the command, the new current point becomes (cpx, y) for the final value of y.

4.39.55 PathMovetoAbs(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.39.56 PathMovetoAbs(x as Double, y as Double) 406

4.39.56 PathMovetoAbs(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.39.55 PathMovetoAbs(c as GMCoordinateMBS) 406

4.39.57 PathMovetoRel(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.39.58 PathMovetoRel(x as Double, y as Double) 407

4.39.58 PathMovetoRel(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The ”moveto” commands establish a new current point.
Notes: The effect is as if the ”pen” were lifted and moved to a new location. A path data segment must begin with either one of the ”moveto” commands or one of the ”arc” commands. Subsequent ”moveto” commands (i.e., when the ”moveto” is not the first command) represent the start of a new subpath.

Start a new sub-path at the given coordinate. PathMovetoAbs indicates that absolute coordinates will follow; PathMovetoRel indicates that relative coordinates will follow. If a relative moveto appears as the first element of the path, then it is treated as a pair of absolute coordinates. If a moveto is followed by multiple pairs of coordinates, the subsequent pairs are treated as implicit lineto commands.
See also:

- 4.39.57 PathMovetoRel(c as GMCoordinateMBS) 407

4.39.59 PathQuadraticCurvetoAbs(c as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.60 PathQuadraticCurvetoAbs(c() as GMPathArgsMBS) 408
- 4.39.61 PathQuadraticCurvetoAbs(x1 as Double, y1 as Double, x as Double, y as Double) 408

4.39.60 PathQuadraticCurvetoAbs(c() as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.59 PathQuadraticCurvetoAbs(c as GMPathArgsMBS) 407
- 4.39.61 PathQuadraticCurvetoAbs(x1 as Double, y1 as Double, x as Double, y as Double) 408

4.39.61 PathQuadraticCurvetoAbs(x1 as Double, y1 as Double, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.59 PathQuadraticCurvetoAbs(c as GMPathArgsMBS) 407
- 4.39.60 PathQuadraticCurvetoAbs(c() as GMPathArgsMBS) 408

4.39.62 PathQuadraticCurvetoRel(c as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.63 PathQuadraticCurvetoRel(c() as GMPathArgsMBS) 409
- 4.39.64 PathQuadraticCurvetoRel(x1 as Double, y1 as Double, x as Double, y as Double) 409

4.39.63 PathQuadraticCurvetoRel(c() as GMPathArgsMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.62 PathQuadraticCurvetoRel(c as GMPathArgsMBS) 408
- 4.39.64 PathQuadraticCurvetoRel(x1 as Double, y1 as Double, x as Double, y as Double) 409

4.39.64 PathQuadraticCurvetoRel(x1 as Double, y1 as Double, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y) using (x1,y1) as the control point.
Notes: PathQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathQuadraticCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.62 PathQuadraticCurvetoRel(c as GMPathArgsMBS) 408

- 4.39.63 PathQuadraticCurvetoRel(c() as GMPathArgsMBS) 409

4.39.65 PathSmoothCurvetoAbs(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.66 PathSmoothCurvetoAbs(c() as GMCoordinateMBS) 410
- 4.39.67 PathSmoothCurvetoAbs(x as Double, y as Double) 411

4.39.66 PathSmoothCurvetoAbs(c() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.65 PathSmoothCurvetoAbs(c as GMCoordinateMBS) 410
- 4.39.67 PathSmoothCurvetoAbs(x as Double, y as Double) 411

4.39.67 PathSmoothCurvetoAbs(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.65 PathSmoothCurvetoAbs(c as GMCoordinateMBS) 410
- 4.39.66 PathSmoothCurvetoAbs(c() as GMCoordinateMBS) 410

4.39.68 PathSmoothCurvetoRel(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.69 PathSmoothCurvetoRel(c() as GMCoordinateMBS) 411
- 4.39.70 PathSmoothCurvetoRel(x as Double, y as Double) 412

4.39.69 PathSmoothCurvetoRel(c() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was

not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x2, y2, x and y.
See also:

- 4.39.68 PathSmoothCurvetoRel(c as GMCoordinateMBS) 411
- 4.39.70 PathSmoothCurvetoRel(x as Double, y as Double) 412

4.39.70 PathSmoothCurvetoRel(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a cubic B√©zier curve from the current point to (x,y).
Notes: The first control point is assumed to be the reflection of the second control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not an PathCurvetoAbs, PathCurvetoRel, PathSmoothCurvetoAbs or PathSmoothCurvetoRel, assume the first control point is coincident with the current point.) (x2,y2) is the second control point (i.e., the control point at the end of the curve). PathSmoothCurvetoAbs indicates that absolute coordinates will follow; PathSmoothCurvetoRel indicates that relative coordinates will follow. Multiple sets of coordinates may be specified to draw a polybezier. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.68 PathSmoothCurvetoRel(c as GMCoordinateMBS) 411
- 4.39.69 PathSmoothCurvetoRel(c() as GMCoordinateMBS) 411

4.39.71 PathSmoothQuadraticCurvetoAbs(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous
command relative to the current point. (If there is no previous command or if the previous command was not
a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.72 PathSmoothQuadraticCurvetoAbs(c() as GMCoordinateMBS) 413
- 4.39.73 PathSmoothQuadraticCurvetoAbs(x as Double, y as Double) 413

4.39.72 PathSmoothQuadraticCurvetoAbs(c() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.71 PathSmoothQuadraticCurvetoAbs(c as GMCoordinateMBS) 412
- 4.39.73 PathSmoothQuadraticCurvetoAbs(x as Double, y as Double) 413

4.39.73 PathSmoothQuadraticCurvetoAbs(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.71 PathSmoothQuadraticCurvetoAbs(c as GMCoordinateMBS) 412
- 4.39.72 PathSmoothQuadraticCurvetoAbs(c() as GMCoordinateMBS) 413

4.39.74 PathSmoothQuadraticCurvetoRel(c as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.75 PathSmoothQuadraticCurvetoRel(c() as GMCoordinateMBS) 414
- 4.39.76 PathSmoothQuadraticCurvetoRel(x as Double, y as Double) 414

4.39.75 PathSmoothQuadraticCurvetoRel(c() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.

In the GMPathArgsMBS object, set the following properties: x1, y1, x and y.
See also:

- 4.39.74 PathSmoothQuadraticCurvetoRel(c as GMCoordinateMBS) 414
- 4.39.76 PathSmoothQuadraticCurvetoRel(x as Double, y as Double) 414

4.39.76 PathSmoothQuadraticCurvetoRel(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draws a quadratic B√©zier curve from the current point to (x,y).
Notes: The control point is assumed to be the reflection of the control point on the previous

command relative to the current point. (If there is no previous command or if the previous command was not a PathQuadraticCurvetoAbs, PathQuadraticCurvetoRel, PathSmoothQuadraticCurvetoAbs or PathSmoothQuadraticCurvetoRel, assume the control point is coincident with the current point.) PathSmoothQuadraticCurvetoAbs indicates that absolute coordinates will follow; PathSmoothQuadraticCurvetoRel indicates that relative coordinates will follow. At the end of the command, the new current point becomes the final (x,y) coordinate pair used in the polybezier.
See also:

- 4.39.74 PathSmoothQuadraticCurvetoRel(c as GMCoordinateMBS) 414
- 4.39.75 PathSmoothQuadraticCurvetoRel(c() as GMCoordinateMBS) 414

4.39.77 Point(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a point using stroke color and thickness at coordinate.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.fillColor = new GMColorRGBMBS("red") // set color

dim draw as GMGraphicsMBS = image.Graphics

// draw cross with pixels
for x as Integer = 240 to 260
draw.Point(x, 250)
next
for y as Integer = 240 to 260
draw.Point(250,y)
next
draw.Draw

Backdrop=image.CopyPicture
```

4.39.78 PointSize(pointSize as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set font point size.

4.39.79 Polygon(values() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an arbitrary polygon using stroke color and thickness consisting of three or more coordinates contained in an array.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.fillColor = new GMColorRGBMBS("red") // set color
image.strokeColor = new GMColorRGBMBS("green") // set color

dim draw as GMGraphicsMBS = image.Graphics
dim coordinates(-1) as GMCoordinateMBS

coordinates.Append new GMCoordinateMBS(70,70)
coordinates.Append new GMCoordinateMBS(100,340)
coordinates.Append new GMCoordinateMBS(380,200)
coordinates.Append new GMCoordinateMBS(70,70)

draw.Polygon coordinates
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.39.80 Polyline(values() as GMCoordinateMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw an arbitrary polyline using stroke color and thickness consisting of three or more coordinates contained in an array.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("green") // set color

dim draw as GMGraphicsMBS = image.Graphics
dim coordinates(-1) as GMCoordinateMBS
```

```
coordinates.Append new GMCoordinateMBS(70,70)
coordinates.Append new GMCoordinateMBS(100,340)
coordinates.Append new GMCoordinateMBS(380,200)

draw.Polyline coordinates
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.39.81 PopClipPath

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pop (terminate) clip path definition started by PushClipPath.

4.39.82 PopGraphicContext

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pop Graphic Context.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorMBS() // transparent fillcolor
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a Rectangle
draw.PushGraphicContext
draw.Translation(250,250)
draw.Rotation(50)
draw.Rectangle(0, 0, 100, 100) // rotated
draw.PopGraphicContext
draw.Rectangle(0, 0, 100, 100) // not rotated
draw.Draw
```

```
Backdrop=image.CopyPicture
```

Notes: Removing the current graphic context from the graphic context stack restores the options to the values they had prior to the preceding PushGraphicContext operation.

4.39.83 PopPattern

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Terminate a pattern definition started via PushPattern.

4.39.84 PushClipPath(id as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Push (create) clip path definition with id.
Notes: Clip patch definition consists of subsequent drawing commands, terminated by PopClipPath.

4.39.85 PushGraphicContext

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Push Graphic Context.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorMBS() // transparent fillcolor
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a Rectangle
draw.PushGraphicContext
draw.Translation(250,250)
draw.Rotation(50)
draw.Rectangle(0, 0, 100, 100) // rotated
draw.PopGraphicContext
draw.Rectangle(0, 0, 100, 100) // not rotated
draw.Draw

Backdrop=image.CopyPicture
```

Notes: When a graphic context is pushed, options set after the context is pushed (such as coordinate transformations, color settings, etc.) are saved to a new graphic context. This allows related options to be saved on a graphic context "stack" in order to support heirarchical nesting of options. When PopGraphicContext is used to pop the current graphic context, the options in effect during the last PushGraphicContext operation are restored.

4.39.86 PushPattern(id as string, x as Integer, y as Integer, width as Integer, height as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Start a pattern definition with arbitrary pattern name specified by id, pattern offset specified by x and y, and pattern size specified by width and height.
Notes: The pattern is defined within the coordinate system defined by the specified offset and size. Arbitrary drawing objects (including DrawableCompositeImage) may be specified between PushPattern and PopPattern in order to draw the pattern. Normally the pair PushGraphicContext & PopGraphicContext are used to enclose a pattern definition. Pattern definitions are terminated by a PopPattern object.

4.39.87 Rectangle(upperLeftX as Double, upperLeftY as Double, lowerRightX as Double, lowerRightY as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a rectangle using stroke color and thickness from upper-left coordinates to lower-right coordinates.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a rectangle
```

```
draw.Rectangle(250, 250, 100, 100)
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.39.88 Rotation(angle as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set rotation to use when drawing (coordinate transformation).
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.StrokeColor new GMColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Rotation 5
draw.StrokeColor new GMColorRGBMBS("blue")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.89 RoundRectangle(centerX as Double, centerY as Double, width as Double, height as Double, cornerWidth as Double, cornerHeight as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a rounded rectangle using stroke color and thickness, with specified center coordinate, specified width and height, and specified corner width and height.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)
```

```
image.strokeColor = new GMColorRGBMBS(”red”) // Outline color
image.fillColor = new GMColorRGBMBS(”green”) // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a round rectangle
draw.RoundRectangle(250, 250, 100, 100,20,20)
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If a fill color is specified, then the object is filled.

4.39.90 Scaling(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply scaling in x and y direction while drawing objects (coordinate transformation).
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS(”white”) // white
dim image as new GMImageMBS(g, c)

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics
draw.FillColor new GMColorRGBMBS(”red”)

draw.StrokeColor new GMColorRGBMBS(”red”)
draw.Line(100,100,400,400)
draw.StrokeColor new GMColorRGBMBS(”blue”)
draw.Scaling 1.2,1.1
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.91 SkewX(angle as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply Skew in X direction (coordinate transformation)
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.StrokeColor new GMColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.SkewX 5
draw.StrokeColor new GMColorRGBMBS("blue")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.92 SkewY(angle as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply Skew in Y direction.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.StrokeColor new GMColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.SkewY 5
draw.StrokeColor new GMColorRGBMBS("blue")
draw.Line(100,100,400,400)
draw.Draw
```

```
Backdrop=image.CopyPicture
```

4.39.93 StrokeAntialias(flag as boolean)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Antialias while drawing lines or object outlines.

4.39.94 StrokeColor(c as GMColorMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set color to use when drawing lines or object outlines.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.StrokeColor new GMColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.95 StrokeLineCap(LineCap as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the end of open subpaths when they are stroked.
Notes: Values of LineCap are UndefinedCap, ButtCap, RoundCap, and SquareCap.

4.39.96 StrokeLineJoin(LineJoin as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the corners of paths (or other vector shapes) when they are stroked.
Notes: Values of LineJoin are UndefinedJoin, MiterJoin, RoundJoin, and BevelJoin.

4.39.97 StrokeOpacity(opacity as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Opacity to use when drawing lines or object outlines.

4.39.98 StrokeWidth(opacity as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set width to use when drawing lines or object outlines.

4.39.99 Text(x as Double, y as Double, text as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate image with text using stroke color, font, font pointsize, and box color (text background color), at specified coordinates.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

dim draw as GMGraphicsMBS = image.Graphics

// draw red text
draw.strokeColor(new GMColorRGBMBS("red")) // Outline color
draw.strokeWidth(1)
draw.Font("/Library/Fonts/Verdana.ttf")
draw.Text(50, 50, "Hello")
draw.Draw

Backdrop=image.CopyPicture
```

Notes: If text contains special format characters the image filename, type, width, height, or other image attributes may be incorporated in the text (see label).
See also:

- 4.39.100 Text(x as Double, y as Double, text as string, encoding as string) 425

4.39.100 Text(x as Double, y as Double, text as string, encoding as string)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate image with text represented with text encoding, using current stroke color, font, font pointsize, and box color (text background color), at specified coordinates.
Notes: If text contains special format characters the image filename, type, width, height, or other image attributes may be incorporated in the text (see label()).

The text encoding specifies the code set to use for text annotations. The only character encoding which may be specified at this time is "UTF-8" for representing Unicode as a sequence of bytes. Specify an empty string to set text encoding to the system's default. Successful text annotation using Unicode may require fonts designed to support Unicode.

Seems like you need ghostscript or the DPS library for text handling, so it may no be available for you.
See also:

- 4.39.99 Text(x as Double, y as Double, text as string) 424

4.39.101 TextAntialias(flag as boolean)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Antialias while drawing text (default true).
Notes: The main reason to disable text antialiasing is to avoid adding new colors to the image.

4.39.102 TextDecoration(DecorationType as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify decoration (e.g. UnderlineDecoration) to apply to text.

4.39.103 TextUnderColor(c as GMColorMBS)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a box under rendered text using the specified color.

4.39.104 Translation(x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply coordinate translation (set new coordinate origin).
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

draw.StrokeColor new GMColorRGBMBS("red")
draw.Line(100,100,400,400)
draw.Translation 5,5
draw.StrokeColor new GMColorRGBMBS("blue")
draw.Line(100,100,400,400)
draw.Draw

Backdrop=image.CopyPicture
```

4.39.105 Viewbox(x1 as Integer, y1 as Integer, x2 as Integer, y2 as Integer)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Dimensions of the output viewbox.
Notes: If the image is to be written to a vector format (e.g. MVG or SVG), then a PushGraphicContext() object should be pushed to the head of the list, followed by a Viewbox() statement to establish the output canvas size. A matching PopGraphicContext() object should be pushed to the tail of the list.

4.39.106 Properties

4.39.107 Image as GMImageMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image this graphics object belongs to.
Notes: (Read only property)

4.40 class GMImageArrayMBS

4.40.1 class GMImageArrayMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an array of images in GraphicsMagick.
Example:

```
// extract all layers of photoshop file
dim file as FolderItem = SpecialFolder.Desktop.Child("test.psd")
dim images as new GMImageArrayMBS

images.readImages(file.NativePath)

dim c as Integer = images.size
for i as Integer = 0 to c-1
dim image as GMImageMBS = images.Image(i)
file = SpecialFolder.Desktop.Child(image.FileName+"."+str(i)+".png")
image.write(file)
next
```

Notes: Can be used to assemble/disassemble gif images.
Blog Entries

- MBS Real Studio Plugins, version 12.3pr4

4.40.2 Methods

4.40.3 animateImages

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Animate a sequence of image frames.
Notes: Image frames are displayed in succession, creating an animated effect. The animation options are taken from the first image frame. This feature is only supported under X11 at the moment.

4.40.4 append(image as GMImageMBS)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds an image to the end of the array.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// put copy of first image on the back
dim n as GMImageMBS = g.FirstImage
g.append n

// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

Notes: Instead of gif, you can also use tif files.

4.40.5 appendImages(stack as boolean = false) as GMImageMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Append a sequence of image frames, writing the result to new image.
Notes: All the input image frames must have the same width or height. Image frames of the same width are stacked top-to-bottom. Image frames of the same height are stacked left-to-right. If the stack parameter is false, rectangular image frames are stacked left-to-right otherwise top-to-bottom.

4.40.6 averageImages as GMImageMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Average a sequence of image frames, writing the result to averagedImage.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// averageImages
dim n as GMImageMBS = g.averageImages
Backdrop = n.CopyPicture
```

Notes: All the input image frames must be the same size in pixels.

4.40.7 coalesceImages as GMImageArrayMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create a coalesced image sequence obtained by "playing" the image sequence (observing page offsets and disposal methods) to create a new image sequence in which all frames are full size and completely rendered.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// deconstruct
g = g.coalesceImages

// write gif
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")
g.writeImages output.NativePath
```

Notes: Note that if the original image sequence relied on page offsets and disposal methods that the resulting sequence will be larger (perhaps much larger) then the original. This is useful for GIF animation sequences that have page offsets and disposal methods. The resuting image sequence is returned.

4.40.8 Constructor

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty image array.

4.40.9 deconstructImages as GMImageArrayMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Break down an image sequence into constituent parts.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// deconstruct
g = g.deconstructImages

// write gif
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")
g.writeImages output.NativePath
```

Notes: This is useful for creating GIF or MNG animation sequences.

4.40.10 displayImages

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Display a sequence of image frames.
Notes: Through use of a pop-up menu, image frames may be selected in succession. This feature is fully supported under X11 but may have only limited support in other environments.
Caution: if an image format is is not compatable with the display visual (e.g. JPEG on a colormapped display) then the original image will be altered. Use a copy of the original if this is a problem.

display methods are not supported currently.

4.40.11 FirstImage as GMImageMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns first image in array.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// show first image
dim img as GMImageMBS = g.FirstImage

// convert to true color for CopyPicture to work
const TrueColorType=6
img.type=TrueColorType
```

```
Backdrop = img.CopyPicture
```

4.40.12 flattenImages as GMImageMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Merge a sequence of image frames which represent image layers into a single composited representation.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// put copy of first image on the back
dim n as GMImageMBS = g.flattenImages

Backdrop = n.CopyPicture
```

Notes: Returns the flattened image. This function is useful for combining Photoshop layers into a single image.

4.40.13 Image(index as Integer) as GMImageMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries image with given index.

4.40.14 insert(image as GMImageMBS)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Inserts an image on the front.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)
```

```
// put copy of first image on the front
dim n as GMImageMBS = g.FirstImage
g.insert n

// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

4.40.15 LastImage as GMImageMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns last image in array.

4.40.16 mapImages(map as GMImageMBS, dither as boolean = true, measureError as boolean = false)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replace the colors of a sequence of images with the closest color from a reference image.
Notes: Set dither to true to enable dithering. Set measureError to true in order to evaluate quantization error.

4.40.17 montageImages(options as GMMontageMBS) as GMImageArrayMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create a composite image by combining several separate image frames.
Example:

```
// build montage
dim StackingMontage as New GM16MontageMBS
StackingMontage.backgroundColor = New GM16ColorMBS(&cE7E7E7)
StackingMontage.fillColor = New GM16ColorMBS(&c000000)
StackingMontage.tile = New GM16GeometryMBS("1x20")
StackingMontage.geometry = New GM16GeometryMBS("160x120+5+5")
StackingMontage.font = "Helvetica"
StackingMontage.pointSize = 12
StackingMontage.title = "Title goes here"
```

```
// make picture
dim logo as Picture = LogoMBS(500)
dim image as New GM16ImageMBS(logo)

image.label("Sample label")

// Put the current image into the array
Dim StackingFrames As new GM16ImageArrayMBS
StackingFrames.insert(image)

// show result
dim resultImages as GM16ImageArrayMBS = StackingFrames.montageImages(StackingMontage)
Backdrop = resultImages.Image(0).CopyPicture
```

Notes: Multiple frames may be generated in the output array depending on the tile setting and the number of image frames montaged. Montage options are provided via the parameter options. Options set in the first image frame (backgroundColor, borderColor, matteColor, fillColor, strokeColor, font and fontPointsize) are also used as options by montageImages().

4.40.18 morphImages(frames as Integer) as GMImageArrayMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Morph a seqence of image frames.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// coalesce to make sure we have full images
g = g.coalesceImages
// morph to 10 pictures
g = g.morphImages(10)

// write gif
dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")
g.writeImages output.NativePath
```

Notes: This algorithm expands the number of image frames (output to the new image array) by adding the number of intervening frames specified by frames such that the original frames morph (blend) into each other when played as an animation.

4.40.19 mosaicImages as GMImageMBS

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Inlay a number of images to form a single coherent picture.
Notes: The result image argument is updated with a mosaic constructed from the image sequence.

4.40.20 quantizeImages(measureError as boolean = false)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Quantize colors in images using current quantization settings.
Notes: Set measureError to true in order to measure quantization error.

4.40.21 readImages(blob as GMBlobMBS)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read a sequence of image frames into existing container (appending to array) from blob.
See also:

- 4.40.22 readImages(imageSpec as string) 435

4.40.22 readImages(imageSpec as string)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read a sequence of image frames into existing container (appending to array) with image names specified in the string imageSpec.
See also:

- 4.40.21 readImages(blob as GMBlobMBS) 435

4.40.23 remove(index as Integer)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes the image with the given index.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// remove first
g.remove 0

// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

Notes: Index should be between 0 and size-1.

4.40.24 reverse

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reverses the order of images in the array.

4.40.25 writeImages(blob as GMBlobMBS, adjoin as boolean = true)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Writes images to the given blob object.
Notes: Write images in container to in-memory BLOB specified by Blob blob. Set adjoin to false to write a set of image frames via a wildcard imageSpec (e.g. image%02d.miff).
Caution: if an image format is selected which is capable of supporting fewer colors than the original image or quantization has been requested, the original image will be quantized to fewer colors. Use a copy of the original if this is a problem.
See also:

- 4.40.26 writeImages(imageSpec as string, adjoin as boolean = true) 436

4.40.26 writeImages(imageSpec as string, adjoin as boolean = true)

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Writes images to the given path.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// write to file

dim output as FolderItem = SpecialFolder.Desktop.Child("output.gif")

g.writeImages(output.NativePath)
```

Notes: Write images in container to file specified by string imageSpec. Set adjoin_ to false to write a set of image frames via a wildcard imageSpec (e.g. image%02d.miff).
The wildcard must be one of %0Nd, %0No, or %0Nx.
Caution: if an image format is selected which is capable of supporting fewer colors than the original image or quantization has been requested, the original image will be quantized to fewer colors. Use a copy of the original if this is a problem.
See also:

- 4.40.25 writeImages(blob as GMBlobMBS, adjoin as boolean = true) 436

4.40.27 Properties

4.40.28 empty as boolean

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Checks whether image array is empty.
Notes: Returns true if array is empty or false if not.
(Read only property)

4.40.29 handle as Integer

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal handle of the image array.
Notes: Should always be non zero.
(Read and Write property)

4.40.30 size as Integer

Plugin Version: 12.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns number of images in this array.
Example:

```
// read gif
dim g as new GMImageArrayMBS
dim file as FolderItem = SpecialFolder.Desktop.Child("test.gif")
g.readImages(file.NativePath)

// display number of images
MsgBox str(g.size)
```

Notes: (Read only property)

4.41 class GMImageChannelStatisticsMBS

4.41.1 class GMImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The statistics for image channel.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim stat as GMImageStatisticsMBS = g.statistics
dim gs as GMImageChannelStatisticsMBS = stat.blue

MsgBox "blue channel: "+str(gs.minimum)+"-"+str(Gs.maximum)+", mean "+str(gs.mean)
```

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.41.2 Methods

4.41.3 Constructor

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The private constructor.

4.41.4 Properties

4.41.5 maximum as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Maximum value observed.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim stat as GMImageStatisticsMBS = g.statistics
dim gs as GMImageChannelStatisticsMBS = stat.green

MsgBox "maximum green color: "+str(gs.maximum)
```

Notes: (Read only property)

4.41.6 mean as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Average (mean) value observed.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim stat as GMImageStatisticsMBS = g.statistics
dim r as GMImageChannelStatisticsMBS = stat.red

MsgBox "mean red color: "+str(R.mean)
```

Notes: (Read only property)

4.41.7 minimum as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Minimum value observed.
Notes: (Read only property)

4.41.8 standardDeviation as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Standard deviation, sqrt(variance).
Notes: (Read only property)

4.41.9 variance as Double

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Variance.

Notes: (Read only property)

4.42 class GMImageMBS

4.42.1 class GMImageMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image is the primary object in Magick++ and represents a single image frame (see image design).
Example:

```
dim c as new GMColorMBS(”white”)
dim g as new GMGeometryMBS(100,100)
dim image as new GMImageMBS(g, c)
```

Notes: With MBS Plugin 14.0 we offer this classes in 8bit (GM prefix) or 16bit (GM16 prefix).

The GMImageArrayMBS class must be used to operate on image sequences or images (e.g. of format GIF, TIFF, MIFF, Postscript, & MNG) which are comprized of multiple image frames. Individual frames of a multi-frame image may be requested by adding array-style notation to the end of the file name (e.g. ”animation.gif [3] ” retrieves the fourth frame of a GIF animation. Various image manipulation operations may be applied to the image. Attributes may be set on the image to influence the operation of the manipulation operations. The GMPixelsMBS class provides low-level access to image pixels.
Blog Entries

- News from the MBS Xojo Plugins Version 23.3
- MonkeyBread Software Releases the MBS Xojo Plugins in version 23.3
- News from the MBS Xojo Plugins Version 23.2
- Several ways for picture to PDF in MBS Plugins
- News from the MBS Xojo Plugins Version 20.4
- News from the MBS Xojo Plugins Version 20.1
- MonkeyBread Software Releases the MBS Xojo Plugins in version 20.1
- Crop a two side page document to a single page document
- Four ways to save picture as Tiff in Xojo
- Gradients in GraphicsMagick

Videos

- Presentation from Xojo Developer Conference 2019 in Miami.

Xojo Developer Magazine

- 21.5, page 10: News
- 19.6, page 73: Fun with GraphicsMagick, Cool methods from the GMImageMBS class by Stefanie Juchmes
- 19.6, pages 68 to 69: Fun with GraphicsMagick, Cool methods from the GMImageMBS class by Stefanie Juchmes
- 19.3, page 10: News
- 18.6, page 10: News
- 18.3, page 10: News
- 17.5, page 39: What's New in the MBS Plugins, With the Plugins growing every year, here are new capabilities you may have missed by Stefanie Juchmes

4.42.2 Methods

4.42.3 adaptiveThreshold(width as UInt32, height as UInt32, offset as double = 0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply adaptive thresholding to the image.
Notes: see:
http://www.dai.ed.ac.uk/HIPR2/adpthrsh.htm

Adaptive thresholding is useful if the ideal threshold level is not known in advance, or if the illumination gradient is not constant across the image. Adaptive thresholding works by evaulating the mean (average) of a pixel region (size specified by width and height) and using the mean as the thresholding value. In order to remove residual noise from the background, the threshold may be adjusted by subtracting a constant offset (default zero) from the mean to compute the threshold.

4.42.4 addNoise(noise as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add noise to image with the specified noise type.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.addNoise(image.GaussianNoise)

Backdrop=image.CopyPicture
```

Notes: Use one of this constants: GaussianNoise, ImpulseNoise, LaplacianNoise, MultiplicativeGaussianNoise, PoissonNoise, UniformNoise.

4.42.5 addNoiseChannel(channel as Integer, noise as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add noise to an image channel with the specified noise type. The channel parameter specifies the channel to add noise to.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.addNoiseChannel(image.BlueChannel, image.ImpulseNoise)

Backdrop=image.CopyPicture
```

Notes: The noiseType parameter specifies the type of noise.
Use one of this constants: GaussianNoise, ImpulseNoise, LaplacianNoise, MultiplicativeGaussianNoise, PoissonNoise, UniformNoise.

4.42.6 affineTransform(sx as Double, sy as Double, rx as Double, ry as Double, tx as Double, ty as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies an affine transformation to the drawing matrix.
Notes: Specify a transformation matrix to adjust scaling, rotation, and translation (coordinate transformation) for subsequently drawn objects in the same or decendent drawing context. The sx & sy parameters represent the x & y scale factors, the rx & ry parameters represent the x & y rotation, and the tx & ty parameters represent the x & y translation.

4.42.7 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate using specified text, bounding area, and placement gravity.

Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.42.8 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer, degrees as Double) 445
- 4.42.9 annotate(text as string, gravity as Integer) 446
- 4.42.10 annotate(text as string, location as GMGeometryMBS) 448

4.42.8 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer, degrees as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate with text using specified text, bounding area, placement gravity, and rotation.
Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.42.7 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer) 444
- 4.42.9 annotate(text as string, gravity as Integer) 446
- 4.42.10 annotate(text as string, location as GMGeometryMBS) 448

4.42.9 annotate(text as string, gravity as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate with text (bounding area is entire image) and placement gravity.
Example:

```
dim White as new GMColorGrayMBS(1)
dim Black as new GMColorGrayMBS(0)
dim geo as new GMGeometryMBS("300x200")

dim g as new GMImageMBS(geo, White)
```

```
g.antiAlias = False
g.fillColor = Black
g.lineWidth = 1
g.strokeColor = Black
g.font = "@/Library/Fonts/Tahoma.ttf"
g.fontPointsize = 15

g.annotate("Hello World", g.SouthGravity)

Backdrop = g.CopyPicture
```

Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.42.7 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer) 444
- 4.42.8 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer, degrees as Double) 445

4.42.10 annotate(text as string, location as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotate using specified text, and placement location.
Notes: Annotate image (draw text on image)

Gravity effects text placement in bounding area according to these rules:

NorthWestGravity	text bottom-left corner placed at top-left
NorthGravity	text bottom-center placed at top-center
NorthEastGravity	text bottom-right corner placed at top-right
WestGravity	text left-center placed at left-center
CenterGravity	text center placed at center
EastGravity	text right-center placed at right-center
SouthWestGravity	text top-left placed at bottom-left
SouthGravity	text top-center placed at bottom-center
SouthEastGravity	text top-right placed at bottom-right

Annotate annotates an image with text. Optionally you can include any of the following bits of information about the image by embedding the appropriate special characters:

%b file size in bytes. %c comment. %d directory in which the image resides. %e extension of the image file. %f original filename of the image. %h height of image. %i filename of the image. %k number of unique colors. %l image label. %m image file format. %n number of images in a image sequence. %o output image filename. %p page number of the image. %q image depth (8 or 16). %p page number of the image. %q image depth (8 or 16). %s image scene number. %t image filename without any extension. %u a unique temporary filename. %w image width. %x x resolution of the image. %y y resolution of the image.

Set a font with full path and @ in front. e.g. "@/Library/Fonts/Arial.ttf". This way the plugin loads it directly.
See also:

- 4.42.7 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer) 444
- 4.42.8 annotate(text as string, boundingArea as GMGeometryMBS, gravity as Integer, degrees as Double) 445
- 4.42.9 annotate(text as string, gravity as Integer) 446

4.42.11 attributeValues as dictionary

Plugin Version: 17.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A dictionary with all attributes.
Notes: As attributes are created on demand, this will only return all so far generated attributes.

4.42.12 autoOrient

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Orient image to be right-side up based on its current orientation attribute.
Notes: This allows the image to be viewed correctly when the orientation attribute is not available, or is not respected.

4.42.13 blur(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur an image with the specified blur factor.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.blur(30,10)

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.42.14 blurChannel(channel as Integer, radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur an image channel with the specified blur factor.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)
```

```
image.blurChannel(image.BlueChannel, 30,10)

Backdrop=image.CopyPicture
```

Notes: The channel parameter specifies the channel to modify. The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.42.15 border

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Border image (add border to image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.border

Backdrop=image.CopyPicture
```

Notes: The color of the border is specified by the borderColor attribute.
See also:

- 4.42.16 border(geometry as GMGeometryMBS) 450

4.42.16 border(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Border image (add border to image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.border GMGeometryMBS.Make(10,10)

Backdrop=image.CopyPicture
```

Notes: The color of the border is specified by the borderColor attribute.
See also:

- 4.42.15 border 450

4.42.17 borderGeometryDefault as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The default geometry description for border.

4.42.18 cacheThreshold(threshold as UInt32)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pixel cache threshold in megabytes.
Notes: Once this memory threshold is exceeded, all subsequent pixels cache operations are to/from disk. This setting is shared by all Image objects.

4.42.19 cdl(cdl as string)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Bake in the ASC-CDL.
Notes: Bake in the ASC-CDL, which is a convention for the for the exchange of basic primary color grading information between for the exchange of basic primary color grading information between equipment and software from different manufacturers. It is a useful transform for other purposes as well.

4.42.20 channel(channel as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extract channel from image.
Notes: Use this option to extract a particular channel from the image. MatteChannel for example, is useful for extracting the opacity values from an image.

4.42.21 charcoal(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Charcoal effect image (looks like charcoal sketch).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.charcoal

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.42.22 chop(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chop image (remove vertical or horizontal subregion of image).

4.42.23 colorHistogram as dictionary

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Calculates histogram.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim d as Dictionary = g.colorHistogram

MsgBox str(d.Count)+" color"

// check first color
dim c as GMColorMBS = d.key(0)

MsgBox "Color "+str(c.colorValue)+": "+str(d.Value(c))
```

Notes: The dictionary has a GMColorMBS/GMColor16MBS object as key for each color and an unsigned integer as value.

4.42.24 colorize(opacity as UInt32, penColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorize image with pen color, using specified percent opacity.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.colorize(10, new GMColorMBS("red"))

Backdrop=image.CopyPicture
```

See also:

- 4.42.25 colorize(opacityRed as UInt32, opacityGreen as UInt32, opacityBlue as UInt32, penColor as GMColorMBS) 453

4.42.25 colorize(opacityRed as UInt32, opacityGreen as UInt32, opacityBlue as UInt32, penColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorize image with pen color, using specified percent opacity for red, green, and blue quantums.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.colorize(10, 0, 5, new GMColorMBS("red"))

Backdrop=image.CopyPicture
```

See also:

- 4.42.24 colorize(opacity as UInt32, penColor as GMColorMBS) 453

4.42.26 colorMap as GMColorMBS()

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries array with all colors in color map.
See also:

4.42.27 colorMatrix(order as Integer, ColorMatrix() as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply a color matrix to the image channels.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)

dim m(8) as Double

m(0) = 0.25
m(1) = 0
m(2) = 0.25

m(3) = 0
m(4) = 0
m(5) = 0

m(6) = 0.25
m(7) = 0
m(8) = 0.25

g.colorMatrix 3, m

Backdrop = g.CopyPicture
```

Notes: The user supplied matrix may be of order 1 to 5 (1x1 through 5x5).

4.42.28 CombinePictureWithMask as picture

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image with mask.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

dim c as new GMColorMBS("white")
image.transparent(c)
```

Backdrop=image.CombinePictureWithMask

Notes: Internally this calls Width and Height, CopyPicture and CopyMask.

4.42.29 compare(image as GMImageMBS) as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compare current image with another image.
Notes: Sets meanErrorPerPixel, normalizedMaxError, and normalizedMeanError in the current image. False is returned if the images are identical. An ErrorOption exception is thrown if the reference image columns, rows, colorspace, or matte differ from the current image:

4.42.30 composite(compositeImage as GMImageMBS, gravity as Integer, CompositeOperator as Integer = 2)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose an image onto another at specified x and y offset and using a specified algorithm.

4.42.31 compositeAt(compositeImage as GMImageMBS, offset as GMGeometryMBS, CompositeOperator as Integer = 2)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose an image onto another at specified x and y offset and using a specified algorithm.

4.42.32 compositeXY(compositeImage as GMImageMBS, xOffset as Integer, yOffset as Integer, CompositeOperator as Integer = 2)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose an image onto another at specified x and y offset and using a specified algorithm.

4.42.33 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Default constructor.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GMImageMBS
dim blob as new GMBlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

See also:

- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.34 Constructor(blob as GMBlobMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image from in-memory Blob.
See also:

- 4.42.33 Constructor 455

- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462

4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size from in-memory Blob.
See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size and depth from in-memory Blob.
See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size, depth, and format from in-memory Blob.
See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461

- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct Image of specified size, depth, and format from in-memory Blob.
Example:

```
// first open file and read data in blob
SVG_File = FolderItem.ShowOpenFileDialog("")
If SVG_File = Nil Then Return

dim stream as BinaryStream = BinaryStream.Open(SVG_File)
dim data as string = stream.Read(stream.Length)
dim blob as new GMBlobMBS(data)

// 400 width and 400 height
Dim geo As New GMGeometryMBS(400, 400, 0, 0)

// pass type here to have GraphicsMagick know it since there is no file name in blob:
dim g as New GMImageMBS(blob, geo, "svg")
```

See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.39 Constructor(file as folderitem)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct from image file.
See also:

- 4.42.33 Constructor 455
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.40 Constructor(other as GMImageMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an image by making a copy of the existing one.
See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462

4.42.41 Constructor(Path as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct from image file or image specification.
See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.42 Constructor(pic as picture)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new GMImage with the given picture.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)
Backdrop=image.CopyPicture
```

Notes: Pixels from both the picture and picture's mask.
See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458

- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Construct a blank image canvas of specified size and color.
Example:

```
dim g as new GMGeometryMBS(600,600)
dim c as new GMColorRGBMBS(1.0,0.0,0.0) // red
dim image as new GMImageMBS(g, c)

const TrueColorType=6

// Ensure that there are no other references to this image.
image.modifyImage
// Set the image type to TrueColor DirectClass representation.
image.type=TrueColorType

Backdrop=image.CopyPicture(0,0,600,600)
```

See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.37 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32, Magick as string) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460

- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 463

4.42.44 Constructor(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes single image frame from an array of raw pixels, with specified storage type (ConstituteImage).
Example:

```
dim data as new memoryblock(2048*2048) // your data
dim image as new GMImageMBS(2048, 2048, "I", GMImageMBS.StorageTypeCharPixel, data)
```

Notes: Returns an Image corresponding to an image stored in a raw memory array format. The pixel data must be in scanline order top-to-bottom. The data can be unsigned char, unsigned short int, unsigned int, unsigned long, float, or double. Float and double require the pixels to be normalized to the range [0..1] , otherwise the range is [0..MaxVal] where MaxVal is the maximum possible value for that type.

Note that for most 32-bit architectures the size of an unsigned long is the same as unsigned int, but for 64-bit architectures observing the LP64 standard, an unsigned long is 64 bits, while an unsigned int remains 32 bits. This should be considered when deciding if the data should be described as "Integer" or "Long".

For example, to create a 640x480 image from unsigned red-green-blue character data, use

image = new GMImageMBS(640, 480, "RGB", GMImageMBS.StorageTypeCharPixel, pixels);

width: width in pixels of the image.
height: height in pixels of the image.
map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (same as Transparency), O = Opacity, T = Transparency, C = cyan, Y = yellow, M = magenta, K = black, or I = intensity (for grayscale). Specify "P" = pad, to skip over a quantum which is intentionally ignored. Creation of an alpha channel for CMYK images is currently not supported.
type: Define the data type of the pixels. Float and double types are expected to be normalized [0..1] otherwise [0..MaxRGB] . Choose from these types: StorageTypeCharPixel, StorageTypeShortPixel, StorageTypeIntegerPixel, StorageTypeLongPixel, StorageTypeFloatPixel, or StorageTypeDoublePixel.
pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.

See also:

- 4.42.33 Constructor 455
- 4.42.34 Constructor(blob as GMBlobMBS) 456
- 4.42.35 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS) 457
- 4.42.36 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, depth as UInt32) 458
- 4.42.38 Constructor(blob as GMBlobMBS, geometry as GMGeometryMBS, Magick as string) 459
- 4.42.39 Constructor(file as folderitem) 460
- 4.42.40 Constructor(other as GMImageMBS) 460
- 4.42.41 Constructor(Path as string) 461
- 4.42.42 Constructor(pic as picture) 461
- 4.42.43 Constructor(size as GMGeometryMBS, ColorValue as GMColorMBS) 462

4.42.45 contrast(sharpen as UInt32)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Contrast image (enhance intensity differences in image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.contrast(10)

Backdrop=image.CopyPicture
```

4.42.46 convolve(order as Integer, ColorMatrix() as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Convolve image.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)

dim m(8) as Double
```

```
m(0) = 0.25
m(1) = 0
m(2) = 0.25

m(3) = 0
m(4) = 0
m(5) = 0

m(6) = 0.25
m(7) = 0
m(8) = 0.25

g.convolve 3, m

Backdrop = g.CopyPicture
```

Notes: Applies a user-specified convolution to the image.
order represents the number of columns and rows in the filter kernel.
kernel is an array of doubles representing the convolution kernel.

4.42.47 CopyPicture as picture

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image and returns it as a new picture.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle
draw.Rectangle(250, 250, 100, 100)

Backdrop=image.CopyPicture
```

Notes: You may need to set image type to RGB to get it working.

See also:

- 4.42.48 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture 466

4.42.48 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image and returns it as a new picture.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.threshold 127

// convert to RGB so CopyPicture works
image.type = image.TrueColorType
Backdrop=image.CopyPicture(0,0,250,250)
```

Notes: You may need to set image type to RGB to get it working.
See also:

- 4.42.47 CopyPicture as picture 465

4.42.49 CopyPictureMask as picture

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image's mask and returns it as a new picture.
Example:

```
Dim f As FolderItem = SpecialFolder.Desktop.Child("test.png")
Dim g As New GMImageMBS(f)

// get image with mask
Dim p As picture = g.CopyPicture
p.mask = g.CopyPictureMask

window1.Backdrop = p
```

See also:

- 4.42.50 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture 467

4.42.50 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the image's mask and returns it as a new picture.
See also:

- 4.42.49 CopyPictureMask as picture 466

4.42.51 CopyPixelsMemory as Memoryblock

Plugin Version: 15.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copy the pixels as they are into a memoryblock.
Notes: Optional specify rectangle.
Returns nil on low memory or bad parameter. Image must be of type class direct (not palette picture).
Order of pixel data is normally Red, Green, Blue, Opacity. Or Cyan, Magenta, Yellow, Black for CMYK images.
For GMImageMBS, the data is 8bit per channel. For GMImage16MBS, the data is 16bit per channel.
See also:

- 4.42.52 CopyPixelsMemory(x as Integer, y as Integer, width as Integer, height as Integer) as Memoryblock 467

4.42.52 CopyPixelsMemory(x as Integer, y as Integer, width as Integer, height as Integer) as Memoryblock

Plugin Version: 15.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copy the pixels as they are into a memoryblock.
Notes: Optional specify rectangle.
Returns nil on low memory or bad parameter. Image must be of type class direct (not palette picture).
Order of pixel data is normally Red, Green, Blue, Opacity. Or Cyan, Magenta, Yellow, Black for CMYK images.
For GMImageMBS, the data is 8bit per channel. For GMImage16MBS, the data is 16bit per channel.
See also:

- 4.42.51 CopyPixelsMemory as Memoryblock 467

4.42.53 CreateHBITMAP as Ptr

Plugin Version: 15.1, Platform: Windows, Targets: All.

Function: Creates a HBITMAP for the image for use with Windows Declares.
Example:

```
// get test image
dim logo as Picture = LogoMBS(500)

// create GraphicsMagick image
dim g as new GMImageMBS(logo)

// make a HBitmap
dim hBitmap as ptr = g.CreateHBITMAP

// convert back to Xojo picture
dim pic as Picture = WindowsBitmapMBS.HBitmapToPicture(hBitmap, true)

// show in window
Backdrop = pic

// and cleanup memory
WindowsBitmapMBS.DeleteBitmap(hBitmap)
```

Notes: The HBITMAP returned needs to be freed when you are done with it or you risk having a memory leak.

4.42.54 crop(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crop image (return subregion of original image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.crop GMGeometryMBS.Make(100,200)

Backdrop=image.CopyPicture
```

4.42.55 cycleColormap(amount as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Cycle (rotate) image colormap.

Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.cycleColormap(5)

image.type = image.TrueColorType

Backdrop=image.CopyPicture
```

4.42.56 Describe(verbose as Integer = 1) as String

Plugin Version: 23.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Describes an image by printing its attributes.
Example:

```
Dim f As FolderItem = FindFile("test.jpg")

Dim g As New GMImageMBS(f)
Dim s As String = g.Describe(2)
Break
```

Notes: Attributes include the image width, height, size, and others.

verbose: Whether output should be verbose.
Default is 1. Pass 0 to get a shorted output.
Pass 2 to count the number of colors in the image.

4.42.57 despeckle

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Despeckle image (reduce speckle noise).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.despeckle

Backdrop=image.CopyPicture
```

4.42.58 display

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Display image on screen.
Notes: Caution: if an image format is is not compatible with the display visual (e.g. JPEG on a colormapped display) then the original image will be altered. Use a copy of the original if this is a problem.

The plugin is not compiled with X11 so this call may not be useful.

4.42.59 edge(radius as Double=0.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Edge image (hilight edges in image).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.edge

Backdrop=image.CopyPicture
```

Notes: The radius is the radius of the pixel neighborhood. Specify a radius of zero for automatic radius selection.

4.42.60 emboss(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Emboss image (hilight edges with 3D effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.emboss

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.42.61 enhance

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhance image (minimize noise).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.enhance

Backdrop=image.CopyPicture
```

4.42.62 erase

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set all image pixels to the current background color.

4.42.63 extent(geo as GMGeometryMBS)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)

// extend image to fit
dim geo as new GMGeometryMBS(500,500)
image.extent geo

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.42.64 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS) 472
- 4.42.65 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS, gravity as Integer) 473
- 4.42.66 extent(geo as GMGeometryMBS, gravity as Integer) 473

4.42.64 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)

// extend image to fit
dim geo as new GMGeometryMBS(500,500)
dim col as GMColorMBS = GMColorMBS.Black
image.extent geo, col

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.42.63 extent(geo as GMGeometryMBS) 471
- 4.42.65 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS, gravity as Integer) 473
- 4.42.66 extent(geo as GMGeometryMBS, gravity as Integer) 473

4.42.65 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS, gravity as Integer)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)

// extend image to fit
dim geo as new GMGeometryMBS(500,500)
dim col as GMColorMBS = GMColorMBS.Black
image.extent geo, col

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.42.63 extent(geo as GMGeometryMBS) 471
- 4.42.64 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS) 472
- 4.42.66 extent(geo as GMGeometryMBS, gravity as Integer) 473

4.42.66 extent(geo as GMGeometryMBS, gravity as Integer)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image canvas using background color sized according to geometry and composite existing image on it, with image placement controlled by gravity.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)

// resize proportionally to fit
dim geo as new GMGeometryMBS(500,500)
image.extent geo, image.CenterGravity

window1.Title = image.formatExpression("%wx%h")
```

```
window1.Backdrop = image.CopyPicture
```

Notes: Parameters are obtained from existing image properties if they are not specified via a method parameter. Parameters which are supported by image properties (gravity and backgroundColor) update those image properties as a side-effect.
See also:

- 4.42.63 extent(geo as GMGeometryMBS) 471
- 4.42.64 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS) 472
- 4.42.65 extent(geo as GMGeometryMBS, backgroundColor as GMColorMBS, gravity as Integer) 473

4.42.67 flip

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flip image (reflect each scanline in the vertical direction).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.flip

Backdrop=image.CopyPicture
```

4.42.68 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.42.69 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS) 475
- 4.42.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS) 475
- 4.42.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS) 475

4.42.69 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match:
See also:

- 4.42.68 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS) 474
- 4.42.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS) 475
- 4.42.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS) 475

4.42.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.42.68 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS) 474
- 4.42.69 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS) 475
- 4.42.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS) 475

4.42.71 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill color across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match:
See also:

- 4.42.68 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS) 474
- 4.42.69 floodFillColor(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS) 475
- 4.42.70 floodFillColor(x as UInt32, y as UInt32, fillColor as GMColorMBS) 475

4.42.72 floodFillOpacity(x as UInt32, y as UInt32, opacity as UInt32, PaintMethod as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill pixels matching color (within fuzz factor) of target pixel(x,y) with replacement opacity value using method.

4.42.73 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.42.74 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS) 476
- 4.42.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS) 476
- 4.42.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS) 477

4.42.74 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.42.73 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS) 476
- 4.42.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS) 476
- 4.42.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS) 477

4.42.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels that match the color of the target pixel and are neighbors of the target pixel.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.42.73 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS) 476
- 4.42.74 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS) 476
- 4.42.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS) 477

4.42.76 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS, borderColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flood-fill texture across pixels starting at target-pixel and stopping at pixels matching specified border color.
Notes: Uses current fuzz setting when determining color match.
See also:

- 4.42.73 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS) 476
- 4.42.74 floodFillTexture(point as GMGeometryMBS, fillColor as GMColorMBS, borderColor as GMColorMBS) 476
- 4.42.75 floodFillTexture(x as UInt32, y as UInt32, fillColor as GMColorMBS) 476

4.42.77 flop

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flop image (reflect each scanline in the horizontal direction).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.flop

Backdrop=image.CopyPicture
```

4.42.78 FontMap as string

Plugin Version: 20.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries current font map in use.
Notes: The MBS Plugin provides to GraphicsMagick the font map to use.
This is a XML defining which fonts are available.
Use this function to learn what fonts may be available or debug to see why a font doesn't load.

4.42.79 fontTypeMetrics(name as string) as GMTypeMetricMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain font metrics for text string given current font, pointsize, and density settings.

4.42.80 formatExpression(expression as string) as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Format the specified expression similar to command line '-format'.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: For example "%wx%h" is converted to a string containing image WIDTHxHEIGHT like "640x480".

4.42.81 frame

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a decorative frame around the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.frame
```

```
Backdrop=image.CopyPicture
```

See also:

- 4.42.82 frame(geometry as GMGeometryMBS) 479
- 4.42.83 frame(width as UInt32, height as UInt32, innerBevel as Integer=6, outerBevel as Integer=6) 479

4.42.82 frame(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a decorative frame around the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.frame(GMGeometryMBS.Make("10x10"))

Backdrop=image.CopyPicture
```

See also:

- 4.42.81 frame 478
- 4.42.83 frame(width as UInt32, height as UInt32, innerBevel as Integer=6, outerBevel as Integer=6) 479

4.42.83 frame(width as UInt32, height as UInt32, innerBevel as Integer=6, outerBevel as Integer=6)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Draw a decorative frame around the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.frame(15,15)

Backdrop=image.CopyPicture
```

See also:

- 4.42.81 frame 478
- 4.42.82 frame(geometry as GMGeometryMBS) 479

4.42.84 frameGeometryDefault as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The default geometry description for frame.

4.42.85 gamma(gammaRed as Double, gammaGreen as Double, gammaBlue as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma correct the image or individual image channels.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.gamma(1,2,3)

Backdrop=image.CopyPicture
```

See also:

- 4.42.252 gamma as Double 536

4.42.86 gaussianBlur(width as Double, sigma as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gaussian blur image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)
```

```
image.gaussianBlur(30, 10)

Backdrop=image.CopyPicture
```

Notes: The number of neighbor pixels to be included in the convolution mask is specified by width. The standard deviation of the gaussian bell curve is specified by sigma

4.42.87 gaussianBlurChannel(channel as Integer, width as Double, sigma as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gaussian blur image channel.
Notes: The number of neighbor pixels to be included in the convolution mask is specified by width. The standard deviation of the gaussian bell curve is specified by sigma.

4.42.88 getChromaBluePrimary(byref x as Double, byref y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity blue primary point.

4.42.89 getchromaGreenPrimary(byref x as Double, byref y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity green primary point.
Notes: e.g. x=0.3, y=0.6

4.42.90 getchromaRedPrimary(byref x as Double, byref y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity red primary point
Notes: e.g. x=0.64, y=0.33

4.42.91 getchromaWhitePoint(byref x as Double, byref y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity white point
Notes: e.g. x=0.3127, y=0.329

4.42.92 getConstPixels(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers read-only pixels from the image to the pixel cache as defined by the specified region

4.42.93 GetEXIFOrientation(byref orientation as integer) as boolean

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries orientation from EXIF.
Notes: Orientation is set to number from 0 to 8 depending on rotation. -1 if unknown.
This function can only read orientation, if there is an EXIF block in image.
Returns true for success and false for failure.

For new development, please use ExifTagsMBS class instead.

4.42.94 getPixels(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers pixels from the image to the pixel cache as defined by the specified region.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)

// get pointer to some pixels to write
dim x as ptr = g.getPixels(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
```

```
x.UInt32(o * 4) = &hFFFF0000
next

// write back
g.syncPixels

// show
me.Backdrop = g.CopyPicture
```

Notes: Modified pixels may be subsequently transferred back to the image via syncPixels. This method is valid for DirectClass images.

4.42.95 Graphics as GMGraphicsMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a graphics object for this image.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS(”white”) // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS(”red”) // Outline color
image.fillColor = new GMColorRGBMBS(”green”) // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: Using the graphics object you can draw on the image.

4.42.96 haldClut(image as GMImageMBS)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply a color lookup table (Hald CLUT) to the image.

4.42.97 Hash(Size as Integer = 8) as String

Plugin Version: 19.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Calculates a hash of the image.
Example:

```
Dim p As Picture = LogoMBS(500)
Dim g As New GMImageMBS(p)
msgbox g.Hash
```

Notes: Hash is returned as 64 characters being 1 or 0.
We convert image to 8x8, turn grayscale and check if pixels are above or below mean value.
This hash is quite immune against resizing, compression artifacts and hue changes.
You can use LevenshteinDistanceMBS or JaroWinklerDistanceMBS to compare two hashes.

Added size parameter for version 22.4:
The size of the bitmap. Value from 8 to 1024. Default is 8.

4.42.98 implode(factor as Double=0.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Implode image (special effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.implode(0.3)

Backdrop=image.CopyPicture
```

4.42.99 IsLoggingEnabled as Boolean

Plugin Version: 21.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Checks whether logging is enabled.
Notes: Returns true if we log GraphicsMagick usage.

4.42.100 JasperLibVersion as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries version string for jasper library.

4.42.101 label(text as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Assign a label to an image.
Notes: Use this option to assign a specific label to the image. Optionally you can include the image filename, type, width, height, or scene number in the label by embedding special format characters. If the first character of string is @, the image label is read from a file titled by the remaining characters in the string. When converting to Postscript, use this option to specify a header string to print above the image.
See also:

- 4.42.263 label as string 539

4.42.102 level(black_point as Double, white_point as Double, mid_point as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Level image to increase image contrast, and/or adjust image gamma.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.level(0, 127.0)

Backdrop=image.CopyPicture
```

Notes: Adjust the levels of the image by scaling the colors falling between specified white and black points to the full available quantum range. The parameters provided represent the black, mid (gamma), and white points. The black point specifies the darkest color in the image. Colors darker than the black point are set to zero. Mid point (gamma) specifies a gamma correction to apply to the image. White point specifies the lightest color in the image. Colors brighter than the white point are set to the maximum quantum value. The black and white point have the valid range 0 to MaxRGB while mid (gamma) has a useful range of 0 to ten:

4.42.103 levelChannel(channel as Integer, black_point as Double, white_point as Double, mid_point as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Level image channel to increase image contrast, and/or adjust image gamma.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.levelChannel(image.BlueChannel, 0, 127.0)

Backdrop=image.CopyPicture
```

Notes: Adjust the levels of the image channel by scaling the colors falling between specified white and black points to the full available quantum range. The parameters provided represent the black, mid (gamma), and white points. The black point specifies the darkest color in the image. Colors darker than the black point are set to zero. Mid point (gamma) specifies a gamma correction to apply to the image. White point specifies the lightest color in the image. Colors brighter than the white point are set to the maximum quantum value. The black and white point have the valid range 0 to MaxRGB while mid (gamma) has a useful range of 0 to ten.

4.42.104 LibVersion as String

Plugin Version: 14.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the version string of the GraphicsMagick library.

4.42.105 LoadIconvLibrary(path as String, byref Error as String) as boolean

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the iconv library.
Notes: The GraphicsMagick classes may use libiconv for text encoding conversion.
If you explicitly need, you can load the library on start of solution.

MBS Plugin may try to load iconv.dll/dylib/so automatically when first iconv function is called.

4.42.106 MagickVersion as string

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries the version text of the GraphicsMagick library.

4.42.107 magnify

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Magnify image by integral size (double the dimensions)
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.magnify

Backdrop=image.CopyPicture
```

4.42.108 map(mapImage as GMImageMBS, dither as boolean=false)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Remap image colors with closest color from a reference image.
Example:

```
// some picture we want to map colors
dim pic as Picture = LogoMBS(500)

// build a picture with palette
dim backgroundColor as new GMColorMBS(255,255,255) // white
dim size as new GMGeometryMBS(10,10)

dim i as new GMImageMBS(pic)
dim x as new GMImageMBS(size, backgroundColor)

x.pixelColor(0,0) = new GMColorMBS(0 ,0 ,0 ) // black
x.pixelColor(0,1) = new GMColorMBS(255,0 ,0 ) // red
x.pixelColor(0,2) = new GMColorMBS(0 ,255,0 ) // green
x.pixelColor(0,3) = new GMColorMBS(0 ,0 ,255) // blue
x.pixelColor(0,4) = new GMColorMBS(255,255,0 ) // yellow
x.pixelColor(0,5) = new GMColorMBS(0 ,255,255) // cyan
x.pixelColor(0,6) = new GMColorMBS(255,0 ,255) // magenta
```

```
// do the map
i.map(x, false)

// convert result from palette picture to bitmap picture
i.type = i.TrueColorType

// and copy picture to backdrop
Backdrop = i.CopyPicture
```

Notes: Set dither to true in to apply Floyd/Steinberg error diffusion to the image. By default, color reduction chooses an optimal set of colors that best represent the original image. Alternatively, you can choose a particular set of colors from an image file with this option.

4.42.109 matteFloodfill(target as GMColorMBS, opacity as UInt32, x as Integer, y as Integer, PaintMethod as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Floodfill designated area with a replacement opacity value.

4.42.110 medianFilter(radius as Double=0.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Filter image by replacing each pixel component with the median color in a circular neighborhood.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.medianFilter(10)

Backdrop=image.CopyPicture
```

4.42.111 minify

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduce image by integral (half) size.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.minify

Backdrop=image.CopyPicture
```

4.42.112 modequalizeifyImage

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Not documented.

4.42.113 modifyImage

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Prepare to update image (copy if reference >1).
Notes: Normally Magick++'s implicit reference counting takes care of all instance management. In the rare case that the automatic instance management does not work, use this method to assure that there is only one reference to the image to be modified. It should be used in the cases where a GraphicsMagick C function is used directly on an image which may have multiple references:

4.42.114 modulate(brightness as Double, saturation as Double, hue as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Modulate percent hue, saturation, and brightness of an image.
Example:

```
dim logo as Picture = LogoMBS(500)
dim image as new GMImageMBS(logo)

image.type = image.TrueColorType

// brightness 150%
image.modulate(150,100,100)
backdrop = image.CopyPicture
```

Notes: Modulation of saturation and brightness is as a ratio of the current value (100 for no change). Modulation of hue is an absolute rotation of -180 degrees to +180 degrees from the current position corresponding to an argument range of 0 to 200 (100 for no change).

4.42.115 montageGeometry as GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile size and offset within an image montage.
Notes: Only valid for montage images.

4.42.116 motionBlur(radius as Double, sigma as Double, angle as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Motion blur image with specified blur factor.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.motionBlur(30,10,90)

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels. The angle parameter specifies the angle the object appears to be comming from (zero degrees is from the right).

4.42.117 negate(grayscale as boolean=false)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negate colors in image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.negate

Backdrop=image.CopyPicture
```

Notes: Set grayscale to only negate grayscale values in image.

4.42.118 normalize

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Normalize image (increase contrast by normalizing the pixel values to span the full range of color values).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.normalize

Backdrop=image.CopyPicture
```

4.42.119 oilPaint(radius as Double=3.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Oilpaint image (image looks like an oil painting).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.oilPaint

Backdrop=image.CopyPicture
```

4.42.120 opacity(opacity as UInt32)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or attenuate the opacity channel in the image.
Notes: If the image pixels are opaque then they are set to the specified opacity value, otherwise they are blended with the supplied opacity value. The value of opacity ranges from 0 (completely opaque) to MaxRGB. The defines OpaqueOpacity and TransparentOpacity are available to specify completely opaque or completely transparent, respectively.

4.42.121 opaque(opaqueColor as GMColorMBS, penColor as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Change color of specified opaque pixel to specified pen color.

4.42.122 ping(data as GMBlobMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads information for an image from the blob.
Notes: Ping is similar to read except only enough of the image is read to determine the image columns, rows, and filesize. Access the columns, rows, and fileSize attributes after invoking ping. The image pixels are not valid after calling ping.
See also:

- 4.42.123 ping(file as folderitem) 492
- 4.42.124 ping(Path as string) 493

4.42.123 ping(file as folderitem)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads information for an image from the file.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.jpg”)

// try with Constructor (same as read)
dim t1 as Double = Microseconds
dim g1 as new GMImageMBS(f)

// now just ping
dim t2 as Double = Microseconds
dim g2 as new GMImageMBS
g2.ping(f)

// or read
dim t3 as Double = Microseconds
dim g3 as new GMImageMBS
g3.read(f)

dim t4 as Double = Microseconds

// show speeds
MsgBox str(T4-t3)+” ¬µs for read”+EndOfLine+_
```

```
str(T3-t2)+" ¬µs for ping"+EndOfLine+_
str(T2-t1)+" ¬µs for Constructor"
```

Notes: Ping is similar to read except only enough of the image is read to determine the image columns, rows, and filesize. Access the columns, rows, and fileSize attributes after invoking ping. The image pixels are not valid after calling ping.
See also:

- 4.42.122 ping(data as GMBlobMBS) 492
- 4.42.124 ping(Path as string) 493

4.42.124 ping(Path as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads information for an image from the image specification.
Notes: Ping is similar to read except only enough of the image is read to determine the image columns, rows, and filesize. Access the columns, rows, and fileSize attributes after invoking ping. The image pixels are not valid after calling ping.
See also:

- 4.42.122 ping(data as GMBlobMBS) 492
- 4.42.123 ping(file as folderitem) 492

4.42.125 PNGLibVersion as string

Plugin Version: 17.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries PNG library version string.

4.42.126 quantize(measureError as boolean=false)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Quantize image (reduce number of colors).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.quantize
image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: Set measureError to true in order to calculate error attributes.

4.42.127 QuantumDepth as Integer

Plugin Version: 14.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the quantum depth.

4.42.128 quantumOperator(channel as Integer, Operator as Integer, rvalue as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply an arithmetic or bitwise operator to the image pixel quantums.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)

const AddQuantumOp = 1
const ThresholdQuantumOp = 10

g.quantumOperator( g.AllChannels, AddQuantumOp, 100)

// show
me.Backdrop = g.CopyPicture
```

See also:

- 4.42.129 quantumOperator(x as Integer, y as Integer, columns as Integer, rows as Integer, channel as Integer, Operator as Integer, rvalue as Double) 494

4.42.129 quantumOperator(x as Integer, y as Integer, columns as Integer, rows as Integer, channel as Integer, Operator as Integer, rvalue as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply an arithmetic or bitwise operator to the image pixel quantums.
See also:

- 4.42.128 quantumOperator(channel as Integer, Operator as Integer, rvalue as Double) 494

4.42.130 raiseGeometryDefault as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The default geometry description for raise.

4.42.131 raiseImage

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Raise image (lighten or darken the edges of an image to give a 3-D raised or lowered effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.raiseImage

Backdrop=image.CopyPicture
```

See also:

- 4.42.132 raiseImage(geometry as GMGeometryMBS, raisedFlag as boolean=false) 495

4.42.132 raiseImage(geometry as GMGeometryMBS, raisedFlag as boolean=false)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Raise image (lighten or darken the edges of an image to give a 3-D raised or lowered effect).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.raiseImage(GMGeometryMBS.Make(5,8))

Backdrop=image.CopyPicture
```

See also:

- 4.42.131 raiseImage 495

4.42.133 randomThreshold(thresholds as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Random threshold image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.randomThreshold(GMGeometryMBS.make("50x200"))

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: Changes the value of individual pixels based on the intensity of each pixel compared to a random threshold. The result is a low-contrast, two color image. The thresholds argument is a geometry containing LOWxHIGH thresholds. If the string contains 2x2, 3x3, or 4x4, then an ordered dither of order 2, 3, or 4 will be performed instead. If a channel argument is specified then only the specified channel is altered. This is a very fast alternative to 'quantize' based dithering.

4.42.134 randomThresholdChannel(thresholds as GMGeometryMBS, channel as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Random threshold image channel.
Notes: Changes the value of individual pixels based on the intensity of each pixel compared to a random threshold. The result is a low-contrast, two color image. The thresholds argument is a geometry containing LOWxHIGH thresholds. If the string contains 2x2, 3x3, or 4x4, then an ordered dither of order 2, 3, or 4 will be performed instead. If a channel argument is specified then only the specified channel is altered. This is a very fast alternative to 'quantize' based dithering.

4.42.135 read(blob as GMBlobMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame from in-memory Blob.
Example:

```
// get some image data (e.g. from blob in database)
dim logo as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(logo, 80)

// new image
Dim mp as new GMImageMBS
dim blob as new GMBlobMBS(jpegData)

// read data from blob into this image object
mp.Read blob

// sometimes you need to explicit convert to RGB/RGBA
'mp.type = mp.TrueColorMatteType
Backdrop=mp.CombinePictureWithMask
```

See also:

- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497
- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size from in-memory Blob.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498

- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size and depth from in-memory Blob.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size, depth, and format from in-memory Blob.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497

- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size, and format from in-memory Blob.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497
- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.140 read(file as folderitem)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame into current object.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497

- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.141 read(path as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame into current object.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497
- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.142 read(size as GMGeometryMBS, file as folderitem)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size into current object.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497

- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.143 read(size as GMGeometryMBS, Path as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame of specified size into current object.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497
- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr) 501

4.42.144 read(width as UInt32, height as UInt32, map as string, StorageType as Integer, data as ptr)

Plugin Version: 14.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Read single image frame from an array of raw pixels, with specified storage type (ConstituteImage).
Notes: Returns an Image corresponding to an image stored in a raw memory array format. The pixel data must be in scanline order top-to-bottom. The data can be unsigned char, unsigned short int, unsigned int, unsigned long, float, or double. Float and double require the pixels to be normalized to the range [0..1] ,

otherwise the range is [0..MaxVal] where MaxVal is the maximum possible value for that type.

Note that for most 32-bit architectures the size of an unsigned long is the same as unsigned int, but for 64-bit architectures observing the LP64 standard, an unsigned long is 64 bits, while an unsigned int remains 32 bits. This should be considered when deciding if the data should be described as ”Integer” or ”Long”.

For example, to create a 640x480 image from unsigned red-green-blue character data, use

image = new GMImageMBS(640, 480, ”RGB”, GMImageMBS.StorageTypeCharPixel, pixels);

width: width in pixels of the image.
height: height in pixels of the image.
map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (same as Transparency), O = Opacity, T = Transparency, C = cyan, Y = yellow, M = magenta, K = black, or I = intensity (for grayscale). Specify ”P” = pad, to skip over a quantum which is intentionally ignored. Creation of an alpha channel for CMYK images is currently not supported.
type: Define the data type of the pixels. Float and double types are expected to be normalized [0..1] otherwise [0..MaxRGB] . Choose from these types: StorageTypeCharPixel, StorageTypeShortPixel, StorageTypeIntegerPixel, StorageTypeLongPixel, StorageTypeFloatPixel, or StorageTypeDoublePixel.
pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.
See also:

- 4.42.135 read(blob as GMBlobMBS) 496
- 4.42.136 read(blob as GMBlobMBS, size as GMGeometryMBS) 497
- 4.42.137 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer) 498
- 4.42.138 read(blob as GMBlobMBS, size as GMGeometryMBS, depth as Integer, magick as string) 498
- 4.42.139 read(blob as GMBlobMBS, size as GMGeometryMBS, magick as string) 499
- 4.42.140 read(file as folderitem) 499
- 4.42.141 read(path as string) 500
- 4.42.142 read(size as GMGeometryMBS, file as folderitem) 500
- 4.42.143 read(size as GMGeometryMBS, Path as string) 501

4.42.145 reduceNoise

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduce noise in image using a noise peak elimination filter.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.reduceNoise

Backdrop=image.CopyPicture
```

See also:

- 4.42.146 reduceNoise(order as Double) 503

4.42.146 reduceNoise(order as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduce noise in image using a noise peak elimination filter.
See also:

- 4.42.145 reduceNoise 502

4.42.147 ReleaseDate as String

Plugin Version: 14.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the release date of the used graphics magick library.
Notes: We update the library only when someone needs an update, so if you need, please contact us.

4.42.148 repage

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image page canvas and position.

4.42.149 resize(geo as GMGeometryMBS)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image, specifying only geometry, with filter and blur obtained from Image default.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)
```

```
// resize proportionally to fit
dim geo as new GMGeometryMBS(500,500)
image.resize geo

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

Notes: Same result as 'zoom' method.
See also:

- 4.42.150 resize(geo as GMGeometryMBS, filterType as Integer) 504
- 4.42.151 resize(geo as GMGeometryMBS, filterType as Integer, blur as double) 504

4.42.150 resize(geo as GMGeometryMBS, filterType as Integer)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image, specifying geometry and filter, with blur using Image default.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child("test.jpg")
dim image as new GMImageMBS(f)

// resize proportionally to fit
dim geo as new GMGeometryMBS(500,500)
image.resize geo, image.CubicFilter

window1.Title = image.formatExpression("%wx%h")
window1.Backdrop = image.CopyPicture
```

See also:

- 4.42.149 resize(geo as GMGeometryMBS) 503
- 4.42.151 resize(geo as GMGeometryMBS, filterType as Integer, blur as double) 504

4.42.151 resize(geo as GMGeometryMBS, filterType as Integer, blur as double)

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image, specifying geometry, filter, and blur.
Example:

```
dim f as folderitem = SpecialFolder.Desktop.Child(”test.jpg”)
dim image as new GMImageMBS(f)

// resize proportionally to fit
dim geo as new GMGeometryMBS(500,500)
image.resize geo, image.CubicFilter, 3

window1.Title = image.formatExpression(”%wx%h”)
window1.Backdrop = image.CopyPicture
```

See also:

- 4.42.149 resize(geo as GMGeometryMBS) 503
- 4.42.150 resize(geo as GMGeometryMBS, filterType as Integer) 504

4.42.152 roll(columns as UInt32, rows as UInt32)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll image (rolls image vertically and horizontally) by specified number of columnms and rows).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.roll(30,30)

Backdrop=image.CopyPicture
```

See also:

- 4.42.153 roll(roll as GMGeometryMBS) 505

4.42.153 roll(roll as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll image (rolls image vertically and horizontally) by specified number of columnms and rows).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)
```

```
image.roll(GMGeometryMBS.Make(0,0,30,30))

Backdrop=image.CopyPicture
```

See also:

- 4.42.152 roll(columns as UInt32, rows as UInt32) 505

4.42.154 rotate(degree as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotate image counter-clockwise by specified number of degrees.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.rotate(30)

Backdrop=image.CopyPicture
```

4.42.155 sample(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image by using pixel sampling algorithm.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.sample GMGeometryMBS.make(100,100)

Backdrop=image.CopyPicture
```

4.42.156 scale(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image by using simple ratio algorithm which provides good quality.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.scale new GMGeometryMBS(100,100)

Backdrop=image.CopyPicture
```

4.42.157 segment(clusterThreshold as Double=1.0, smoothingThreshold as Double=1.5)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Segment (coalesce similar image components) by analyzing the histograms of the color components and identifying units that are homogeneous with the fuzzy c-means technique.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.segment

image.type = image.TrueColorType

Backdrop=image.CopyPicture
```

Notes: A histogram is built for the image. This histogram is filtered to reduce noise and a second derivative of the histogram plot is built and used to identify potential cluster colors (peaks in the histogram). The cluster colors are then validated by scanning through all of the pixels to see how many pixels fall within each cluster. Some candidate cluster colors may not match any of the image pixels at all and should be discarded. Specify clusterThreshold, as the number of pixels matching a cluster color in order for the cluster to be considered valid. SmoothingThreshold eliminates noise in the second derivative of the histogram. As the value is increased, you can expect a smoother second derivative. The default is 1.5.

4.42.158 setChromaBluePrimary(x as Double, y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity blue primary point.
Notes: e.g. x=0.15, y=0.06

4.42.159 setchromaGreenPrimary(x as Double, y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity green primary point.
Notes: e.g. x=0.3, y=0.6

4.42.160 setchromaRedPrimary(x as Double, y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity red primary point
Notes: e.g. x=0.64, y=0.33

4.42.161 setchromaWhitePoint(x as Double, y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chromaticity white point
Notes: e.g. x=0.3127, y=0.329

4.42.162 SetEXIFOrientation(orientation as integer) as boolean

Plugin Version: 18.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets orientation for EXIF.
Notes: Changing orientation may need to set orientation via SetOrientation and SetEXIFOrientation. With a JPEG you have orientation both in JPEG header and in EXIF metadata.
Returns true for success and false for failure.

For new development, please use ExifTagsMBS class instead.

4.42.163 SetLogEventMask(events as String)

Plugin Version: 21.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set log event mask.
Example:

```
GMImageMBS.SetLogEventMask(”coder,annotate”)
```

Notes: Defines which events are logged.
By default logging goes to stderr, so on macOS you may need to run your app via Terminal to see logs there.

List of events includes: none, Configure, Annotate, Render, Transform, Locale, Coder, X11, Cache, Blob, Deprecate, User, Resource, TemporaryFile, Exception, Option, Information, Warning, Error, FatalError and All.

4.42.164 SetPicture(pic as picture, x as Integer, y as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the picture into the Image at the given position.

4.42.165 SetPictureMask(maskpic as picture, x as Integer, y as Integer)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the picture into the Image's mask at the given position.
Example:

```
// this converts 32 bit PNG with alpha channel to BMP

dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim p as Picture = Picture.Open(f)

dim g as new GMImageMBS( new GMGeometryMBS(p.Width, p.Height), new GMColorGrayMBS(1.0))

g.type = g.TrueColorMatteType
g.matte = True
g.magick = "BMP"

g.SetPicture(p, 0, 0)
g.SetPictureMask(p.mask.invertMBS, 0, 0)

f = SpecialFolder.Desktop.Child("test.bmp")
g.write(f)
```

4.42.166 setPixels(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Allocates a pixel cache region to store image pixels as defined by the region rectangle.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)

// get pointer to some pixels to write
dim x as ptr = g.setPixels(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
g.syncPixels

// show
me.Backdrop = g.CopyPicture
```

Notes: This area is subsequently transferred from the pixel cache to the image via syncPixels.

4.42.167 setStrokeDashArray(values() as Double)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets stroke dash pattern.
Notes: Specify the pattern of dashes and gaps used to stroke paths. The strokeDashArray represents a zero-terminated array of numbers that specify the lengths of alternating dashes and gaps in pixels. If an odd number of values is provided, then the list of values is repeated to yield an even number of values. A typical strokeDashArray array might contain the members 5 3 2 0, where the zero value indicates the end of the pattern array.

4.42.168 shade(azimuth as Double=30.0, elevation as Double=30.0, colorShading as boolean=false)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shade image using distant light source.
Notes: Specify azimuth and elevation as the position of the light source. By default, the shading results as a grayscale image.. Set colorShading to true to shade the red, green, and blue components of the image.

4.42.169 sharpen(radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpen pixels in image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.sharpen

Backdrop=image.CopyPicture
```

Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.42.170 sharpenChannel(channel as Integer, radius as Double=0.0, sigma as Double=1.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpen pixels in image channel.
Notes: The radius parameter specifies the radius of the Gaussian, in pixels, not counting the center pixel. The sigma parameter specifies the standard deviation of the Laplacian, in pixels.

4.42.171 shave(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shave pixels from image edges.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.shave(new GMGeometryMBS(200,200))

Backdrop=image.CopyPicture
```

4.42.172 shear(xShearAngle as Double, yShearAngle as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shear image (create parallelogram by sliding image by X or Y axis).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.shear(10,20)

Backdrop=image.CopyPicture
```

Notes: Shearing slides one edge of an image along the X or Y axis, creating a parallelogram. An X direction shear slides an edge along the X axis, while a Y direction shear slides an edge along the Y axis. The amount of the shear is controlled by a shear angle. For X direction shears, x degrees is measured relative to the Y axis, and similarly, for Y direction shears y degrees is measured relative to the X axis. Empty triangles left over from shearing the image are filled with the color defined as borderColor.

4.42.173 signature(force as boolean=false) as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image textual signature.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

MsgBox image.signature

Backdrop=image.CopyPicture
```

Notes: Set force to true in order to re-calculate the signature regardless of whether the image data has been modified.

4.42.174 solarize(factor as Double=50.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Solarize image (similar to effect seen when exposing a photographic film to light during the development process)
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.solarize

Backdrop=image.CopyPicture
```

4.42.175 spread(amount as UInt32=3)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Spread pixels randomly within image by specified ammount
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.spread 5

Backdrop=image.CopyPicture
```

4.42.176 statistics as GMImageStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain image statistics.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim stat as GMImageStatisticsMBS = g.statistics
dim gs as GMImageChannelStatisticsMBS = stat.blue

MsgBox "blue channel: "+str(gs.minimum)+"-"+str(Gs.maximum)+", mean "+str(gs.mean)
```

Notes: Statistics are normalized to the range of 0.0 to 1.0 and are output to the specified ImageStatistics structure.

4.42.177 stegano(watermark as GMImageMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add a digital watermark to the image (based on second image).
Example:

```
dim p as Picture = LogoMBS(500)
dim p1 as Picture = New Picture(550,500,32)
dim p2 as Picture = New Picture(550,500,32)

p1.Graphics.DrawPicture p, 0,0
p2.Graphics.DrawPicture p,50,0

dim image1 as new GMImageMBS(p1)
dim image2 as new GMImageMBS(p2)

image2.zoom(new GMGeometryMBS(100,100)) // scale down

// add watermark
image1.stegano(image2)

// now make a threshold so you see the difference
image1.threshold 254

image1.type = image1.TrueColorType
Backdrop=image1.CopyPicture
```

4.42.178 stereo(rightImage as GMImageMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Create an image which appears in stereo when viewed with red-blue glasses (Red image on left, blue on right)
Example:

```
dim p as Picture = LogoMBS(500)
dim p1 as Picture = New Picture(550,500,32)
dim p2 as Picture = New Picture(550,500,32)

p1.Graphics.DrawPicture p, 0,0
p2.Graphics.DrawPicture p,50,0

dim image1 as new GMImageMBS(p1)
dim image2 as new GMImageMBS(p2)

image1.stereo(IMAGE2)

Backdrop=image1.CopyPicture
```

4.42.179 strip

Plugin Version: 13.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: Remove all profiles and text attributes from the image.

4.42.180 strokeDashArray as Double()

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries stroke dash pattern.
Notes: Specify the pattern of dashes and gaps used to stroke paths. The strokeDashArray represents a zero-terminated array of numbers that specify the lengths of alternating dashes and gaps in pixels. If an odd number of values is provided, then the list of values is repeated to yield an even number of values. A typical strokeDashArray array might contain the members 5 3 2 0, where the zero value indicates the end of the pattern array.

4.42.181 swirl(degree as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Swirl image (image pixels are rotated by degrees).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.swirl 200
```

```
Backdrop=image.CopyPicture
```

4.42.182 syncPixels

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers the image cache pixels to the image.

4.42.183 texture(texture as GMImageMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Channel a texture on pixels matching image background color.

4.42.184 threshold(degree as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Threshold image channels (below threshold becomes black, above threshold becomes white).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.threshold 127

// convert to RGB so CopyPicture works
image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: The range of the threshold parameter is 0 to MaxRGB.

4.42.185 thumbnail(geometry as GMGeometryMBS)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image using several algorithms to make smaller images very quickly.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)

// make thumbnail
dim geo as new GMGeometryMBS(100, 100)
g.thumbnail(geo)

// show
me.Backdrop = g.CopyPicture
```

4.42.186 TIFFLibVersion as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries version string for tiff library.

4.42.187 transform(imageGeometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform image based on image and crop geometries.
Notes: Crop geometry is optional.
See also:

- 4.42.188 transform(imageGeometry as GMGeometryMBS, cropGeometry as GMGeometryMBS) 517

4.42.188 transform(imageGeometry as GMGeometryMBS, cropGeometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform image based on image and crop geometries.
Notes: Crop geometry is optional.
See also:

- 4.42.187 transform(imageGeometry as GMGeometryMBS) 517

4.42.189 transformOrigin(tx as Double, ty as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Origin of coordinate system to use when annotating with text or drawing.

4.42.190 transformReset

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reset transformation parameters to default.

4.42.191 transformRotation(angle as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotation to use when annotating with text or drawing.

4.42.192 transformScale(tx as Double, ty as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scale to use when annotating with text or drawing.

4.42.193 transformSkewX(x as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Skew to use in X axis when annotating with text or drawing.

4.42.194 transformSkewY(y as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Skew to use in Y axis when annotating with text or drawing.

4.42.195 transparent(color as GMColorMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Add matte channel to image, setting pixels matching color to transparent.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

dim c as new GMColorMBS(”white”)
image.transparent(c)

Backdrop=image.CombinePictureWithMask
```

4.42.196 trim

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Trim edges that are the background color from the image.
Example:

```
dim p as Picture = LogoMBS(500)
// make the logo picture bigger
dim q as Picture = New Picture(700,700,32)

q.Graphics.DrawPicture p,100,100

dim image as new GMImageMBS(q)

// now trim the white border away
image.trim

Backdrop=image.CopyPicture
```

Notes: See ColorFuzz property for how far the pixel value can differentiate.

4.42.197 unregisterId

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Not documented.

4.42.198 unsharpmask(radius as Double, sigma as Double, amount as Double, threshold as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replace image with a sharpened version of the original image using the unsharp mask algorithm.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.unsharpmask(10,1,0.5,50)

Backdrop=image.CopyPicture
```

Notes: radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
amount: the percentage of the difference between the original and the blur image that is added back into the original.
threshold: the threshold in pixels needed to apply the diffence amount.

4.42.199 unsharpmaskChannel(channel as Integer, radius as Double, sigma as Double, amount as Double, threshold as Double)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replace image channel with a sharpened version of the original image using the unsharp mask algorithm.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.unsharpmaskChannel(Image.RedChannel, 10,1,0.5,50)

Backdrop=image.CopyPicture
```

Notes:

4.42.200 wave(amplitude as Double=25.0, wavelength as Double=150.0)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

channel: image channel to modify.
radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
amount: the percentage of the difference between the original and the blur image that is added back into the original.
threshold: the threshold in pixels needed to apply the diffence amount.

Function: Map image pixels to a sine wave.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.wave

Backdrop=image.CopyPicture
```

4.42.201 WebPVersion as String

Plugin Version: 23.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns version of WebP library.
Notes: Should be a version string like "1.3.0".

4.42.202 write(blob as GMBlobMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to in-memory Blob, with optional format and adjoin parameters.
See also:

- 4.42.203 write(blob as GMBlobMBS, magick as string) 522
- 4.42.204 write(blob as GMBlobMBS, magick as string, depth as UInt32) 522
- 4.42.205 write(file as folderitem) 522
- 4.42.206 write(Path as string) 523
- 4.42.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 523

4.42.203 write(blob as GMBlobMBS, magick as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to in-memory Blob, with optional format and adjoin parameters.
See also:

- 4.42.202 write(blob as GMBlobMBS) 521
- 4.42.204 write(blob as GMBlobMBS, magick as string, depth as UInt32) 522
- 4.42.205 write(file as folderitem) 522
- 4.42.206 write(Path as string) 523
- 4.42.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 523

4.42.204 write(blob as GMBlobMBS, magick as string, depth as UInt32)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to in-memory Blob, with optional format and adjoin parameters.
See also:

- 4.42.202 write(blob as GMBlobMBS) 521
- 4.42.203 write(blob as GMBlobMBS, magick as string) 522
- 4.42.205 write(file as folderitem) 522
- 4.42.206 write(Path as string) 523
- 4.42.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 523

4.42.205 write(file as folderitem)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to a file.
Example:

```
// this converts 32 bit PNG with alpha channel to BMP

dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim p as Picture = Picture.Open(f)

dim g as new GMImageMBS( new GMGeometryMBS(p.Width, p.Height), new GMColorGrayMBS(1.0))
```

```
g.type = g.TrueColorMatteType
g.matte = True
g.magick = "BMP"

g.SetPicture(p, 0, 0)
g.SetPictureMask(p.mask.invertMBS, 0, 0)

f = SpecialFolder.Desktop.Child("test.bmp")
g.write(f)
```

See also:

- 4.42.202 write(blob as GMBlobMBS) 521
- 4.42.203 write(blob as GMBlobMBS, magick as string) 522
- 4.42.204 write(blob as GMBlobMBS, magick as string, depth as UInt32) 522
- 4.42.206 write(Path as string) 523
- 4.42.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 523

4.42.206 write(Path as string)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to a file.
See also:

- 4.42.202 write(blob as GMBlobMBS) 521
- 4.42.203 write(blob as GMBlobMBS, magick as string) 522
- 4.42.204 write(blob as GMBlobMBS, magick as string, depth as UInt32) 522
- 4.42.205 write(file as folderitem) 522
- 4.42.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr) 523

4.42.207 write(x as Integer, y as Integer, columns as Integer, rows as Integer, map as string, type as Integer, Pixels as Ptr)

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Write single image frame to an array of pixels with storage type specified by user (DispatchImage).
Notes: e.g. image.write(0, 0, 640, 1, "RGB", 0, pixels)
See also:

- 4.42.202 write(blob as GMBlobMBS) 521
- 4.42.203 write(blob as GMBlobMBS, magick as string) 522
- 4.42.204 write(blob as GMBlobMBS, magick as string, depth as UInt32) 522
- 4.42.205 write(file as folderitem) 522
- 4.42.206 write(Path as string) 523

4.42.208 ZLibVersion as string

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries version string for zlib library.

4.42.209 zoom(geometry as GMGeometryMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Zoom (resize) image to specified size.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.zoom(new GMGeometryMBS(200,200))

Backdrop=image.CopyPicture
```

4.42.210 ZPL(Header as boolean = true) as String

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries image as ZPL hex image.
Notes: This is for sending data to a receipt printer. The picture is taken as black & white image and we build the hex string, you can send to the printer.
Picture needs to have a width dividable by 8.

Set Header to false for skipping header and footer.

4.42.211 Properties

4.42.212 adjoin as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Join images into a single multi-image file.
Notes: (Read and Write property)

4.42.213 animationDelay as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Time in 1/100ths of a second (0 to 65535) which must expire before displaying the next image in an animated sequence.
Notes: This option is useful for regulating the animation of a sequence of GIF images within Netscape.
(Read and Write property)

4.42.214 animationIterations as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of iterations to loop an animation (e.g. Netscape loop extension) for.
Notes: (Read and Write property)

4.42.215 antiAlias as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Control antialiasing of rendered Postscript and Postscript or TrueType fonts.
Notes: Enabled by default.
(Read and Write property)

4.42.216 backgroundColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Example:

```
// make a red and turn it to 100% transparent
Dim red As New GMColorRGBMBS("red")
```

```
red.alpha = 1

// now make image and make it RGBA with the transparnet background
dim RastoredVectorImage as New GMImageMBS
RastoredVectorImage.type = GMImageMBS.TrueColorMatteType
RastoredVectorImage.backgroundColor = red

// now read SVG, so we get a transparent background
dim SVG_File as FolderItem = SpecialFolders.desktop.Child(”test.svg”)
RastoredVectorImage.read(SVG_File)
```

Notes: (Read and Write property)

4.42.217 backgroundTexture as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image file name to use as the background texture.
Notes: Does not modify image pixels.
(Read and Write property)

4.42.218 baseColumns as UInt32

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base image width (before transformations)
Notes: (Read only property)

4.42.219 baseFilename as String

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base image filename (before transformations)
Notes: (Read only property)

4.42.220 baseRows as Uint32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base image height (before transformations).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

Title = str(image.baseRows)+" x "+str(image.baseColumns)
```

Notes: (Read only property)

4.42.221 borderColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

4.42.222 boundingBox as GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Return smallest bounding box enclosing non-border pixels.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.fillColor = new GMColorRGBMBS("red") // set color
image.strokeColor = new GMColorRGBMBS("green") // set color

image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle

draw.Circle(250, 250, 120, 150)
draw.Draw

draw = nil
image.type = image.TrueColorType

Backdrop = image.CopyPicture
```

MsgBox image.boundingBox.StringValue

Notes: The current fuzz value is used when discriminating between pixels. This is the crop bounding box used by crop(Geometry(0,0)).
(Read only property)

4.42.223 boxColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Base color that annotation text is rendered on (default none).
Notes: (Read and Write property)

4.42.224 classType as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image class (DirectClass or PseudoClass).
Notes: NOTE: setting a DirectClass image to PseudoClass will result in the loss of color information if the number of colors in the image is greater than the maximum palette size (either 256 or 65536 entries depending on the value of QuantumDepth when ImageMagick was built):
(Read and Write property)

4.42.225 clipMask as GMImageMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Associate a clip mask image with the current image.
Notes: The clip mask image must have the same dimensions as the current image or an exception is thrown. Clipping occurs wherever pixels are transparent in the clip mask image. Clipping Pass an invalid image to unset an existing clip mask.
(Read and Write property)

4.42.226 colorFuzz as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this

option to match colors that are close to the target color in RGB space.

e.g. set to 50 for 8 bit class and 50 * 257 for the 16 bit class to allow 20% divagation in pixel values.
(Read and Write property)

4.42.227 colorMapSize as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of entries in the colormap.
Notes: Setting the colormap size may extend or truncate the colormap. The maximum number of supported entries is specified by the MaxColormapSize constant, and is dependent on the value of QuantumDepth when GraphicsMagick is compiled. An exception is thrown if more entries are requested than may be supported. Care should be taken when truncating the colormap to ensure that the image colormap indexes reference valid colormap entries.
(Read and Write property)

4.42.228 colorSpace as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The colorspace (e.g. CMYK) used to represent the image pixel colors.
Notes:

UndefinedColorspace	= 0
RGBColorspace	= 1 (Plain old RGB colorspace)
GRAYColorspace	= 2 (Plain old full-range grayscale)
TransparentColorspace	= 3 (RGB but preserve matte channel during quantize)
OHTAColorspace	= 4
XYZColorspace	= 5 (CIE XYZ)
YCCColorspace	= 6 (Kodak PhotoCD PhotoYCC)
YIQColorspace	= 7
YPbPrColorspace	= 8
YUVColorspace	= 9
CMYKColorspace	= 10 (Cyan, magenta, yellow, black, alpha)
sRGBColorspace	= 11 (Kodak PhotoCD sRGB)
HSLColorspace	= 12 (Hue, saturation, luminosity)
HWBColorspace	= 13 (Hue, whiteness, blackness)
LABColorspace	= 14 (LAB colorspace not supported yet other than via lcms)
CineonLogRGBColorspace	= 15 (RGB data with Cineon Log scaling, 2.048 density range)
Rec601LumaColorspace	= 16 (Luma (Y) according to ITU-R 601)
Rec601YCbCrColorspace	= 17 (YCbCr according to ITU-R 601)
Rec709LumaColorspace	= 18 (Luma (Y) according to ITU-R 709)
Rec709YCbCrColorspace	= 19 (YCbCr according to ITU-R 709)

(Read and Write property)

4.42.229 columns as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image width.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

Title = str(image.columns)+" x "+str(image.rows)
Backdrop=image.CopyPicture
```

Notes: (Read only property)

4.42.230 comment as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image comment.
Notes: When you set this property, you add comment string to image.
By default, each image is commented with its file name. Use this method to assign a specific comment to the image. Optionally you can include the image filename, type, width, height, or other image attributes by embedding special format characters:
(Read and Write property)

4.42.231 compose as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composition operator to be used when composition is implicitly used (such as for image flattening).
Notes: (Read and Write property)

4.42.232 compressType as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.
Notes: The default is the compression type of the input image file.
(Read and Write property)

4.42.233 debug as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enable printing of debug messages from GraphicsMagick as it executes.
Notes: (Read and Write property)

4.42.234 density as GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Vertical and horizontal resolution in pixels of the image.
Example:

```
dim p as new GMImageMBS

dim item as FolderItem = SpecialFolder.Desktop.Child(”input.png”)
p.read(item)
p.scale new GMGeometryMBS(3750,3750)
p.quality = 95
p.resolutionUnits = p.PixelsPerInchResolution
p.density = new GMGeometryMBS(300, 300)
dim out as FolderItem = SpecialFolder.Desktop.Child(”output.png”)
p.write out
```

Notes: This option specifies an image density when decoding a Postscript or Portable Document page. Often used with psPageSize.
(Read and Write property)

4.42.235 depth as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (bits allocated to red/green/blue components).
Notes: Used to specify the bit depth when reading or writing raw images or when the output format supports multiple depths. Defaults to the quantum depth that GraphicsMagick is compiled with.
(Read and Write property)

4.42.236 directory as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile names from within an image montage.
Notes: (Read only property)

4.42.237 endian as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian mode.
Notes: Endianness (LSBEndian like Intel, MSBEndian like SPARC, or NativeEndian for what this computer uses) for image formats which support endian-specific options.
(Read and Write property)

4.42.238 ExifThumbnail as String

Plugin Version: 20.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extracts an embedded thumbnail in EXIF data.
Example:

```
Dim g As New GMImageMBS

// not load, but just read header & metadata
g.ping("/Users/cs/Desktop/test.JPG")

// get thumbnail
Dim Thumbnail As String = g.ExifThumbnail

// show it
window1.Backdrop = picture.FromData(Thumbnail)
```

Notes: Returns string containing JPEG compressed image data.

For new development, please use ExifTagsMBS class instead.
(Read only property)

4.42.239 fileName as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image file name.
Notes: (Read and Write property)

4.42.240 fileSize as Int64

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of bytes of the image on disk.
Notes: (Read only property)

4.42.241 fillColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color to use when filling drawn objects.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.42.242 fillPattern as GMImageMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pattern to use while filling drawn objects.
Notes: (Read and Write property)

4.42.243 fillRule as Integer

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rule to use when filling drawn objects
Notes: (Read and Write property)

4.42.244 filterType as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The reduction filter employed has a sigificant effect on the time required to resize an image and the resulting quality. The default filter is Lanczos which has been shown to produce high quality results when reducing most images.
Notes: Filter to use when resizing image.
(Read and Write property)

4.42.245 font as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font.
Notes: If the font is a fully qualified X server font name, the font is obtained from an X server. To use a TrueType font, precede the TrueType filename with an @. Otherwise, specify a Postscript font name (e.g. "helvetica").
(Read and Write property)

4.42.246 FontFamily as String

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font family to use.
Notes: The plugin will look through the list to find best match for combination of family, style, stretch and weight.
You can use either FontFamily or Font property, but not both.
Setting font family clears font.
(Read and Write property)

4.42.247 fontPointsize as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font point size.
Notes: (Read and Write property)

4.42.248 FontStretch as Integer

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font stretch.
Notes: Can be Normal, UltraCondensed, ExtraCondensed, Condensed, SemiCondensed, SemiExpanded, Expanded, ExtraExpanded, UltraExpanded or Any.
See stretch constants.
(Read and Write property)

4.42.249 FontStyle as Integer

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font style to use.
Notes: Can be Normal, Italic, Oblique or Any.
See font style constants.
(Read and Write property)

4.42.250 FontWeight as Integer

Plugin Version: 20.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font weight.
Notes: The font weight in range from 0 to 1000.
400 is normal and 800 bold.
(Read and Write property)

4.42.251 format as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Long image format description.
Notes: (Read only property)

4.42.252 gamma as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma correct the image or individual image channels.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.gamma = 3.0

Backdrop=image.CopyPicture
```

Notes: If you get the value, it is the gamma level of the image. Gamma is a pow() function which converts between the linear light representation and the representation for the computer display. Most computer images are gamma corrected to 2.2 (1/0.4545) so that each step results in a visually linear step on a computer or video display:
(Read and Write property)
See also:

- 4.42.85 gamma(gammaRed as Double, gammaGreen as Double, gammaBlue as Double) 480

4.42.253 geometry as GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size of the image when encoding.
Notes: (Read only property)

4.42.254 getConstIndexes as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain immutable image pixel indexes (valid for PseudoClass images)
Notes: (Read only property)

4.42.255 getIndexes as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtain mutable image pixel indexes (valid for PseudoClass images)
Notes: (Read only property)

4.42.256 gifDisposeMethod as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: GIF disposal method.
Notes: This option (specific to the GIF file format) is used to control how successive frames are rendered (how the preceding frame is disposed of) when creating a GIF animation.

Constant	Disposal	Description
UndefinedDispose	0	No disposal specified.
NoneDispose	1	Do not dispose between frames.
BackgroundDispose	2	Overwrite frame with background color from header.
PreviousDispose	3	Overwrite with previous frame.

(Read and Write property)

4.42.257 handle as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Example:

```
dim c as new GMColorMBS("white")
dim g as new GMGeometryMBS(100,100)
dim image as new GMImageMBS(g, c)
MsgBox hex(Image.handle) // valid if not zero
```

Notes: (Read and Write property)

4.42.258 height as Integer

Plugin Version: 10.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the image.
Example:

```
dim c as new GMColorRGBMBS(1.0,0.0,0.0)
dim size as new GMGeometryMBS(100,100)
dim g as new GMImageMBS(size, c)
```

```
MsgBox str(g.width)+" "+str(g.height)
```

Notes: This is a convenience function for you which calls size.height.
(Read only property)

4.42.259 iccColorProfile as GMBlobMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: ICC color profile.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("IMG_0793.tif")
dim Image as new GMImageMBS(f)
dim ProfileBlob as GMBlobMBS = Image.iccColorProfile
dim ProfileData as string = ProfileBlob.CopyString
dim cm as LCMS2ProfileMBS = LCMS2ProfileMBS.OpenProfileFromString(ProfileData)
dim name as string = cm.Name

Break // check data in debugger
```

Notes: Supplied via a Blob since Magick++/ and GraphicsMagick do not currently support formating this data structure directly. Specifications are available from the International Color Consortium for the format of ICC color profiles.
(Read and Write property)

4.42.260 interlaceType as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.
(Read and Write property)

4.42.261 iptcProfile as GMBlobMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: IPTC profile.
Notes: Supplied via a Blob since Magick++ and GraphicsMagick do not currently support formating this data structure directly. Specifications are available from the International Press Telecommunications Council for IPTC profiles.
(Read and Write property)

4.42.262 isValid as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Does object contain valid image?
Notes: Set to false in order to invalidate the image. Images constructed via the default constructor are invalid images and isValid() will return false.
(Read and Write property)

4.42.263 label as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image label.
Notes: (Read only property)
See also:

- 4.42.101 label(text as string) 485

4.42.264 lineWidth as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Stroke width for drawing vector objects (default one)
Notes: This method is now deprecated. Please use strokeWidth instead.
(Read and Write property)

4.42.265 magick as string

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The name of the codec to use for compression.
Example:

```
// this converts 32 bit PNG with alpha channel to BMP

dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim p as Picture = Picture.Open(f)

dim g as new GMImageMBS( new GMGeometryMBS(p.Width, p.Height), new GMColorGrayMBS(1.0))

g.type = g.TrueColorMatteType
g.matte = True
g.magick = "BMP"

g.SetPicture(p, 0, 0)
g.SetPictureMask(p.mask.invertMBS, 0, 0)

f = SpecialFolder.Desktop.Child("test.bmp")
g.write(f)
```

Notes: (Read and Write property)

4.42.266 matte as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image supports transparency (matte channel)
Notes: (Read and Write property)

4.42.267 matteColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (frame) color.
Notes: (Read and Write property)

4.42.268 meanErrorPerPixel as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mean error per pixel computed when an image is color reduced.
Notes: This parameter is only valid if verbose is set to true and the image has just been quantized.
(Read only property)

4.42.269 modulusDepth as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image modulus depth (minimum number of bits required to support red/green/blue components without loss of accuracy).
Notes: The pixel modulus depth may be decreased by supplying a value which is less than the current value, updating the pixels (reducing accuracy) to the new depth. The pixel modulus depth can not be increased over the current value using this method.
(Read and Write property)

4.42.270 monochrome as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform image to black and white while color reducing (quantizing).
Notes: (Read and Write property)

4.42.271 normalizedMaxError as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The normalized max error per pixel computed when an image is color reduced.
Notes: This parameter is only valid if verbose is set to true and the image has just been quantized.
(Read only property)

4.42.272 normalizedMeanError as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The normalized mean error per pixel computed when an image is color reduced.
Notes: This parameter is only valid if verbose is set to true and the image has just been quantized.
(Read only property)

4.42.273 orientation as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image orientation. Supported by some file formats such as DPX and TIFF. Useful for turning the right way up.
Notes: (Read and Write property)

4.42.274 page as GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size and location of an image canvas.
Notes: Use this option to specify the dimensions and position of the Postscript page in dots per inch or a TEXT page in pixels. This option is typically used in concert with density .

Page may also be used to position a GIF image (such as for a scene in an animation).
(Read and Write property)

4.42.275 penColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The pen color.
Notes: (Read and Write property)

4.42.276 quality as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level (default 75).
Notes: (Read and Write property)

4.42.277 quantizeColors as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Maximum number of colors to quantize to.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.quantizeColors = 10
image.quantize

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.42.278 quantizeColorSpace as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorspace to quantize in (default RGB).
Example:

```
// load a picture
dim f as FolderItem = SpecialFolder.Desktop.Child("test.png")
dim pic as Picture = Picture.Open(f)

const GrayColorSpace = 2

Dim Converter As New GMImageMBS(Pic)

// quantize with dither
Converter.type = GMImageMBS.BilevelType
Converter.quantizeColorSpace = GrayColorSpace
Converter.quantizeColors = 2
Converter.quantizeDither = True
Converter.quantize

// convert back to Xojo
Converter.type = GMImageMBS.TrueColorType
Backdrop = Converter.CopyPicture
```

Notes: Empirical evidence suggests that distances in color spaces such as YUV or YIQ correspond to perceptual color differences more closely than do distances in RGB space. These color spaces may give better results when color reducing an image.
(Read and Write property)

4.42.279 quantizeDither as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Apply Floyd/Steinberg error diffusion to the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.quantizeColors = 10
image.quantizeDither = true
image.quantize

image.type = image.TrueColorType
Backdrop=image.CopyPicture
```

Notes: The basic strategy of dithering is to trade intensity resolution for spatial resolution by averaging the intensities of several neighboring pixels. Images which suffer from severe contouring when reducing colors can be improved with this option. The quantizeColors or monochrome option must be set for this option to take effect.
(Read and Write property)

4.42.280 quantizeTreeDepth as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Depth of the quantization color classification tree.
Notes: Values of 0 or 1 allow selection of the optimal tree depth for the color reduction algorithm. Values between 2 and 8 may be used to manually adjust the tree depth.
(Read and Write property)

4.42.281 Quiet as Boolean

Plugin Version: 19.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Suppress all warning messages.
Notes: Error messages are still reported.
(Read and Write property)

4.42.282 renderingIntent as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of rendering intent (used when applying an ICC color profile).
Notes: (Read and Write property)

4.42.283 resolutionUnits as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Example:

```
Dim item As FolderItem = SpecialFolder.Desktop.Child("test.jpeg")
Dim p As New GMImageMBS(item)

// scale image
p.quality = 95
p.scale New GMGeometryMBS(1000,1000)

// change resolution
p.density = New GMGeometryMBS(300, 300)
p.resolutionUnits = p.PixelsPerInchResolution

// remove metadata
Dim empty As New GMBlobMBS
p.profile("EXIF") = empty
p.profile("IPTC") = empty
p.profile("XMP") = empty

Dim out As FolderItem = SpecialFolder.Desktop.Child("output.jpeg")
p.write out
```

Notes: (Read and Write property)

4.42.284 rows as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of pixel rows in the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

Title = str(image.columns)+" x "+str(image.rows)
Backdrop=image.CopyPicture
```

Notes: (Read only property)

4.42.285 scene as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image scene number.
Notes: (Read and Write property)

4.42.286 size as GMGeometryMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height of a raw image (an image which does not support width and height information).
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

MsgBox image.size.StringValue

image.size = new GMGeometryMBS(200,200)

Backdrop=image.CopyPicture
```

Notes: Size may also be used to affect the image size read from a multi-resolution format (e.g. Photo CD, JBIG, or JPEG.
(Read and Write property)

4.42.287 strokeAntiAlias as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enable/disable stroke anti-aliasing.
Notes: (Read and Write property)

4.42.288 strokeColor as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color to use when drawing object outlines.
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)
```

```
image.strokeColor = new GMColorRGBMBS(”red”) // Outline color
image.fillColor = new GMColorRGBMBS(”green”) // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.42.289 strokeDashOffset as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: While drawing using a dash pattern, specify distance into the dash pattern to start the dash (default 0).
Notes: (Read and Write property)

4.42.290 strokeLineCap as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the end of open subpaths when they are stroked. #
Notes: Values of LineCap are UndefinedCap, ButtCap, RoundCap, and SquareCap.
(Read and Write property)

4.42.291 strokeLineJoin as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify the shape to be used at the corners of paths (or other vector shapes) when they are stroked. Values of LineJoin are UndefinedJoin, MiterJoin, RoundJoin, and BevelJoin.
Notes: (Read and Write property)

4.42.292 strokeMiterLimit as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specify miter limit.
Notes: When two line segments meet at a sharp angle and miter joins have been specified for 'lineJoin', it is possible for the miter to extend far beyond the thickness of the line stroking the path. The miterLimit' imposes a limit on the ratio of the miter length to the 'lineWidth'. The default value of this parameter is 4. (Read and Write property)

4.42.293 strokePattern as GMImageMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pattern image to use while stroking object outlines.
Notes: (Read and Write property)

4.42.294 strokeWidth as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Stroke width for drawing vector objects (default one).
Example:

```
dim g as new GMGeometryMBS(500,500)
dim c as new GMColorRGBMBS("white") // white
dim image as new GMImageMBS(g, c)

image.strokeColor = new GMColorRGBMBS("red") // Outline color
image.fillColor = new GMColorRGBMBS("green") // Fill color
image.strokeWidth = 5

dim draw as GMGraphicsMBS = image.Graphics

// Draw a circle
draw.Circle(250, 250, 120, 150)

Backdrop=image.CopyPicture
```

Notes: (Read and Write property)

4.42.295 subImage as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Subimage of an image sequence.
Notes: (Read and Write property)

4.42.296 subRange as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of images relative to the base image.
Notes: (Read and Write property)

4.42.297 textEncoding as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Annotation text encoding (e.g. "UTF-16").
Notes: (Read and Write property)

4.42.298 tileName as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile name.
Notes: (Read and Write property)

4.42.299 totalColors as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of colors in the image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

Title = str(image.totalColors) // shows 5284
Backdrop=image.CombinePictureWithMask
```

Notes: (Read only property)

4.42.300 type as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of this image.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)

image.type = image.GrayscaleType

Backdrop=image.CopyPicture
```

Notes: You can set this value to convert the image to the type.

Convert the image representation to the specified type or retrieve the current image type. If the image is reduced to an inferior type, then image information may be lost (e.g. color changed to grayscale).

Available enumerations for the type parameter:

BilevelType	1	black/white
GrayscaleType	2	grayscale
GrayscaleMatteType	3	grayscale with alpha (opacity) channel
PaletteType	4	colormapped
PaletteMatteType	5	colormapped with transparency
TrueColorType	6	true (full) color
TrueColorMatteType	7	true (full) color with alpha (opacity) channel
ColorSeparationType	8	Cyan, magenta, yellow, and black
ColorSeparationMatteType	9	Cyan, magenta, yellow, and black with alpha (opacity) channel
OptimizeType	10	Optimize the image type to best represent the existing pixels

(Read and Write property)

4.42.301 verbose as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Print detailed information about the image.

Notes: (Read and Write property)

4.42.302 view as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: FlashPix viewing parameters.
Notes: (Read and Write property)

4.42.303 width as Integer

Plugin Version: 10.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the image.
Example:

```
dim c as new GMColorRGBMBS(1.0,0.0,0.0)
dim size as new GMGeometryMBS(100,100)
dim g as new GMImageMBS(size, c)

MsgBox str(g.width)+" "+str(g.height)
```

Notes: This is a convenience function for you which calls size.width.
(Read only property)

4.42.304 x11Display as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: X11 display to display to, obtain fonts from, or to capture image from.
Notes: (Read and Write property)

4.42.305 XResolution as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: x resolution of the image.
Notes: See also density functions.
Settable with version 19.0.
(Read and Write property)

4.42.306 YResolution as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: y resolution of the image.
Notes: Settable with version 19.0.
(Read and Write property)

4.42.307 attributeValue(name as string) as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Access an arbitrary named image attribute.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("IMG_4048.jpg")
dim g as new GMImageMBS(f)
dim a as string = g.attributeValue("EXIF:DateTime")
MsgBox a
```

Notes: Any number of named attributes may be attached to the image. For example, the image comment is a named image attribute with the name "comment". EXIF tags are attached to the image as named attributes. Use the syntax "EXIF:<tag>" to request an EXIF tag similar to "EXIF:DateTime":
(Read and Write computed property)

4.42.308 channelDepth(channel as Integer) as UInt32

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or obtain modulus channel depth.
Notes: (Read and Write computed property)

4.42.309 colorMap(index as UInt32) as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color at colormap position index.
Notes: (Read and Write computed property)
See also:

- 4.42.26 colorMap as GMColorMBS() 453

4.42.310 defineSet(magick as string, key as string) as boolean

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or obtain a definition flag to applied when encoding or decoding the specified format.
Notes: Similar to the defineValue() method except that passing the flag value 'true' creates a value-less define with that format and key. Passing the flag value 'false' removes any existing matching definition. The method returns 'true' if a matching key exists, and 'false' if no matching key exists.
(Read and Write computed property)

4.42.311 defineValue(magick as string, key as string) as string

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set or obtain a definition string to applied when encoding or decoding the specified format.
Notes: The meanings of the definitions are format specific. The format is designated by the magick argument, the format-specific key is designated by key, and the associated value is specified by value. See the defineSet() method if the key must be removed entirely.
(Read and Write computed property)

4.42.312 pixelColor(x as UInt32, y as UInt32) as GMColorMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Get/set pixel color at location x & y.
Example:

```
dim p as Picture = LogoMBS(500)
dim image as new GMImageMBS(p)
dim c as new GMColorMBS("red")

for x as Integer = 240 to 260
image.pixelColor(x,250)=c
next

for y as Integer = 240 to 260
image.pixelColor(250,y)=c
next

Backdrop=image.CopyPicture
```

Notes: (Read and Write computed property)

4.42.313 profile(name as string) as GMBlobMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Get or set a named profile.
Example:

```
Dim item As FolderItem = SpecialFolder.Desktop.Child("test.jpeg")
Dim p As New GMImageMBS(item)

// scale image
p.quality = 95
p.scale New GMGeometryMBS(1000,1000)

// change resolution
p.density = New GMGeometryMBS(300, 300)
p.resolutionUnits = p.PixelsPerInchResolution

// remove metadata
Dim empty As New GMBlobMBS
p.profile("EXIF") = empty
p.profile("IPTC") = empty
p.profile("XMP") = empty

Dim out As FolderItem = SpecialFolder.Desktop.Child("output.jpeg")
p.write out
```

Notes: Add or remove a named profile to/from the image. Remove the profile by passing an empty Blob (e.g. Blob()). Valid names are "*", "8BIM", "ICM", "IPTC", or a user/format-defined profile name.

Retrieve a named profile from the image. Valid names are: "8BIM", "8BIMTEXT", "APP1", "APP1JPEG", "ICC", "ICM", & "IPTC" or an existing user/format-defined profile name
(Read and Write computed property)

4.42.314 Constants

Constants

Constant	Value	Description
AbsoluteIntent	3	One of the intent type constants.
AddCompositeOp	8	One of the composite type constants.
AllChannels	10	One of the possible channel constants.
AllCompliance	&hffff	One of the Compliance type constants.
AssociatedAlpha	1	One of the possible alpha type constants.
AtopCompositeOp	4	One of the composite type constants.
BackgroundDispose	2	One of the gif dispose type constants.
BesselFilter	14	One of the filter type constants.
BilevelType	1	One of the image type constants.
BlackChannel	8	One of the possible channel constants.
BlackmanFilter	7	One of the filter type constants.
BlueChannel	5	One of the possible channel constants.
BottomLeftOrientation	4	One of the orientation type constants. Line direction: Left to right Frame Direction: Bottom to top
BottomRightOrientation	3	One of the orientation type constants. Line direction: Right to left Frame Direction: Bottom to top
BoxFilter	2	One of the filter type constants.
BumpmapCompositeOp	12	One of the composite type constants.
BZipCompression	2	One of the compression type constants.
CatromFilter	11	One of the filter type constants.
CenterGravity	5	One of the possible gravity constants.
ClearCompositeOp	18	One of the composite type constants.
ColorizeCompositeOp	28	One of the composite type constants.
ColorSeparationMatteType	9	One of the image type constants.
ColorSeparationType	8	One of the image type constants.
ConcatenateMode	3	One of the image type constants.
CopyBlackCompositeOp	35	One of the composite type constants.
CopyBlueCompositeOp	16	One of the composite type constants.
CopyCompositeOp	13	One of the composite type constants.
CopyCyanCompositeOp	32	One of the composite type constants.
CopyGreenCompositeOp	15	One of the composite type constants.
CopyMagentaCompositeOp	33	One of the composite type constants.
CopyOpacityCompositeOp	17	One of the composite type constants.
CopyRedCompositeOp	14	One of the composite type constants.
CopyYellowCompositeOp	34	One of the composite type constants.
CubicFilter	10	One of the filter type constants.
CyanChannel	2	One of the possible channel constants.
DarkenCompositeOp	24	One of the composite type constants.
DifferenceCompositeOp	10	One of the composite type constants.
DirectClass	1	One of the class type constants.
DisplaceCompositeOp	20	One of the composite type constants.
DissolveCompositeOp	19	One of the composite type constants.
DivideCompositeOp	36	One of the composite type constants.
EastGravity	6	One of the possible gravity constants.
FaxCompression	3	One of the compression type constants.
ForgetGravity	0	One of the possible gravity constants.
FrameMode	1	One of the mode type constants.
GaussianFilter	8	One of the filter type constants.
GaussianNoise	1	One of the possible noise constants.
GrayChannel	11	One of the possible channel constants.
GrayscaleMatteType	3	One of the image type constants.
GrayscaleType	2	One of the image type constants.
GreenChannel	3	One of the possible channel constants.
Group4Compression	4	One of the compression type constants.
HammingFilter	6	One of the filter type constants.
HanningFilter	5	One of the filter type constants.

Font Stretch

Constant	Value	Description
AnyStretch	9	Don't care.
CondensedStretch	3	Condensed
ExpandedStretch	6	Expanded
ExtraCondensedStretch	2	Extra Condensed
ExtraExpandedStretch	7	Extra Expanded
NormalStretch	0	Normal (Default)
SemiCondensedStretch	4	Semi Condensed
SemiExpandedStretch	5	Semi Expanded
UltraCondensedStretch	1	Ultra Condensed
UltraExpandedStretch	8	Ultra Expanded

Font Style

Constant	Value	Description
AnyStyle	3	Don't care.
ItalicStyle	1	Italic font.
NormalStyle	0	Normal (Default)
ObliqueStyle	2	Oblique font.

Storage Types

Constant	Value	Description
StorageTypeCharPixel	0	8bit numbers.
StorageTypeDoublePixel	5	64bit floating numbers.
StorageTypeFloatPixel	4	32bit floating numbers.
StorageTypeIntegerPixel	2	32bit numbers.
StorageTypeLongPixel	3	64bit numbers.
StorageTypeShortPixel	1	16bit numbers.

4.43 class GMImageStatisticsMBS

4.43.1 class GMImageStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for image statistics.
Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

4.43.2 Methods

4.43.3 Constructor

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The private constructor.

4.43.4 Properties

4.43.5 blue as GMImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue channel statistics.
Notes: (Read only property)

4.43.6 green as GMImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green channel statistics.
Notes: (Read only property)

4.43.7 opacity as GMImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity channel statistics.
Notes: (Read only property)

4.43.8 red as GMImageChannelStatisticsMBS

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red channel statistics.
Notes: (Read only property)

4.44 class GMLockMBS

4.44.1 class GMLockMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for locking a certain resource.
Notes: The idea is to pass the constructor a mutexlock and keep the only reference to this new lock object on the stack. On the end of the method, the destructor is called by Xojo and releases the mutexlock automatically.

4.44.2 Methods

4.44.3 Constructor(mutexlock as GMMutexLockMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new Lock based on the given mutexlock.

4.44.4 Properties

4.44.5 handle as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.44.6 target as GMMutexLockMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mutexlock this lock is referencing to.
Notes: (Read and Write property)

4.45 class GMMontageFramedMBS

4.45.1 class GMMontageFramedMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: MontageFramed provides the means to specify montage options when it is desired to have decorative frames around the image thumbnails.
Notes: MontageFramed inherits from Montage and therefore provides all the methods of Montage as well as those shown in the table ”MontageFramed Methods”.

Framed thumbnails consist of four components: the thumbnail image, the thumbnail frame, the thumbnail border, an optional thumbnail shadow, and an optional thumbnail label area.
Subclass of the GMMontageMBS class.

4.45.2 Methods

4.45.3 Constructor

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.45.4 Properties

4.45.5 borderColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the background color within the thumbnail frame.
Notes: (Read and Write computed property)

4.45.6 borderWidth as Uint32

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the border (in pixels) to place between a thumbnail and its surrounding frame.
Notes: This option only takes effect if thumbnail frames are enabled (via frameGeometry) and the thumbnail geometry specification doesn’t also specify the thumbnail border width.
(Read and Write computed property)

4.45.7 frameGeometry as GMGeometryMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the geometry specification for frame to place around thumbnail.
Notes: If this parameter is not specified, then the montage is unframed.
(Read and Write computed property)

4.45.8 matteColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail frame color.
Notes: (Read and Write computed property)

4.46 class GMMontageMBS

4.46.1 class GMMontageMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Montage is the base class to provide montage options and provides methods to set all options required to render simple (unframed) montages.
Example:

```
// build montage
dim StackingMontage as New GM16MontageMBS
StackingMontage.backgroundColor = New GM16ColorMBS(&cE7E7E7)
StackingMontage.fillColor = New GM16ColorMBS(&c000000)
StackingMontage.tile = New GM16GeometryMBS(”1x20”)
StackingMontage.geometry = New GM16GeometryMBS(”160x120+5+5”)
StackingMontage.font = ”Helvetica”
StackingMontage.pointSize = 12
StackingMontage.title = ”Title goes here”

// make picture
dim logo as Picture = LogoMBS(500)
dim image as New GM16ImageMBS(logo)

image.label(”Sample label”)

// Put the current image into the array
Dim StackingFrames As new GM16ImageArrayMBS
StackingFrames.insert(image)

// show result
dim resultImages as GM16ImageArrayMBS = StackingFrames.montageImages(StackingMontage)
Backdrop = resultImages.Image(0).CopyPicture
```

Notes: See GMMontageFramedMBS if you would like to create a framed montage.

Unframed thumbnails consist of four components: the thumbnail image, the thumbnail border, an optional thumbnail shadow, and an optional thumbnail label area.

4.46.2 Methods

4.46.3 Constructor

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.46.4 Properties

4.46.5 handle as Integer

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.46.6 backgroundColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the background color that thumbnails are imaged upon.
Notes: (Read and Write computed property)

4.46.7 compose as Integer

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the image composition algorithm for thumbnails.
Notes: This controls the algorithm by which the thumbnail image is placed on the background. Use of OverCompositeOp is recommended for use with images that have transparency. This option may have negative side-effects for images without transparency.
(Read and Write computed property)

4.46.8 fileName as string

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the image filename to be used for the generated montage images.
Notes: To handle the case were multiple montage images are generated, a printf-style format may be embedded within the filename. For example, a filename specification of image%02d.miff names the montage

images as image00.miff, image01.miff, etc.
(Read and Write computed property)

4.46.9 fillColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the fill color to use for the label text.
Notes: (Read and Write computed property)

4.46.10 font as string

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail label font.
Notes: (Read and Write computed property)

4.46.11 geometry as GMGeometryMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the size of the generated thumbnail.
Notes: (Read and Write computed property)

4.46.12 gravity as Integer

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail positioning within the specified geometry area.
Notes: If the thumbnail is smaller in any dimension than the geometry, then it is placed according to this specification.
See Gravity constants in GMImageMBS class.
(Read and Write computed property)

4.46.13 label as string

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the format used for the image label.
Notes: Special format characters may be embedded in the format string to include information about the

image.
(Read and Write computed property)

4.46.14 penColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the pen color to use for the label text (same as fill).
Notes: (Read and Write computed property)

4.46.15 pointSize as UInt32

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the thumbnail label font size.
Notes: (Read and Write computed property)

4.46.16 shadow as boolean

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enable/disable drop-shadow on thumbnails.
Notes: (Read and Write computed property)

4.46.17 strokeColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the stroke color to use for the label text.
Notes: (Read and Write computed property)

4.46.18 texture as string

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies a texture image to use as montage background.
Notes: The built-in textures ”granite:” and ”plasma:” are available. A texture is the same as a background image.
(Read and Write computed property)

4.46.19 tile as GMGeometryMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the maximum number of montage columns and rows in the montage.
Notes: The montage is built by filling out all cells in a row before advancing to the next row. Once the montage has reached the maximum number of columns and rows, a new montage image is started.
(Read and Write computed property)

4.46.20 title as string

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies the montage title.
Notes: (Read and Write computed property)

4.46.21 transparentColor as GMColorMBS

Plugin Version: 13.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Specifies a montage color to set transparent.
Notes: This option can be set the same as the background color in order for the thumbnails to appear without a background when rendered on an HTML page. For best effect, ensure that the transparent color selected does not occur in the rendered thumbnail colors.
(Read and Write computed property)

4.47 class GMMutexLockMBS

4.47.1 class GMMutexLockMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mutex class for GraphicsMagick.

4.47.2 Methods

4.47.3 lock

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Locks the lock.
Notes: Only one thread at a time can get the lock. The other threads will wait when lock is called.

4.47.4 unlock

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unlocks the lock.

4.47.5 Properties

4.47.6 handle as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.48 class GMNotInitializedExceptionMBS

4.48.1 class GMNotInitializedExceptionMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception raised if you access a method/property in an object and the object was not initialized propertly.
Notes: Check the message property for details.
Subclass of the GMErrorExceptionMBS class.

4.49 class GMPathArgsMBS

4.49.1 class GMPathArgsMBS

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: This is a class for arguments to the path arc/curve methods in GMGraphicsMBS.
Example:

```
dim g as new GMPathArgsMBS(1,2,3,4) // for a QuadraticCurveto

MsgBox str(g.x1)+EndOfLine+str(g.y1)+EndOfLine+str(g.x)+EndOfLine+str(g.y)
```

Notes: Due we use this class for three different ways, we have three constructors to fill in the value you need for the calls.

4.49.2 Methods

4.49.3 Constructor

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor for creating an empty object.
See also:

- 4.49.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 570
- 4.49.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double) 571
- 4.49.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 571

4.49.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create the arguments object for the PathArc methods in GMGraphicsMBS.
See also:

- 4.49.3 Constructor 570

- 4.49.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double) 571
- 4.49.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 571

4.49.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create the arguments object for the QuadraticCurveto methods in GMGraphicsMBS.
Example:

```
dim g as new GMPathArgsMBS(1,2,3,4)

MsgBox str(g.x1)+EndOfLine+str(g.y1)+EndOfLine+str(g.x)+EndOfLine+str(g.y)
```

See also:

- 4.49.3 Constructor 570
- 4.49.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 570
- 4.49.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double) 571

4.49.6 Constructor(x1 as Double, y1 as Double, x2 as Double, y2 as Double, x as Double, y as Double)

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor to create the arguments object for the Curveto methods in GMGraphicsMBS.
Example:

```
dim g as new GMPathArgsMBS(1,2,3,4,5,6)

MsgBox str(g.x1)+EndOfLine+str(g.y1)+EndOfLine+str(g.x2)+EndOfLine+str(g.y2)+EndOfLine+str(g.x)+EndOfLine+str(g.y)
```

See also:

- 4.49.3 Constructor 570
- 4.49.4 Constructor(radiusX as Double, radiusY as Double, xAxisRotation as Double, largeArcFlag as boolean, sweepFlag as Boolean, x as Double, y as Double) 570

- 4.49.5 Constructor(x1 as Double, y1 as Double, x as Double, y as Double) 571

4.49.7 Properties

4.49.8 largeArcFlag as Boolean

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The large arc flag.
Notes: Draw longer of the two matching arcs
(Read and Write property)

4.49.9 radiusX as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The radius x value.
Notes: (Read and Write property)

4.49.10 radiusY as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The radius y value.
Notes: (Read and Write property)

4.49.11 sweepFlag as Boolean

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The sweep flag value.
Notes: Draw arc matching clock-wise rotation.
(Read and Write property)

4.49.12 x as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x value.
Notes: For an arc: End-point X

(Read and Write property)

4.49.13 x1 as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x1 value.
Notes: (Read and Write property)

4.49.14 x2 as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x2 value.
Notes: (Read and Write property)

4.49.15 xAxisRotation as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x Axis Rotation value.
Notes: Rotation relative to X axis.
(Read and Write property)

4.49.16 y as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y value.
Notes: for an arc: End-point Y
(Read and Write property)

4.49.17 y1 as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y1 value.
Notes: (Read and Write property)

4.49.18 y2 as Double

Plugin Version: 10.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y2 value.
Notes: (Read and Write property)

4.50 class GMPixelsMBS

4.50.1 class GMPixelsMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty pixels object.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim p as new GMPixelsMBS(g)

// get pointer to some pixels to read/write
dim x as ptr = p.get(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
p.sync

// show
window1.Backdrop = g.CopyPicture
```

4.50.2 Methods

4.50.3 Constructor(Image as GMImageMBS)

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new Pixels object with the pixels from an image.

4.50.4 get(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfer pixels from the image to the pixel view as defined by the specified region.

Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim p as new GMPixelsMBS(g)

// get pointer to some pixels
dim x as ptr = p.get(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
p.sync

// show
window1.Backdrop = g.CopyPicture
```

Notes: Modified pixels may be subsequently transferred back to the image via sync.

4.50.5 getConst(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfer read-only pixels from the image to the pixel view as defined by the specified region.

4.50.6 indexes as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Return pixel colormap index array.

4.50.7 set(x as Integer, y as Integer, columns as Integer, rows as Integer) as Ptr

Plugin Version: 16.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Allocate a pixel view region to store image pixels as defined by the region rectangle.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim p as new GMPixelsMBS(g)

// get pointer to some pixels to write
dim x as ptr = p.set(0, 0, 100, 100)

// draw a red line to the pixel buffer
dim o as Integer
for i as Integer = 0 to 99
o = 100 * i + i
x.UInt32(o * 4) = &hFFFF0000
next

// write back
p.sync

// show
window1.Backdrop = g.CopyPicture
```

Notes: This area is subsequently transferred from the pixel view to the image via sync.

4.50.8 sync

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transfers the image cache pixels to the image.

4.50.9 Properties

4.50.10 columns as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width of view.
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
dim g as new GMImageMBS(f)
dim p as new GMPixelsMBS(g)
```

```
// get pointer to some pixels
dim x as ptr = p.get(0, 0, 100, 100)

// and show size
MsgBox str(p.columns)+" x "+str(p.rows)
```

Notes: (Read only property)

4.50.11 handle as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

4.50.12 rows as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Height of view.
Notes: (Read only property)

4.50.13 x as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Left ordinate of view.
Notes: (Read only property)

4.50.14 y as Integer

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Top ordinate of view.
Notes: (Read only property)

4.51 class GMTypeMetricMBS

4.51.1 class GMTypeMetricMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The TypeMetric class provides the means to pass data from the Image class's TypeMetric method to the user.
Notes: It provides information regarding font metrics such as ascent, descent, text width, text height, and maximum horizontal advance. The units of these font metrics are in pixels, and that the metrics are dependent on the current Image font (default Ghostscript's "Helvetica"), pointsize (default 12 points), and x/y resolution (default 72 DPI) settings.

The pixel units may be converted to points (the standard resolution-independent measure used by the typesetting industry) via the following equation:

size_points = (size_pixels * 72)/resolution
where resolution is in dots-per-inch (DPI). This means that at the default image resolution, there is one pixel per point.

Note that a font's pointsize is only a first-order approximation of the font height (ascender + descender) in points. The relationship between the specified pointsize and the rendered font height is determined by the font designer.

See FreeType Glyph Conventions for a detailed description of font metrics related issues.

4.51.2 Methods

4.51.3 Constructor

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

4.51.4 Properties

4.51.5 ascent as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the distance in pixels from the text baseline to the highest/upper grid coordinate used

to place an outline point.
Notes: Always a positive value.
(Read only property)

4.51.6 descent as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the the distance in pixels from the baseline to the lowest grid coordinate used to place an outline point.
Notes: Always a negative value.
(Read only property)

4.51.7 maxHorizontalAdvance as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the maximum horizontal advance (advance from the beginning of a character to the beginning of the next character) in pixels.
Notes: (Read only property)

4.51.8 textHeight as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns text height in pixels.
Notes: (Read only property)

4.51.9 textWidth as Double

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns text width in pixels.
Notes: (Read only property)

4.52 class GMUnsupportedExceptionMBS

4.52.1 class GMUnsupportedExceptionMBS

Plugin Version: 9.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: An exception raised if you call the GM functions on an unsupported platform.
Notes: Check the message property for details.
This exception is currently only used on Windows.
(Windows support may come later)
Subclass of the GMErrorExceptionMBS class.

Chapter 5

Image Magick

5.1 class ImageMagickQ16MBS

5.1.1 class ImageMagickQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for global functions from the Image Magick library
Notes: Before using this class you need to load the ImageMagick dylib or dll.

Not all functions from the library are available through the plugin. If you need something special, please ask.

For Mac OS X you need the ImageMagick dylib/bundle and for Windows the normal ImageMagick installation with the DLL.

For more details please check the ImageMagick documentation.

The plugin implements three versions of this ImageMagick classes. One with Q8 for 8 bit quantum depth, one with Q16 for 16 bit depth and Q32 for 32 bit depth.

5.1.2 Methods

5.1.3 Copyright as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The copyright notice for this format.

Notes: For more details please check the ImageMagick documentation.

5.1.4 Features as String

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick features.
Notes: For example whether library is compiled with OpenMP for faster performance.

5.1.5 HomeURL as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the home url of the library.
Notes: For more details please check the ImageMagick documentation.

5.1.6 InitializeMagick(path as string = "")

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the ImageMagick environment.
Example:

```
dim i as new ImageMagickQ16MBS
i.InitializeMagick("")
```

Notes: Path: The execution path of the current ImageMagick client.

For more details please check the ImageMagick documentation.
You need to call LoadLibrary functions to load the library before calling this.

5.1.7 IsMagickInstantiated as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the ImageMagick environment is currently instantiated.
Notes: In other words: True if InitializeMagick has been called before.

For more details please check the ImageMagick documentation.

5.1.8 LoadErrorString as string

Plugin Version: 9.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error message from loading the image magick library.

5.1.9 LoadLibrary(path as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the dll/bundle on the give path.
Example:

```
dim i as new ImageMagickQ16MBS

if TargetLinux then
if i.LoadLibrary("libMagick.so.6") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
elseif TargetWin32 then
if i.LoadLibrary("CORE_RL_magick_.dll") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
else
// Mac OS X
if i.LoadLibraryFile(GetFolderItem("ImageMagick.bundle")) then
//MsgBox "library loaded."
else
MsgBox "The library failed to load."
end if
end if
```

Notes: In case the loading fails the library may be linked to some other libraries (e.g. X11) and you need to install them to get it working.

On Windows you can just pass the name of the library and the system will search it on the paths in the environment variable "PATH" (or the Windows folder).

On Linux, pass the path or name of the library and the system will search for it.

For more details please check the ImageMagick documentation.

With plugin version 6.1 the Mac OS X part accepts a path to a dylib file, too.
Changed to a shared method in plugin version 10.4.

5.1.10 LoadLibraryFile(path as folderitem) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the dll/bundle on the give path.
Example:

```
dim i as new ImageMagickQ16MBS

if TargetLinux then
if i.LoadLibrary("libMagick.so.6") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
elseif TargetWin32 then
if i.LoadLibrary("CORE_RL_magick_.dll") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
else
// Mac OS X
if i.LoadLibraryFile(GetFolderItem("ImageMagick.bundle")) then
//MsgBox "library loaded."
else
MsgBox "The library failed to load."
end if
end if
```

Notes: In case the loading fails the library may be linked to some other libraries (e.g. X11) and you need to install them to get it working.

This is the preferred way for Mac OS X as paths may not be unique.

For more details please check the ImageMagick documentation.

With plugin version 6.1 the Mac OS X part accepts a folderitem for a dylib file, too.
Changed to a shared method in plugin version 10.4.

5.1.11 MagickInfoList as IMMagickInfoListQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the list of known image formats.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.1.12 MagickToMime(name as string) as string

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the officially registered (or de facto) MIME media-type corresponding to a magick string.
Notes: If there is no registered media-type, then the string ”image/x-magick” (all lower case) is returned.

For more details please check the ImageMagick documentation.

5.1.13 NewImageInfo as IMImageInfoQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new ImageInfo object.
Notes: Returns nil on low memory.
For more details please check the ImageMagick documentation.

5.1.14 NewImageList as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new empty image list.
Notes: For more details please check the ImageMagick documentation.

5.1.15 PackageName as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The package name of the library.
Notes: For more details please check the ImageMagick documentation.

5.1.16 QuantumDepth as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Quantum Depth of the library.
Notes: For more details please check the ImageMagick documentation.

5.1.17 QuantumDepthLibrary as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum depth used to compile the library.
Notes: QuantumDepthLibrary and QuantumDepthPlugin must be equal for the plugin to work correctly. Currently it is compiled for 16bit support.

5.1.18 QuantumRange as String

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum range used by this library.
Notes: Should be a string like ”Q16”.

5.1.19 ReadImage(info as IMImageInfoQ16MBS) as IMImageQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image from a file.
Notes: Sets the last exception property.
Returns nil on any error.
You need to pass in an info object to describe the image.

For more details please check the ImageMagick documentation.

5.1.20 ReadImageFromString(info as IMImageInfoQ16MBS, data as string) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image from a string.
Notes: Sets the last exception property.
Returns nil on any error.
You need to pass in an info object to describe the image.

For more details please check the ImageMagick documentation.

5.1.21 ReadImageHeaderFromString(info as IMImageInfoQ16MBS, data as string) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads the image header.
Notes: Same as ReadImageFromString except the pixel data is not read.
Sets the last exception property.
For more details please check the ImageMagick documentation.

5.1.22 ReleaseDate as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The release date of the library.
Notes: For more details please check the ImageMagick documentation.

5.1.23 SetCurrentDirectory(path as folderitem) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the current working directory.
Notes: This is needed for most installations to point to the folder with the libraries in order for LoadLibrary to find the dependencies.

5.1.24 Version as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The version of the library.
Notes: For more details please check the ImageMagick documentation.

5.1.25 Properties

5.1.26 LastError as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code reported.
Notes: If an exception is raised and it is not a warning exception, this exception code is saved in this property.
(Read and Write property)

5.1.27 LastException as IMExceptionQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception thrown by the Image Magick library.
Notes: You should check this value after every call to the library, process the error and set the property to nil.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.2 class ImageMagickQ32MBS

5.2.1 class ImageMagickQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for global functions from the Image Magick library
Notes: Before using this class you need to load the ImageMagick dylib or dll.

Not all functions from the library are available through the plugin. If you need something special, please ask.

For Mac OS X you need the ImageMagick dylib/bundle and for Windows the normal ImageMagick installation with the DLL.

For more details please check the ImageMagick documentation.

The plugin implements three versions of this ImageMagick classes. One with Q8 for 8 bit quantum depth, one with Q16 for 16 bit depth and Q32 for 32 bit depth.

5.2.2 Methods

5.2.3 Copyright as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The copyright notice for this format.
Notes: For more details please check the ImageMagick documentation.

5.2.4 Features as String

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick features.
Notes: For example whether library is compiled with OpenMP for faster performance.

5.2.5 HomeURL as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the home url of the library.

Notes: For more details please check the ImageMagick documentation.

5.2.6 InitializeMagick(path as string = "")

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the ImageMagick environment.
Example:

```
dim i as new ImageMagickQ32MBS
i.InitializeMagick("")
```

Notes: Path: The execution path of the current ImageMagick client.

For more details please check the ImageMagick documentation.
You need to call LoadLibrary functions to load the library before calling this.

5.2.7 IsMagickInstantiated as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the ImageMagick environment is currently instantiated.
Notes: In other words: True if InitializeMagick has been called before.

For more details please check the ImageMagick documentation.

5.2.8 LoadErrorString as string

Plugin Version: 9.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error message from loading the image magick library.

5.2.9 LoadLibrary(path as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the dll/bundle on the give path.
Example:

```
dim i as new ImageMagickQ32MBS

if TargetLinux then
if i.LoadLibrary("libMagick.so.6") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
elseif TargetWin32 then
if i.LoadLibrary("CORE_RL_magick_.dll") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
else
// Mac OS X
if i.LoadLibraryFile(GetFolderItem("ImageMagick.bundle")) then
//MsgBox "library loaded."
else
MsgBox "The library failed to load."
end if
end if
```

Notes: In case the loading fails the library may be linked to some other libraries (e.g. X11) and you need to install them to get it working.

On Windows you can just pass the name of the library and the system will search it on the paths in the environment variable "PATH" (or the Windows folder).

On Linux, pass the path or name of the library and the system will search for it.

For more details please check the ImageMagick documentation.

With plugin version 6.1 the Mac OS X part accepts a path to a dylib file, too.
Changed to a shared method in plugin version 10.4.

5.2.10 LoadLibraryFile(path as folderitem) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the dll/bundle on the give path.
Example:

```
dim i as new ImageMagickQ32MBS

if TargetLinux then
if i.LoadLibrary("libMagick.so.6") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
elseif TargetWin32 then
if i.LoadLibrary("CORE_RL_magick_.dll") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
else
// Mac OS X
if i.LoadLibraryFile(GetFolderItem("ImageMagick.bundle")) then
//MsgBox "library loaded."
else
MsgBox "The library failed to load."
end if
end if
```

Notes: In case the loading fails the library may be linked to some other libraries (e.g. X11) and you need to install them to get it working.

This is the preferred way for Mac OS X as paths may not be unique.

For more details please check the ImageMagick documentation.

With plugin version 6.1 the Mac OS X part accepts a folderitem for a dylib file, too.
Changed to a shared method in plugin version 10.4.

5.2.11 MagickInfoList as IMMagickInfoListQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the list of known image formats.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.2.12 MagickToMime(name as string) as string

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the officially registered (or de facto) MIME media-type corresponding to a magick string.
Notes: If there is no registered media-type, then the string ”image/x-magick” (all lower case) is returned.

For more details please check the ImageMagick documentation.

5.2.13 NewImageInfo as IMImageInfoQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new ImageInfo object.
Notes: Returns nil on low memory.
For more details please check the ImageMagick documentation.

5.2.14 NewImageList as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new empty image list.
Notes: For more details please check the ImageMagick documentation.

5.2.15 PackageName as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The package name of the library.
Notes: For more details please check the ImageMagick documentation.

5.2.16 QuantumDepth as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Quantum Depth of the library.
Notes: For more details please check the ImageMagick documentation.

5.2.17 QuantumDepthLibrary as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum depth used to compile the library.
Notes: QuantumDepthLibrary and QuantumDepthPlugin must be equal for the plugin to work correctly.
Currently it is compiled for 16bit support.

5.2.18 QuantumRange as String

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum range used by this library.
Notes: Should be a string like "Q16".

5.2.19 ReadImage(info as IMImageInfoQ32MBS) as IMImageQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image from a file.
Notes: Sets the last exception property.
Returns nil on any error.
You need to pass in an info object to describe the image.

For more details please check the ImageMagick documentation.

5.2.20 ReadImageFromString(info as IMImageInfoQ32MBS, data as string) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image from a string.
Notes: Sets the last exception property.
Returns nil on any error.
You need to pass in an info object to describe the image.

For more details please check the ImageMagick documentation.

5.2.21 ReadImageHeaderFromString(info as IMImageInfoQ32MBS, data as string) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads the image header.
Notes: Same as ReadImageFromString except the pixel data is not read.
Sets the last exception property.
For more details please check the ImageMagick documentation.

5.2.22 ReleaseDate as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The release date of the library.
Notes: For more details please check the ImageMagick documentation.

5.2.23 SetCurrentDirectory(path as folderitem) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the current working directory.
Notes: This is needed for most installations to point to the folder with the libraries in order for LoadLibrary to find the dependencies.

5.2.24 Version as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The version of the library.
Notes: For more details please check the ImageMagick documentation.

5.2.25 Properties

5.2.26 LastError as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code reported.
Notes: If an exception is raised and it is not a warning exception, this exception code is saved in this property.

(Read and Write property)

5.2.27 LastException as IMExceptionQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception thrown by the Image Magick library.
Notes: You should check this value after every call to the library, process the error and set the property to nil.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.3 class ImageMagickQ8MBS

5.3.1 class ImageMagickQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for global functions from the Image Magick library
Notes: Before using this class you need to load the ImageMagick dylib or dll.

Not all functions from the library are available through the plugin. If you need something special, please ask.

For Mac OS X you need the ImageMagick dylib/bundle and for Windows the normal ImageMagick installation with the DLL.

For more details please check the ImageMagick documentation.

The plugin implements three versions of this ImageMagick classes. One with Q8 for 8 bit quantum depth, one with Q16 for 16 bit depth and Q32 for 32 bit depth.

5.3.2 Methods

5.3.3 Copyright as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The copyright notice for this format.
Notes: For more details please check the ImageMagick documentation.

5.3.4 Features as String

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick features.
Notes: For example whether library is compiled with OpenMP for faster performance.

5.3.5 HomeURL as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the home url of the library.

Notes: For more details please check the ImageMagick documentation.

5.3.6 InitializeMagick(path as string = "")

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the ImageMagick environment.
Example:

```
dim i as new ImageMagickQ8MBS
i.InitializeMagick("")
```

Notes: Path: The execution path of the current ImageMagick client.

For more details please check the ImageMagick documentation.
You need to call LoadLibrary functions to load the library before calling this.

5.3.7 IsMagickInstantiated as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the ImageMagick environment is currently instantiated.
Notes: In other words: True if InitializeMagick has been called before.

For more details please check the ImageMagick documentation.

5.3.8 LoadErrorString as string

Plugin Version: 9.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error message from loading the image magick library.

5.3.9 LoadLibrary(path as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the dll/bundle on the give path.
Example:

```
dim i as new ImageMagickQ8MBS

if TargetLinux then
if i.LoadLibrary("libMagick.so.6") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
elseif TargetWin32 then
if i.LoadLibrary("CORE_RL_magick_.dll") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
else
// Mac OS X
if i.LoadLibraryFile(GetFolderItem("ImageMagick.bundle")) then
//MsgBox "library loaded."
else
MsgBox "The library failed to load."
end if
end if
```

Notes: In case the loading fails the library may be linked to some other libraries (e.g. X11) and you need to install them to get it working.

On Windows you can just pass the name of the library and the system will search it on the paths in the environment variable "PATH" (or the Windows folder).

On Linux, pass the path or name of the library and the system will search for it.

For more details please check the ImageMagick documentation.

With plugin version 6.1 the Mac OS X part accepts a path to a dylib file, too.
Changed to a shared method in plugin version 10.4.

5.3.10 LoadLibraryFile(path as folderitem) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the dll/bundle on the give path.
Example:

```
dim i as new ImageMagickQ8MBS

if TargetLinux then
if i.LoadLibrary("libMagick.so.6") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
elseif TargetWin32 then
if i.LoadLibrary("CORE_RL_magick_.dll") then
//MsgBox "library loaded."
else
MsgBox "library failed."
end if
else
// Mac OS X
if i.LoadLibraryFile(GetFolderItem("ImageMagick.bundle")) then
//MsgBox "library loaded."
else
MsgBox "The library failed to load."
end if
end if
```

Notes: In case the loading fails the library may be linked to some other libraries (e.g. X11) and you need to install them to get it working.

This is the preferred way for Mac OS X as paths may not be unique.

For more details please check the ImageMagick documentation.

With plugin version 6.1 the Mac OS X part accepts a folderitem for a dylib file, too.
Changed to a shared method in plugin version 10.4.

5.3.11 MagickInfoList as IMMagickInfoListQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the list of known image formats.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.3.12 MagickToMime(name as string) as string

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the officially registered (or de facto) MIME media-type corresponding to a magick string.
Notes: If there is no registered media-type, then the string "image/x-magick" (all lower case) is returned.

For more details please check the ImageMagick documentation.

5.3.13 NewImageInfo as IMImageInfoQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new ImageInfo object.
Notes: Returns nil on low memory.
For more details please check the ImageMagick documentation.

5.3.14 NewImageList as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new empty image list.
Notes: For more details please check the ImageMagick documentation.

5.3.15 PackageName as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The package name of the library.
Notes: For more details please check the ImageMagick documentation.

5.3.16 QuantumDepth as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Quantum Depth of the library.
Notes: For more details please check the ImageMagick documentation.

5.3.17 QuantumDepthLibrary as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum depth used to compile the library.
Notes: QuantumDepthLibrary and QuantumDepthPlugin must be equal for the plugin to work correctly. Currently it is compiled for 16bit support.

5.3.18 QuantumRange as String

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum range used by this library.
Notes: Should be a string like ”Q16”.

5.3.19 ReadImage(info as IMImageInfoQ8MBS) as IMImageQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image from a file.
Notes: Sets the last exception property.
Returns nil on any error.
You need to pass in an info object to describe the image.

For more details please check the ImageMagick documentation.

5.3.20 ReadImageFromString(info as IMImageInfoQ8MBS, data as string) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image from a string.
Notes: Sets the last exception property.
Returns nil on any error.
You need to pass in an info object to describe the image.

For more details please check the ImageMagick documentation.

5.3.21 ReadImageHeaderFromString(info as IMImageInfoQ8MBS, data as string) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads the image header.
Notes: Same as ReadImageFromString except the pixel data is not read.
Sets the last exception property.
For more details please check the ImageMagick documentation.

5.3.22 ReleaseDate as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The release date of the library.
Notes: For more details please check the ImageMagick documentation.

5.3.23 SetCurrentDirectory(path as folderitem) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the current working directory.
Notes: This is needed for most installations to point to the folder with the libraries in order for LoadLibrary to find the dependencies.

5.3.24 Version as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The version of the library.
Notes: For more details please check the ImageMagick documentation.

5.3.25 Properties

5.3.26 LastError as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code reported.
Notes: If an exception is raised and it is not a warning exception, this exception code is saved in this property.

(Read and Write property)

5.3.27 LastException as IMExceptionQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception thrown by the Image Magick library.
Notes: You should check this value after every call to the library, process the error and set the property to nil.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.4 class IMColorQ16MBS

5.4.1 class IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Image Magick class to represent a color.
Example:

```
dim i as IMImageInfoQ16MBS
dim c as IMColorQ16MBS

c=i.BackgroundColor
c.red=65535 // full red
i.BackgroundColor=c
```

Notes: As you see above the IMColorQ16MBS object does not reference the original values, but contains a copy, so you must assign the modified color back to store it.
(Same as on the Xojo color class)

5.4.2 Methods

5.4.3 Constructor

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a color object where all four properties are zero.
See also:

- 5.4.4 Constructor(c as color) 607
- 5.4.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0) 608

5.4.4 Constructor(c as color)

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color value and fills it with the given Xojo color.
See also:

- 5.4.3 Constructor 607
- 5.4.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0) 608

5.4.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0)

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color object with the given values.
See also:

- 5.4.3 Constructor 607
- 5.4.4 Constructor(c as color) 607

5.4.6 Properties

5.4.7 Blue as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.4.8 ColorValue as Color

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color as a Xojo color.
Notes: Please note that Xojo colors are 8 bit. So for Q16 and Q32 classes the colors are scaled up or down.
This reads/writes the red, green and blue property, but not the opacity property.
(Read and Write property)

5.4.9 Green as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.4.10 Opacity as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity part of the color.
Notes: Value from 0 to 65535.
(Read and Write property)

5.4.11 Red as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.5 class IMColorQ32MBS

5.5.1 class IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Image Magick class to represent a color.
Example:

```
dim i as IMImageInfoQ32MBS
dim c as IMColorQ32MBS

c=i.BackgroundColor
c.red=65535 // full red
i.BackgroundColor=c
```

Notes: As you see above the IMColorQ32MBS object does not reference the original values, but contains a copy, so you must assign the modified color back to store it.
(Same as on the Xojo color class)

5.5.2 Methods

5.5.3 Constructor

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a color object where all four properties are zero.
See also:

- 5.5.4 Constructor(c as color) 610
- 5.5.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0) 611

5.5.4 Constructor(c as color)

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color value and fills it with the given Xojo color.
See also:

- 5.5.3 Constructor 610
- 5.5.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0) 611

5.5.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0)

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color object with the given values.
See also:

- 5.5.3 Constructor 610
- 5.5.4 Constructor(c as color) 610

5.5.6 Properties

5.5.7 Blue as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.5.8 ColorValue as Color

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color as a Xojo color.
Notes: Please note that Xojo colors are 8 bit. So for Q16 and Q32 classes the colors are scaled up or down.
This reads/writes the red, green and blue property, but not the opacity property.
(Read and Write property)

5.5.9 Green as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.5.10 Opacity as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity part of the color.
Notes: Value from 0 to 65535.
(Read and Write property)

5.5.11 Red as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.6 class IMColorQ8MBS

5.6.1 class IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Image Magick class to represent a color.
Example:

```
dim i as IMImageInfoQ8MBS
dim c as IMColorQ8MBS

c=i.BackgroundColor
c.red=65535 // full red
i.BackgroundColor=c
```

Notes: As you see above the IMColorQ8MBS object does not reference the original values, but contains a copy, so you must assign the modified color back to store it.
(Same as on the Xojo color class)

5.6.2 Methods

5.6.3 Constructor

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a color object where all four properties are zero.
See also:

- 5.6.4 Constructor(c as color) 613
- 5.6.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0) 614

5.6.4 Constructor(c as color)

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color value and fills it with the given Xojo color.
See also:

- 5.6.3 Constructor 613
- 5.6.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0) 614

5.6.5 Constructor(red as UInt32, green as UInt32, blue as UInt32, Opacity as UInt32 = 0)

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new color object with the given values.
See also:

- 5.6.3 Constructor 613
- 5.6.4 Constructor(c as color) 613

5.6.6 Properties

5.6.7 Blue as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.6.8 ColorValue as Color

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color as a Xojo color.
Notes: Please note that Xojo colors are 8 bit. So for Q16 and Q32 classes the colors are scaled up or down.
This reads/writes the red, green and blue property, but not the opacity property.
(Read and Write property)

5.6.9 Green as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.6.10 Opacity as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity part of the color.
Notes: Value from 0 to 65535.
(Read and Write property)

5.6.11 Red as UInt32

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red component.
Notes: Value from 0 to 65535.
(Read and Write property)

5.7 class IMExceptionQ16MBS

5.7.1 class IMExceptionQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for information about an Image Magick Exception.
Notes: Some functions can throw an exception and you find this exception object after calling the function inside the class. For Example after calling IMImageQ16MBS.resize, the IMImageQ16MBS.LastException property will be nil for no exception or just contain the exception from the resize operation.

For more details please check the ImageMagick documentation.
Subclass of the RuntimeException class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.2pr10

5.7.2 Methods

5.7.3 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.7.4 Properties

5.7.5 Description as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The description of the exception.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.7.6 Reason as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The reason of the exception.
Notes: For more details please check the ImageMagick documentation.

(Read and Write property)

5.7.7 Severity as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception type.
Notes: some usefull constants:

For more details please check the ImageMagick documentation.
(Read and Write property)

5.7.8 Signature as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The signature of the exception.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

```
const UndefinedException           = 0
 const WarningException            = 300
 const ResourceLimitWarning        = 300
 const TypeWarning                 = 305
 const OptionWarning               = 310
 const DelegateWarning             = 315
 const MissingDelegateWarning      = 320
 const CorruptImageWarning         = 325
 const FileOpenWarning             = 330
 const BlobWarning                 = 335
 const StreamWarning               = 340
 const CacheWarning                = 345
 const CoderWarning                = 350
 const ModuleWarning               = 355
 const DrawWarning                 = 360
 const ImageWarning                = 365
 const XServerWarning              = 380
 const MonitorWarning              = 385
 const RegistryWarning             = 390
 const ConfigureWarning            = 395
 const ErrorException              = 400
 const ResourceLimitError          = 400
 const TypeError                   = 405
 const OptionError                 = 410
 const DelegateError               = 415
 const MissingDelegateError        = 420
 const CorruptImageError           = 425
 const FileOpenError               = 430
 const BlobError                   = 435
 const StreamError                 = 440
 const CacheError                  = 445
 const CoderError                  = 450
 const ModuleError                 = 455
 const DrawError                   = 460
 const ImageError                  = 465
 const XServerError                = 480
 const MonitorError                = 485
 const RegistryError               = 490
 const ConfigureError              = 495
 const FatalErrorException         = 700
 const ResourceLimitFatalError     = 700
 const TypeFatalError              = 705
 const OptionFatalError            = 710
 const DelegateFatalError          = 715
 const MissingDelegateFatalError   = 720
 const CorruptImageFatalError      = 725
 const FileOpenFatalError          = 730
 const BlobFatalError              = 735
 const StreamFatalError            = 740
 const CacheFatalError             = 745
 const CoderFatalError             = 750
 const ModuleFatalError            = 755
 const DrawFatalError              = 760
 const ImageFatalError             = 765
 const XServerFatalError           = 780
 const MonitorFatalError           = 785
 const RegistryFatalError          = 790
 const ConfigureFatalError         = 795
```

5.8 class IMExceptionQ32MBS

5.8.1 class IMExceptionQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for information about an Image Magick Exception.
Notes: Some functions can throw an exception and you find this exception object after calling the function inside the class. For Example after calling IMImageQ32MBS.resize, the IMImageQ32MBS.LastException property will be nil for no exception or just contain the exception from the resize operation.

For more details please check the ImageMagick documentation.
Subclass of the RuntimeException class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.2pr10

5.8.2 Methods

5.8.3 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.8.4 Properties

5.8.5 Description as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The description of the exception.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.8.6 Reason as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The reason of the exception.
Notes: For more details please check the ImageMagick documentation.

(Read and Write property)

5.8.7 Severity as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception type.
Notes: some usefull constants:

For more details please check the ImageMagick documentation.
(Read and Write property)

5.8.8 Signature as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The signature of the exception.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

```
const UndefinedException            = 0
 const WarningException             = 300
 const ResourceLimitWarning         = 300
 const TypeWarning                  = 305
 const OptionWarning                = 310
 const DelegateWarning              = 315
 const MissingDelegateWarning       = 320
 const CorruptImageWarning          = 325
 const FileOpenWarning              = 330
 const BlobWarning                  = 335
 const StreamWarning                = 340
 const CacheWarning                 = 345
 const CoderWarning                 = 350
 const ModuleWarning                = 355
 const DrawWarning                  = 360
 const ImageWarning                 = 365
 const XServerWarning               = 380
 const MonitorWarning               = 385
 const RegistryWarning              = 390
 const ConfigureWarning             = 395
 const ErrorException               = 400
 const ResourceLimitError           = 400
 const TypeError                    = 405
 const OptionError                  = 410
 const DelegateError                = 415
 const MissingDelegateError         = 420
 const CorruptImageError            = 425
 const FileOpenError                = 430
 const BlobError                    = 435
 const StreamError                  = 440
 const CacheError                   = 445
 const CoderError                   = 450
 const ModuleError                  = 455
 const DrawError                    = 460
 const ImageError                   = 465
 const XServerError                 = 480
 const MonitorError                 = 485
 const RegistryError                = 490
 const ConfigureError               = 495
 const FatalErrorException          = 700
 const ResourceLimitFatalError      = 700
 const TypeFatalError               = 705
 const OptionFatalError             = 710
 const DelegateFatalError           = 715
 const MissingDelegateFatalError    = 720
 const CorruptImageFatalError       = 725
 const FileOpenFatalError           = 730
 const BlobFatalError               = 735
 const StreamFatalError             = 740
 const CacheFatalError              = 745
 const CoderFatalError              = 750
 const ModuleFatalError             = 755
 const DrawFatalError               = 760
 const ImageFatalError              = 765
 const XServerFatalError            = 780
 const MonitorFatalError            = 785
 const RegistryFatalError           = 790
 const ConfigureFatalError          = 795
```

5.9 class IMExceptionQ8MBS

5.9.1 class IMExceptionQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for information about an Image Magick Exception.
Notes: Some functions can throw an exception and you find this exception object after calling the function inside the class. For Example after calling IMImageQ8MBS.resize, the IMImageQ8MBS.LastException property will be nil for no exception or just contain the exception from the resize operation.

For more details please check the ImageMagick documentation.
Subclass of the RuntimeException class.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 14.2pr10

5.9.2 Methods

5.9.3 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.9.4 Properties

5.9.5 Description as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The description of the exception.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.9.6 Reason as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The reason of the exception.
Notes: For more details please check the ImageMagick documentation.

(Read and Write property)

5.9.7 Severity as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The exception type.
Notes: some usefull constants:

For more details please check the ImageMagick documentation.
(Read and Write property)

5.9.8 Signature as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The signature of the exception.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

```
const UndefinedException             = 0
 const WarningException              = 300
 const ResourceLimitWarning          = 300
 const TypeWarning                   = 305
 const OptionWarning                 = 310
 const DelegateWarning               = 315
 const MissingDelegateWarning        = 320
 const CorruptImageWarning           = 325
 const FileOpenWarning               = 330
 const BlobWarning                   = 335
 const StreamWarning                 = 340
 const CacheWarning                  = 345
 const CoderWarning                  = 350
 const ModuleWarning                 = 355
 const DrawWarning                   = 360
 const ImageWarning                  = 365
 const XServerWarning                = 380
 const MonitorWarning                = 385
 const RegistryWarning               = 390
 const ConfigureWarning              = 395
 const ErrorException                = 400
 const ResourceLimitError            = 400
 const TypeError                     = 405
 const OptionError                   = 410
 const DelegateError                 = 415
 const MissingDelegateError          = 420
 const CorruptImageError             = 425
 const FileOpenError                 = 430
 const BlobError                     = 435
 const StreamError                   = 440
 const CacheError                    = 445
 const CoderError                    = 450
 const ModuleError                   = 455
 const DrawError                     = 460
 const ImageError                    = 465
 const XServerError                  = 480
 const MonitorError                  = 485
 const RegistryError                 = 490
 const ConfigureError                = 495
 const FatalErrorException           = 700
 const ResourceLimitFatalError       = 700
 const TypeFatalError                = 705
 const OptionFatalError              = 710
 const DelegateFatalError            = 715
 const MissingDelegateFatalError     = 720
 const CorruptImageFatalError        = 725
 const FileOpenFatalError            = 730
 const BlobFatalError                = 735
 const StreamFatalError              = 740
 const CacheFatalError               = 745
 const CoderFatalError               = 750
 const ModuleFatalError              = 755
 const DrawFatalError                = 760
 const ImageFatalError               = 765
 const XServerFatalError             = 780
 const MonitorFatalError             = 785
 const RegistryFatalError            = 790
 const ConfigureFatalError           = 795
```

5.10 class IMImageAffineMatrixQ16MBS

5.10.1 class IMImageAffineMatrixQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an Image Magick affine transformation matrix.

5.10.2 Methods

5.10.3 Constructor

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an identity matrix.

5.10.4 Properties

5.10.5 RX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotate x value.
Notes: (Read and Write property)

5.10.6 RY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotate y value.
Notes: (Read and Write property)

5.10.7 SX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale x value.
Notes: (Read and Write property)

5.10.8 SY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale y value.
Notes: (Read and Write property)

5.10.9 TX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate y value.
Notes: (Read and Write property)

5.10.10 TY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate y value.
Notes: (Read and Write property)

5.11 class IMImageAffineMatrixQ32MBS

5.11.1 class IMImageAffineMatrixQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an Image Magick affine transformation matrix.

5.11.2 Methods

5.11.3 Constructor

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an identity matrix.

5.11.4 Properties

5.11.5 RX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotate x value.
Notes: (Read and Write property)

5.11.6 RY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotate y value.
Notes: (Read and Write property)

5.11.7 SX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale x value.
Notes: (Read and Write property)

5.11.8 SY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale y value.
Notes: (Read and Write property)

5.11.9 TX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate y value.
Notes: (Read and Write property)

5.11.10 TY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate y value.
Notes: (Read and Write property)

5.12 class IMImageAffineMatrixQ8MBS

5.12.1 class IMImageAffineMatrixQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an Image Magick affine transformation matrix.

5.12.2 Methods

5.12.3 Constructor

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an identity matrix.

5.12.4 Properties

5.12.5 RX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotate x value.
Notes: (Read and Write property)

5.12.6 RY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotate y value.
Notes: (Read and Write property)

5.12.7 SX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale x value.
Notes: (Read and Write property)

5.12.8 SY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale y value.
Notes: (Read and Write property)

5.12.9 TX as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate y value.
Notes: (Read and Write property)

5.12.10 TY as Double

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate y value.
Notes: (Read and Write property)

5.13 class IMImageAttributeQ16MBS

5.13.1 class IMImageAttributeQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for an image attribute.
Notes: Used only for reading the attributes.
Do not keep references over long times as memory of key/value pairs may be released.

5.13.2 Properties

5.13.3 Compression as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether compression is used.
Notes: (Read only property)

5.13.4 Key as String

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The key of this attribute.
Notes: String is in binary text encoding.
(Read only property)

5.13.5 Value as String

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The value of this attribute.
Notes: String is in binary text encoding.
(Read only property)

5.14 class IMImageAttributeQ32MBS

5.14.1 class IMImageAttributeQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for an image attribute.
Notes: Used only for reading the attributes.
Do not keep references over long times as memory of key/value pairs may be released.

5.14.2 Properties

5.14.3 Compression as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether compression is used.
Notes: (Read only property)

5.14.4 Key as String

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The key of this attribute.
Notes: String is in binary text encoding.
(Read only property)

5.14.5 Value as String

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The value of this attribute.
Notes: String is in binary text encoding.
(Read only property)

5.15 class IMImageAttributeQ8MBS

5.15.1 class IMImageAttributeQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for an image attribute.
Notes: Used only for reading the attributes.
Do not keep references over long times as memory of key/value pairs may be released.

5.15.2 Properties

5.15.3 Compression as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether compression is used.
Notes: (Read only property)

5.15.4 Key as String

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The key of this attribute.
Notes: String is in binary text encoding.
(Read only property)

5.15.5 Value as String

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The value of this attribute.
Notes: String is in binary text encoding.
(Read only property)

5.16 class IMImageInfoQ16MBS

5.16.1 class IMImageInfoQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for information about an image.
Notes: For more details please check the ImageMagick documentation.

5.16.2 Methods

5.16.3 Clone as IMImageInfoQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones the ImageInfo object.
Notes: For more details please check the ImageMagick documentation.

5.16.4 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.16.5 DestroyImageInfo

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys the image info and sets the handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.16.6 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole ImageInfo structure copied into a memoryblock.
Notes: Returns nil on any error.

5.16.7 Properties

5.16.8 Adjoin as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Join images into a single multi-image file.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.9 Affirm as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.10 Antialias as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Control antialiasing of rendered Postscript and Postscript or TrueType fonts.
Notes: Enabled by default.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.11 Authenticate as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.12 BackgroundColor as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Notes: (Read and Write property)

5.16.13 BorderColor as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

5.16.14 Channel as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The channel to use.
Notes: Constants for channel:

```
const UndefinedChannel    = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

(Read and Write property)

5.16.15 Colors as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.16 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Example:

```
dim im as ImageMagickQ16MBS // global

Function IMPictureToString(p as picture, magick as string, quality as Integer) As string
dim image as new IMImageQ16MBS
dim imageinfo as IMImageInfoQ16MBS
dim s,data as string
dim impp as new IMMagickPixelPacketQ16MBS

// empty string for nil picture
if p = nil then
Return ""
end if

// create a new picture info

imageinfo = im.NewImageInfo
imageinfo.ColorSpace=1
// only color space is needed. 1 for RGB.

// background color of image
impp.red = 0
impp.Green = 0
impp.Blue = 0

// creates a new image object
if not image.NewImage(imageinfo,p.Width,p.Height,impp) then
Return ""
end if

// copy RB picture into IM Image at position 0/0
image.ColorSpace = 1
image.SetPicture(p,0,0)

// set compression data
imageinfo.Magick = magick
imageinfo.Quality = quality

// and rendering intent: 2=PerceptualIntent
'image.RenderingIntent = 2

// create image data
data = image.ImageToBlob(imageinfo)

// release memory
image.DestroyImage
```

```
imageinfo.DestroyImageInfo

// return result
Return data

Exception
// in case of an exception return nothing
Return ""


End Function
```

Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

(Read and Write property)

5.16.17 Compression as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.

Notes: useful constants:

```
const UndefinedCompression       = 0
 const NoCompression             = 1
 const BZipCompression           = 2
 const FaxCompression            = 3
 const Group4Compression         = 4
 const JPEGCompression           = 5
 const LosslessJPEGCompression   = 6
 const LZWCompression            = 7
 const RLECompression            = 8
 const ZipCompression            = 9
```

The default is the compression type of the specified image file.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.18 Density as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Vertical and horizontal resolution in pixels of the image.
Notes: This option specifies an image density when decoding a Postscript or Portable Document page.
(Read and Write property)

5.16.19 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: QuantumLeap must be defined before a depth of 16 is valid.
(Read and Write property)

5.16.20 Dither as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.21 Endian as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian setting to use.
Notes: constants:

UndefinedEndian	0	
LSBEndian	1	(Windows)
MSBEndian	2	(Mac)

e.g. tiff files support different endian settings.
(Read and Write property)

5.16.22 Extract as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.23 Filename as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path/name.
Notes: The string must be in the encoding of the library and is limited to 4000 bytes.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.24 Font as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font.
Notes: If the font is a fully qualified X server font name, the font is obtained from an X server. To use a TrueType font, precede the TrueType filename with an @. Otherwise, specify a Postscript font name (e.g. ”helvetica”).
(Read and Write property)

5.16.25 Group as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.26 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a ImageInfo structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.27 HeaderOnly as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if only the header was read from the image data.
Notes: (Read and Write property)

5.16.28 Interlace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.

constants:

(Read and Write property)

UndefinedInterlace	0	Unset value.
NoInterlace	1	Don't interlace image (RGBRGBRGBRGBRGBRGB...)
LineInterlace	2	Use scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...)
PlaneInterlace	3	Use plane interlacing (RRRRRR...GGGGGG...BBBBBB...)
PartitionInterlace	4	Similar to plane interlaing except that the different planes are saved to individual files (e.g. image.R, image.G, and image.B)

5.16.29 Magick as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image encoding format (e.g. "GIF").
Example:

```
dim imageinfo as IMImageInfoQ16MBS
dim blob as string
dim image as IMImageQ16MBS

// Now lets convert to tiff
imageinfo.Filename = "image"
imageinfo.Magick="JPEG"
imageinfo.Quality = 10 //since we are displaying, lets use highest quality, lowest compression
blob = image.ImageToBlob(imageinfo)
```

Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.30 MatteColor as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (transparent) color.
Notes: (Read and Write property)

5.16.31 Monochrome as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform the image to black and white.
Notes: (Read and Write property)

5.16.32 Orientation as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image orientation.
Notes: constants:

```
const UndefinedOrientation      = 0
 const TopLeftOrientation       = 1
 const TopRightOrientation      = 2
 const BottomRightOrientation   = 3
 const BottomLeftOrientation    = 4
 const LeftTopOrientation       = 5
 const RightTopOrientation      = 6
 const RightBottomOrientation   = 7
 const LeftBottomOrientation    = 8
```

For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.33 Page as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Equivalent size of Postscript page.
Notes: (Read and Write property)

5.16.34 PointSize as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font point size.
Notes: (Read and Write property)

5.16.35 Preview as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image manipulation preview option.
Notes: Used by 'display'.

constants:

UndefinedPreview	0
RotatePreview	1
ShearPreview	2
RollPreview	3
HuePreview	4
SaturationPreview	5
BrightnessPreview	6
GammaPreview	7
SpiffPreview	8
DullPreview	9
GrayscalePreview	10
QuantizePreview	11
DespecklePreview	12
ReduceNoisePreview	13
AddNoisePreview	14
SharpenPreview	15
BlurPreview	16
ThresholdPreview	17
EdgeDetectPreview	18
SpreadPreview	19
SolarizePreview	20
ShadePreview	21
RaisePreview	22
SegmentPreview	23
SwirlPreview	24
ImplodePreview	25
WavePreview	26
OilPaintPreview	27
CharcoalDrawingPreview	28
JPEGPreview	29

(Read and Write property)

5.16.36 Quality as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level.
Notes: Default value is 75.
(Read and Write property)

5.16.37 Release as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: If true, the destructor will release the handle.
Notes: (Read and Write property)

5.16.38 ResolutionUnits as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Notes: constants:

UndefinedResolution	0	Unset value.
PixelsPerInchResolution	1	Density specifications are specified in units of pixels per inch (english units).
PixelsPerCentimeterResolution	2	Density specifications are specified in units of pixels per centimeter (metric units).

(Read and Write property)

5.16.39 SamplingFactor as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.40 Scene as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.41 SceneCount as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.42 Scenes as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.16.43 ServerName as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: X11 display to display to.
Notes: obtain fonts from, or to capture image from.
(Read and Write property)

5.16.44 Size as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height of a raw image (an image which does not support width and height information).
Notes: Size may also be used to affect the image size read from a multi-resolution format (e.g. Photo CD, JBIG, or JPEG.
(Read and Write property)

5.16.45 Temporary as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.16.46 Texture as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image filename to use as background texture.
Notes: (Read and Write property)

5.16.47 Type as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Image type.
Notes: constants:

UndefinedType	0
BilevelType	1
GrayscaleType	2
GrayscaleMatteType	3
PaletteType	4
PaletteMatteType	5
TrueColorType	6
TrueColorMatteType	7
ColorSeparationType	8
ColorSeparationMatteType	9
OptimizeType	10

(Read and Write property)

5.16.48 Verbose as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Print detailed information about the image if True.
Notes: (Read and Write property)

5.16.49 View as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: FlashPix viewing parameters.
Notes: (Read and Write property)

5.17 class IMImageInfoQ32MBS

5.17.1 class IMImageInfoQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for information about an image.
Notes: For more details please check the ImageMagick documentation.

5.17.2 Methods

5.17.3 Clone as IMImageInfoQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones the ImageInfo object.
Notes: For more details please check the ImageMagick documentation.

5.17.4 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.17.5 DestroyImageInfo

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys the image info and sets the handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.17.6 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole ImageInfo structure copied into a memoryblock.
Notes: Returns nil on any error.

5.17.7 Properties

5.17.8 Adjoin as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Join images into a single multi-image file.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.9 Affirm as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.10 Antialias as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Control antialiasing of rendered Postscript and Postscript or TrueType fonts.
Notes: Enabled by default.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.11 Authenticate as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.12 BackgroundColor as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Notes: (Read and Write property)

5.17.13 BorderColor as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

5.17.14 Channel as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The channel to use.
Notes: Constants for channel:

```
const UndefinedChannel    = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

(Read and Write property)

5.17.15 Colors as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.16 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Example:

```
dim im as ImageMagickQ32MBS // global

Function IMPictureToString(p as picture, magick as string, quality as Integer) As string
dim image as new IMImageQ32MBS
dim imageinfo as IMImageInfoQ32MBS
dim s,data as string
dim impp as new IMMagickPixelPacketQ32MBS

// empty string for nil picture
if p = nil then
Return ""
end if

// create a new picture info

imageinfo = im.NewImageInfo
imageinfo.ColorSpace=1
// only color space is needed. 1 for RGB.

// background color of image
impp.red = 0
impp.Green = 0
impp.Blue = 0

// creates a new image object
if not image.NewImage(imageinfo,p.Width,p.Height,impp) then
Return ""
end if

// copy RB picture into IM Image at position 0/0
image.ColorSpace = 1
image.SetPicture(p,0,0)

// set compression data
imageinfo.Magick = magick
imageinfo.Quality = quality

// and rendering intent: 2=PerceptualIntent
'image.RenderingIntent = 2

// create image data
data = image.ImageToBlob(imageinfo)

// release memory
image.DestroyImage
```

```
imageinfo.DestroyImageInfo

// return result
Return data

Exception
// in case of an exception return nothing
Return ””


End Function
```

Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

(Read and Write property)

5.17.17 Compression as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.

Notes: useful constants:

```
const UndefinedCompression         = 0
 const NoCompression                = 1
 const BZipCompression              = 2
 const FaxCompression               = 3
 const Group4Compression            = 4
 const JPEGCompression              = 5
 const LosslessJPEGCompression      = 6
 const LZWCompression               = 7
 const RLECompression               = 8
 const ZipCompression               = 9
```

The default is the compression type of the specified image file.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.18 Density as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Vertical and horizontal resolution in pixels of the image.
Notes: This option specifies an image density when decoding a Postscript or Portable Document page.
(Read and Write property)

5.17.19 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: QuantumLeap must be defined before a depth of 16 is valid.
(Read and Write property)

5.17.20 Dither as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.21 Endian as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian setting to use.
Notes: constants:

UndefinedEndian	0	
LSBEndian	1	(Windows)
MSBEndian	2	(Mac)

e.g. tiff files support different endian settings.
(Read and Write property)

5.17.22 Extract as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.23 Filename as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path/name.
Notes: The string must be in the encoding of the library and is limited to 4000 bytes.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.24 Font as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font.
Notes: If the font is a fully qualified X server font name, the font is obtained from an X server. To use a TrueType font, precede the TrueType filename with an @. Otherwise, specify a Postscript font name (e.g. ”helvetica”).
(Read and Write property)

5.17.25 Group as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.26 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a ImageInfo structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.27 HeaderOnly as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if only the header was read from the image data.
Notes: (Read and Write property)

5.17.28 Interlace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.

constants:

(Read and Write property)

UndefinedInterlace	0	Unset value.
NoInterlace	1	Don't interlace image (RGBRGBRGBRGBRGBRGB...)
LineInterlace	2	Use scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...)
PlaneInterlace	3	Use plane interlacing (RRRRRR...GGGGGG...BBBBBB...)
PartitionInterlace	4	Similar to plane interlaing except that the different planes are saved to individual files (e.g. image.R, image.G, and image.B)

5.17.29 Magick as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image encoding format (e.g. "GIF").
Example:

```
dim imageinfo as IMImageInfoQ32MBS
dim blob as string
dim image as IMImageQ32MBS

// Now lets convert to tiff
imageinfo.Filename = "image"
imageinfo.Magick="JPEG"
imageinfo.Quality = 10 //since we are displaying, lets use highest quality, lowest compression
blob = image.ImageToBlob(imageinfo)
```

Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.30 MatteColor as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (transparent) color.
Notes: (Read and Write property)

5.17.31 Monochrome as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform the image to black and white.
Notes: (Read and Write property)

5.17.32 Orientation as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image orientation.
Notes: constants:

```
const UndefinedOrientation      = 0
 const TopLeftOrientation        = 1
 const TopRightOrientation       = 2
 const BottomRightOrientation    = 3
 const BottomLeftOrientation     = 4
 const LeftTopOrientation        = 5
 const RightTopOrientation       = 6
 const RightBottomOrientation    = 7
 const LeftBottomOrientation     = 8
```

For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.33 Page as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Equivalent size of Postscript page.
Notes: (Read and Write property)

5.17.34 PointSize as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font point size.
Notes: (Read and Write property)

5.17.35 Preview as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image manipulation preview option.
Notes: Used by 'display'.

constants:

UndefinedPreview	0
RotatePreview	1
ShearPreview	2
RollPreview	3
HuePreview	4
SaturationPreview	5
BrightnessPreview	6
GammaPreview	7
SpiffPreview	8
DullPreview	9
GrayscalePreview	10
QuantizePreview	11
DespecklePreview	12
ReduceNoisePreview	13
AddNoisePreview	14
SharpenPreview	15
BlurPreview	16
ThresholdPreview	17
EdgeDetectPreview	18
SpreadPreview	19
SolarizePreview	20
ShadePreview	21
RaisePreview	22
SegmentPreview	23
SwirlPreview	24
ImplodePreview	25
WavePreview	26
OilPaintPreview	27
CharcoalDrawingPreview	28
JPEGPreview	29

(Read and Write property)

5.17.36 Quality as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level.
Notes: Default value is 75.
(Read and Write property)

5.17.37 Release as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: If true, the destructor will release the handle.
Notes: (Read and Write property)

5.17.38 ResolutionUnits as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Notes: constants:

UndefinedResolution	0	Unset value.
PixelsPerInchResolution	1	Density specifications are specified in units of pixels per inch (english units).
PixelsPerCentimeterResolution	2	Density specifications are specified in units of pixels per centimeter (metric units).

(Read and Write property)

5.17.39 SamplingFactor as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.40 Scene as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.41 SceneCount as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.42 Scenes as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.17.43 ServerName as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: X11 display to display to.
Notes: obtain fonts from, or to capture image from.
(Read and Write property)

5.17.44 Size as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height of a raw image (an image which does not support width and height information).
Notes: Size may also be used to affect the image size read from a multi-resolution format (e.g. Photo CD, JBIG, or JPEG.
(Read and Write property)

5.17.45 Temporary as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.17.46 Texture as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image filename to use as background texture.
Notes: (Read and Write property)

5.17.47 Type as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Image type.
Notes: constants:

UndefinedType	0
BilevelType	1
GrayscaleType	2
GrayscaleMatteType	3
PaletteType	4
PaletteMatteType	5
TrueColorType	6
TrueColorMatteType	7
ColorSeparationType	8
ColorSeparationMatteType	9
OptimizeType	10

(Read and Write property)

5.17.48 Verbose as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Print detailed information about the image if True.
Notes: (Read and Write property)

5.17.49 View as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: FlashPix viewing parameters.
Notes: (Read and Write property)

5.18 class IMImageInfoQ8MBS

5.18.1 class IMImageInfoQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for information about an image.
Notes: For more details please check the ImageMagick documentation.

5.18.2 Methods

5.18.3 Clone as IMImageInfoQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones the ImageInfo object.
Notes: For more details please check the ImageMagick documentation.

5.18.4 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.18.5 DestroyImageInfo

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys the image info and sets the handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.18.6 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole ImageInfo structure copied into a memoryblock.
Notes: Returns nil on any error.

5.18.7 Properties

5.18.8 Adjoin as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Join images into a single multi-image file.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.9 Affirm as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.10 Antialias as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Control antialiasing of rendered Postscript and Postscript or TrueType fonts.
Notes: Enabled by default.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.11 Authenticate as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.12 BackgroundColor as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Notes: (Read and Write property)

5.18.13 BorderColor as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

5.18.14 Channel as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The channel to use.
Notes: Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

(Read and Write property)

5.18.15 Colors as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.16 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Example:

```
dim im as ImageMagickQ8MBS // global

Function IMPictureToString(p as picture, magick as string, quality as Integer) As string
dim image as new IMImageQ8MBS
dim imageinfo as IMImageInfoQ8MBS
dim s,data as string
dim impp as new IMMagickPixelPacketQ8MBS

// empty string for nil picture
if p = nil then
Return ""
end if

// create a new picture info

imageinfo = im.NewImageInfo
imageinfo.ColorSpace=1
// only color space is needed. 1 for RGB.

// background color of image
impp.red = 0
impp.Green = 0
impp.Blue = 0

// creates a new image object
if not image.NewImage(imageinfo,p.Width,p.Height,impp) then
Return ""
end if

// copy RB picture into IM Image at position 0/0
image.ColorSpace = 1
image.SetPicture(p,0,0)

// set compression data
imageinfo.Magick = magick
imageinfo.Quality = quality

// and rendering intent: 2=PerceptualIntent
'image.RenderingIntent = 2

// create image data
data = image.ImageToBlob(imageinfo)

// release memory
image.DestroyImage
```

```
imageinfo.DestroyImageInfo

// return result
Return data

Exception
// in case of an exception return nothing
Return ””


End Function
```

Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

(Read and Write property)

5.18.17 Compression as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.

Notes: useful constants:

```
const UndefinedCompression       = 0
 const NoCompression             = 1
 const BZipCompression           = 2
 const FaxCompression            = 3
 const Group4Compression         = 4
 const JPEGCompression           = 5
 const LosslessJPEGCompression   = 6
 const LZWCompression            = 7
 const RLECompression            = 8
 const ZipCompression            = 9
```

The default is the compression type of the specified image file.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.18 Density as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Vertical and horizontal resolution in pixels of the image.
Notes: This option specifies an image density when decoding a Postscript or Portable Document page.
(Read and Write property)

5.18.19 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: QuantumLeap must be defined before a depth of 16 is valid.
(Read and Write property)

5.18.20 Dither as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.21 Endian as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian setting to use.
Notes: constants:

UndefinedEndian	0	
LSBEndian	1	(Windows)
MSBEndian	2	(Mac)

e.g. tiff files support different endian settings.
(Read and Write property)

5.18.22 Extract as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.23 Filename as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path/name.
Notes: The string must be in the encoding of the library and is limited to 4000 bytes.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.24 Font as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font.
Notes: If the font is a fully qualified X server font name, the font is obtained from an X server. To use a TrueType font, precede the TrueType filename with an @. Otherwise, specify a Postscript font name (e.g. ”helvetica”).
(Read and Write property)

5.18.25 Group as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.26 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a ImageInfo structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.27 HeaderOnly as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if only the header was read from the image data.
Notes: (Read and Write property)

5.18.28 Interlace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.

constants:

(Read and Write property)

UndefinedInterlace	0	Unset value.
NoInterlace	1	Don't interlace image (RGBRGBRGBRGBRGBRGB...)
LineInterlace	2	Use scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...)
PlaneInterlace	3	Use plane interlacing (RRRRRR...GGGGGG...BBBBBB...)
PartitionInterlace	4	Similar to plane interlaing except that the different planes are saved to individual files (e.g. image.R, image.G, and image.B)

5.18.29 Magick as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image encoding format (e.g. "GIF").
Example:

```
dim imageinfo as IMImageInfoQ8MBS
dim blob as string
dim image as IMImageQ8MBS

// Now lets convert to tiff
imageinfo.Filename = "image"
imageinfo.Magick="JPEG"
imageinfo.Quality = 10 //since we are displaying, lets use highest quality, lowest compression
blob = image.ImageToBlob(imageinfo)
```

Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.30 MatteColor as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (transparent) color.
Notes: (Read and Write property)

5.18.31 Monochrome as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transform the image to black and white.
Notes: (Read and Write property)

5.18.32 Orientation as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image orientation.
Notes: constants:

```
const UndefinedOrientation      = 0
 const TopLeftOrientation       = 1
 const TopRightOrientation      = 2
 const BottomRightOrientation   = 3
 const BottomLeftOrientation    = 4
 const LeftTopOrientation       = 5
 const RightTopOrientation      = 6
 const RightBottomOrientation   = 7
 const LeftBottomOrientation    = 8
```

For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.33 Page as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Equivalent size of Postscript page.
Notes: (Read and Write property)

5.18.34 PointSize as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Text rendering font point size.
Notes: (Read and Write property)

5.18.35 Preview as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image manipulation preview option.
Notes: Used by 'display'.

constants:

UndefinedPreview	0
RotatePreview	1
ShearPreview	2
RollPreview	3
HuePreview	4
SaturationPreview	5
BrightnessPreview	6
GammaPreview	7
SpiffPreview	8
DullPreview	9
GrayscalePreview	10
QuantizePreview	11
DespecklePreview	12
ReduceNoisePreview	13
AddNoisePreview	14
SharpenPreview	15
BlurPreview	16
ThresholdPreview	17
EdgeDetectPreview	18
SpreadPreview	19
SolarizePreview	20
ShadePreview	21
RaisePreview	22
SegmentPreview	23
SwirlPreview	24
ImplodePreview	25
WavePreview	26
OilPaintPreview	27
CharcoalDrawingPreview	28
JPEGPreview	29

(Read and Write property)

5.18.36 Quality as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level.
Notes: Default value is 75.
(Read and Write property)

5.18.37 Release as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: If true, the destructor will release the handle.
Notes: (Read and Write property)

5.18.38 ResolutionUnits as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Notes: constants:

UndefinedResolution	0	Unset value.
PixelsPerInchResolution	1	Density specifications are specified in units of pixels per inch (english units).
PixelsPerCentimeterResolution	2	Density specifications are specified in units of pixels per centimeter (metric units).

(Read and Write property)

5.18.39 SamplingFactor as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.40 Scene as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.41 SceneCount as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.42 Scenes as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.18.43 ServerName as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: X11 display to display to.
Notes: obtain fonts from, or to capture image from.
(Read and Write property)

5.18.44 Size as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width and height of a raw image (an image which does not support width and height information).
Notes: Size may also be used to affect the image size read from a multi-resolution format (e.g. Photo CD, JBIG, or JPEG.
(Read and Write property)

5.18.45 Temporary as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.18.46 Texture as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image filename to use as background texture.
Notes: (Read and Write property)

5.18.47 Type as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Image type.
Notes: constants:

UndefinedType	0
BilevelType	1
GrayscaleType	2
GrayscaleMatteType	3
PaletteType	4
PaletteMatteType	5
TrueColorType	6
TrueColorMatteType	7
ColorSeparationType	8
ColorSeparationMatteType	9
OptimizeType	10

(Read and Write property)

5.18.48 Verbose as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Print detailed information about the image if True.
Notes: (Read and Write property)

5.18.49 View as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: FlashPix viewing parameters.
Notes: (Read and Write property)

5.19 class IMImageQ16MBS

5.19.1 class IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for an Image Magick Image in memory.
Notes: Can exist with or without pixel data.

For more details please check the ImageMagick documentation.
Blog Entries

- ImageMagick 7 for Xojo
- MBS Xojo / Real Studio Plugins, version 15.1pr4

5.19.2 Methods

5.19.3 AdaptiveThreshold(width as Integer, height as Integer, offset as Integer) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: AdaptiveThreshold selects an individual threshold for each pixel based on the range of intensity values in its local neighborhood.
Notes: This allows for thresholding of an image whose global intensity histogram doesn't contain distinctive peaks.
Sets the last exception property.

width: The width of the local neighborhood.
height: The height of the local neighborhood.
offset: The mean offset.

For more details please check the ImageMagick documentation.

5.19.4 AddNoise(NoiseType as Integer) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds random noise to the image.
Notes: Constants

UndefinedNoise	=0
UniformNoise	=1
GaussianNoise	=2
MultiplicativeGaussianNoise	=3
ImpulseNoise	=4
LaplacianNoise	=5
PoissonNoise	=6

For more details please check the ImageMagick documentation.
Sets the last exception property.

5.19.5 AffineTransformImage(matrix as IMImageAffineMatrixQ16MBS) as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transforms an image as dictated by the affine matrix.

5.19.6 AppendImageToList(img as IMImageQ16MBS)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds an image to the image list.
Notes: For more details please check the ImageMagick documentation.

5.19.7 AutoGammaImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoGammaImage extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.
Notes: Returns true on success or false on failure.

5.19.8 AutoGammaImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoGammaImage extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.

Notes: Returns true on success or false on failure.

channelType: The channels to auto-level. If the special 'SyncChannels' flag is set all given channels is adjusted in the same way using the mean average of those channels.

Constants for channel:

```
const UndefinedChannel    = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

5.19.9 AutoLevelImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoLevelImage adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: Returns true on success or false on failure.

5.19.10 AutoLevelImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoLevelImage adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: Returns true on success or false on failure.

ChannelType: The channels to auto-level. If the special 'SyncChannels' flag is set the min/max/mean value of all given channels is used for all given channels, to all channels in the same way.

Constants for channel:

const UndefinedChannel	= 0
const RedChannel	= &h0001
const GrayChannel	= &h0001
const CyanChannel	= &h0001
const GreenChannel	= &h0002
const MagentaChannel	= &h0002
const BlueChannel	= &h0004
const YellowChannel	= &h0004
const AlphaChannel	= &h0008
const OpacityChannel	= &h0008
const BlackChannel	= &h0020
const IndexChannel	= &h0020
const AllChannels	= &h7fffffff

5.19.11 Average as IMImageQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Average() method takes a set of images and averages them together.
Notes: Each image in the set must have the same width and height. Average() returns a single image with each corresponding pixel component of each image averaged. On failure, a nil image is returned and exception describes the reason for the failure.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.12 BilevelChannel(channel as Integer, threshold as Double) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the value of individual pixels based on the intensity of each pixel channel.
Notes: The result is a high-contrast image.

channel: The channel type.
threshold: define the threshold values.

Constants for channel:

For more details please check the ImageMagick documentation.

```
const UndefinedChannel    = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

5.19.13 BlackThreshold(threshold as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: BlackThreshold is like Threshold but forces all pixels below the threshold into black while leaving all pixels above the threshold unchanged.
Notes: No exceptions are generated.
threshold: Define the threshold value. (ASCII string)
For more details please check the ImageMagick documentation.

5.19.14 BlobSize as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The expected size for this image written to a file.
Notes: For more details please check the ImageMagick documentation.

5.19.15 Blur(radius as Double, sigma as Double) as IMImageQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and BlurImage selects a suitable radius for you.

radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Gaussian, in pixels.

For more details please check the ImageMagick documentation.

5.19.16 BlurImageChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and BlurImageChannel selects a suitable radius for you.

channel: The channel type.
radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Gaussian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.19.17 BorderImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Surrounds the image with a border of the color defined by the bordercolor member of the image.
Notes: The width and height of the border are defined by the corresponding parameters.

5.19.18 BrightnessContrastImage(brightness as Double, contrast as Double) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image. It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.
Notes: Returns true on success or false on failure.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).

5.19.19 BrightnessContrastImageChannel(ChannelType as Integer, brightness as Double, contrast as Double) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image. It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.
Notes: Returns true on success or false on failure.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).
ChannelType: The channels to use.

Constants for channel:

5.19.20 Charcoal(radius as Double, sigma as Double) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Charcoal creates a new image that is a copy of an existing one with the edge highlighted.
Notes: radius: the radius of the pixel neighborhood.
sigma: The standard deviation of the Gaussian, in pixels.

Returns nil on any error.
Sets the last exception property.

```
const UndefinedChannel   = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

5.19.21 Chop(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chop removes a region of an image and collapses the image to occupy the removed portion.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.22 ClipPath(path as string, inside as boolean) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the image clip mask based any clipping path information if it exists.
Notes:

pathname: name of clipping path resource. If name is preceded by #, use clipping path numbered by name.

inside: if true, later operations take effect inside clipping path. Otherwise later operations take effect outside clipping path.

Returns true on success and false on any error.

5.19.23 Clone as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of this image object.
Notes: For more details please check the ImageMagick documentation.

5.19.24 CloneImageAttributes(image as IMImageAttributeQ16MBS) as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: CloneImageAttributes() clones one or more image attributes.
Notes: Returns false on any error.

5.19.25 CloneImageProfiles(SourceImage as IMImageQ16MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones one or more image profiles.
Notes: Returns false on any error and true on success.

5.19.26 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.19.27 ClutImage(clutImage as IMImageQ16MBS) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either

for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

ClutImage: the color lookup table image for replacement color values.

Returns true on success or false on failure.

5.19.28 ClutImageChannel(ChannelType as Integer, clutImage as IMImageQ16MBS) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

ClutImage: the color lookup table image for replacement color values.
ChannelType: The channels to use.

Returns true on success or false on failure.

Constants for channel:

5.19.29 CoalesceImages as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: CoalesceImages composites a set of images while respecting any page offsets and disposal meth-

```
const UndefinedChannel   = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

ods.
Notes: GIF, MIFF, and MNG animation sequences typically start with an image background and each subsequent image varies in size and offset. CoalesceImages() returns a new sequence where each image in the sequence is the same size as the first and composited with the next image in the sequence.

Returns nil on any error.
Sets the last exception property.

5.19.30 Colorize(opacity as string, PenColorRed as Integer, PenColorGreen as Integer, PenColorBlue as Integer, PenColorOpacity as Integer) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ColorizeImage creates a new image that is a copy of an existing one with the image pixels colorized.
Notes: The colorization is controlled with the pen color and the opacity levels.

opacity: A character string indicating the level of opacity as a percentage (0-100).
PenColorRed, PenColorGreen, PenColorBlue and PenColorOpacity define the pen color used.

Returns nil on any error.
Sets the last exception property.

5.19.31 Combine(channel as Integer) as IMImageQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Combines one or more images into a single image.
Notes: The grayscale value of the pixels of each image in the sequence is assigned in order to the specified channels of the combined image. The typical ordering would be image 1 =>Red, 2 =>Green, 3 =>Blue, etc.

The lastexception property is set.

5.19.32 CompareImageLayers(ImageLayerMethod as Integer) as IMImageQ16MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: CompareImageLayers() compares each image with the next in a sequence and returns the minimum bounding region of all the pixel differences (of the mageLayerMethod specified) it discovers.
Notes: Images do NOT have to be the same size, though it is best that all the images are 'coalesced' (images are all the same size, on a flattened canvas, so as to represent exactly how an specific frame should look).

No GIF dispose methods are applied, so GIF animations must be coalesced before applying this image operator to find differences to them.

ImageLayerMethod:
the layers type to compare images with. Must be one of... CompareAnyLayer, CompareClearLayer, CompareOverlayLayer.

Can raise an exception.

5.19.33 Composite(ComposeOperator as Integer, Image as IMImageQ16MBS, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the second image composited onto the first at the specified offsets.
Notes: compose: Specifies an image composite operator.
Image: The second image.
x: An integer that specifies the column offset of the composited image.
y: An integer that specifies the row offset of the composited image.

No error code and exception!

5.19.34 ConsolidateCMYKImages as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Consolidates a sequence of CMYK images.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.35 ContrastImage(sharpen as boolean) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the intensity differences between the lighter and darker elements of the image.
Notes: Returns true on success or false on failure.
Set sharpen to true to increase the image contrast otherwise the contrast is reduced.

5.19.36 CopyPicture as picture

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ16MBS // your image
Canvas1.Backdrop=image.CopyPicture
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
See also:

- 5.19.37 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture 688

5.19.37 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a portion of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ16MBS // your image
Canvas1.Backdrop=image.CopyPicture(0,0,image.Width,image.Height)
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.
See also:

- 5.19.36 CopyPicture as picture 688

5.19.38 CopyPictureMask as picture

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the mask of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ16MBS // your image
Canvas1.Backdrop=image.CopyPictureMask
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
See also:

- 5.19.39 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture 689

5.19.39 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a portion of the mask of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ16MBS // your image
Canvas1.Backdrop=image.CopyPictureMask(0,0,image.Width,image.Height)
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.
See also:

- 5.19.38 CopyPictureMask as picture 689

5.19.40 CopyPixel(x as Integer, y as Integer) as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a pixel.
Notes: Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.

5.19.41 CreateHBITMAP as Ptr

Plugin Version: 15.1, Platform: Windows, Targets: All.

Function: Creates a HBITMAP for the image for use with Windows Declares.
Notes: The HBITMAP returned needs to be freed when you are done with it or you risk having a memory leak.

5.19.42 Crop(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crop extracts a region of the image starting at the offset defined by geometry.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.43 CropImageToTiles(CropGeometry as string) as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crops a single image, into a possible list of tiles.
Notes: This may include a single sub-region of the image. This basically applies all the normal geometry flags for Crop.

5.19.44 CycleColormap(displace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Displaces an image's colormap by a given number of positions.
Notes: If you cycle the colormap a number of times you can produce a psychodelic effect.
Returns true on success.
displace: displace the colormap this amount.

5.19.45 DecipherImage(passkey as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts cipher pixels to plain pixels.
Notes: Passkey: decipher cipher pixels with this passphrase.
Returns true on success.

5.19.46 DeconstructImages as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: DeconstructImages() compares each image with the next in a sequence and returns the minimum bounding region of all differences from the first image.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.47 DeleteImageAttribute(key as string) as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: DeleteImageAttribute() deletes an attribute from the image.
Notes: Returns false on any error.

5.19.48 Despeckle() as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduces the speckle noise in an image while perserving the edges of the original image.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.19.49 DestroyImage

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases the memory used for this image and sets handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.19.50 DestroyImageAttributes

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Deallocates memory associated with the image attribute list.

5.19.51 DestroyImageList

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys the image list and sets the handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.19.52 DestroyImageProfiles

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases memory associated with an image profile map.

5.19.53 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ16MBS

Plugin Version: 12.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: DistortImage() distorts an image using various distortion methods, by mapping color lookups of the source image to a new destination image usally of the same size as the source image, unless 'bestfit' is set to true.
Notes: If 'bestfit' is enabled, and distortion allows it, the destination image is adjusted to ensure the whole source 'image' will just fit within the final destination image, which will be sized and offset accordingly. Also in many cases the virtual offset of the source image will be taken into account in the mapping.

If the '-verbose' control option has been set print to standard error the equicelent '-fx' formula with coefficients for the function, if practical.

A description of each parameter follows:
self: the image to be distorted.
m: the method of image distortion. ArcDistortion always ignores source image offset, and always 'bestfit' the destination image with the top left corner offset relative to the polar mapping center. Affine, Perspective, and Bilinear, do least squares fitting of the distrotion when more than the minimum number of control point pairs are provided. Perspective, and Bilinear, fall back to a Affine distortion when less than 4 control point pairs are provided. While Affine distortions let you use any number of control point pairs, that is Zero pairs is a No-Op (viewport only) distortion, one pair is a translation and two pairs of control points do a scale-rotate-translate, without any shearing.
values: arguments given.
bestfit: Attempt to 'bestfit' the size of the resulting image. This also forces the resulting image to be a 'layered' virtual canvas image. Can be overridden using 'distort:viewport' setting.

Extra Controls from Image meta-data (artifacts)...

- "verbose" Output to stderr alternatives, internal coefficents, and FX equivalents for the distortion operation (if feasible). This forms an extra check of the distortion method, and allows users access to the internal constants IM calculates for the distortion.
- "distort:viewport" Directly set the output image canvas area and offest to use for the resulting image, rather than use the original images canvas, or a calculated 'bestfit' canvas.
- "distort:scale" Scale the size of the output canvas by this amount to provide a method of Zooming, and for super-sampling the results.

Other settings that can effect results include

- 'interpolate' For source image lookups (scale enlargements)
- 'filter' Set filter to use for area-resampling (scale shrinking). Set to 'point' to turn off and use 'interpolate' lookup instead

See also:

- 5.19.54 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ16MBS 693

5.19.54 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ16MBS

Plugin Version: 12.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: DistortImage() distorts an image using various distortion methods, by mapping color lookups of the source image to a new destination image usally of the same size as the source image, unless 'bestfit' is set to true.
Notes: If 'bestfit' is enabled, and distortion allows it, the destination image is adjusted to ensure the whole source 'image' will just fit within the final destination image, which will be sized and offset accordingly. Also in many cases the virtual offset of the source image will be taken into account in the mapping.

If the '-verbose' control option has been set print to standard error the equicelent '-fx' formula with coefficients for the function, if practical.

A description of each parameter follows:
self: the image to be distorted.
m: the method of image distortion. ArcDistortion always ignores source image offset, and always 'bestfit' the destination image with the top left corner offset relative to the polar mapping center. Affine, Perspective, and Bilinear, do least squares fitting of the distrotion when more than the minimum number of control point pairs are provided. Perspective, and Bilinear, fall back to a Affine distortion when less than 4 control point pairs are provided. While Affine distortions let you use any number of control point pairs, that is Zero pairs is a No-Op (viewport only) distortion, one pair is a translation and two pairs of control points do a scale-rotate-translate, without any shearing.
values: arguments given.
bestfit: Attempt to 'bestfit' the size of the resulting image. This also forces the resulting image to be a 'layered' virtual canvas image. Can be overridden using 'distort:viewport' setting.

Extra Controls from Image meta-data (artifacts)...

- "verbose" Output to stderr alternatives, internal coefficents, and FX equivalents for the distortion operation (if feasible). This forms an extra check of the distortion method, and allows users access to the internal constants IM calculates for the distortion.
- "distort:viewport" Directly set the output image canvas area and offest to use for the resulting image, rather than use the original images canvas, or a calculated 'bestfit' canvas.
- "distort:scale" Scale the size of the output canvas by this amount to provide a method of Zooming, and for super-sampling the results.

Other settings that can effect results include

- 'interpolate' For source image lookups (scale enlargements)
- 'filter' Set filter to use for area-resampling (scale shrinking). Set to 'point' to turn off and use 'interpolate' lookup instead

See also:

- 5.19.53 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ16MBS 692

5.19.55 Edge(radius as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Finds edges in an image.
Notes: Radius defines the radius of the convolution filter. Use a radius of 0 and Edge selects a suitable radius for you.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.56 Emboss(radius as Double, sigma as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a grayscale image with a three-dimensional effect.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, radius should be larger than sigma. Use a radius of 0 and Emboss selects a suitable radius for you.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.57 EncipherImage(passkey as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts pixels to cipher-pixels.
Notes: passkey: encipher pixels with this passphrase.
Returns true on success.

5.19.58 EqualizeImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: Returns true on success or false on failure.

ChannelType: The channels to use.

5.19.59 EqualizeImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: Returns true on success or false on failure.

ChannelType: The channels to use.

Constants for channel:

```
const UndefinedChannel    = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

5.19.60 ExcerptImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a excerpt of the image as defined by the geometry.
Notes: Define the region of the image to extend with x, y, width, and height.

5.19.61 ExtentImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends the image as defined by the geometry, gravity, and image background color.
Notes: Define the region of the image to extend with x, y, width, and height.

Set the (x,y) offset of the geometry to move the original image relative to the extended image.

5.19.62 FlattenImages as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flatten composites all images from the current image pointer to the end of the image list and returns a single flattened image.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.63 Flip as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flip creates a vertical mirror image by reflecting the pixels around the central x-axis.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.64 Flop as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flop creates a horizontal mirror image by reflecting the pixels around the central y-axis.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.65 FrameImage(x as Integer, y as Integer, width as Integer, height as Integer, innerBevel as Integer, OuterBevel as Integer) as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a simulated three-dimensional border around the image.
Notes: The color of the border is defined by the MatteColor of image. Width and height specify the border width of the vertical and horizontal sides of the frame. innerBevel and OuterBevel indicate the width of the inner and outer shadows of the frame.

5.19.66 FxImage(expression as string) as IMImageQ16MBS

Plugin Version: 8.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: FxImage() applies a mathematical expression to the specified image.
Notes: Can raise an exception.

5.19.67 GaussianBlurChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and GaussianBlur selects a suitable radius for you.
Sets the last exception property.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
channel: The channel type.
sigma: the standard deviation of the Gaussian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7ffffff
```

For more details please check the ImageMagick documentation.

5.19.68 GetImageAttribute(key as string) as IMImageAttributeQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetImageAttribute searches the list of image attributes and returns a reference to the attribute if it exists otherwise nil.

5.19.69 GetImageClippingPathAttribute as IMImageAttributeQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetImageClippingPathAttribute searches the list of image attributes and returns a reference to a clipping path if it exists otherwise nil.

5.19.70 GetImageProfile(name as string) as string

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets a profile associated with an image by name.
Notes: Returns ”” on any error.

5.19.71 GetNextImageAttribute as IMImageAttributeQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetNextImageAttribute() gets the next image attribute.
Notes: Returns nil on any error.

5.19.72 GetNextImageProfile as string

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets the next profile name for an image.
Notes: Returns ”” on any error.

5.19.73 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole Image structure copied into a memoryblock.
Notes: Returns nil on any error.

5.19.74 ImagesToBlob(info as IMImageInfoQ16MBS) as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImagesToBlob implements direct to memory image formats.
Notes: It returns the image sequence as a string. The magick member of the ImageInfo structure determines the format of the returned blob (GIF, JPEG, PNG, etc.)

Note, some image formats do not permit multiple images to the same image stream (e.g. JPEG). in this instance, just the first image of the sequence is returned as a blob.

Sets the last exception property and returns "" on any error.
For more details please check the ImageMagick documentation.

5.19.75 ImageToBlob(info as IMImageInfoQ16MBS) as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImagesToBlob implements direct to memory image formats.
Example:

```
dim im as ImageMagickQ16MBS // global

Function IMPictureToString(p as picture, magick as string, quality as Integer) As string
dim image as new IMImageQ16MBS
dim imageinfo as IMImageInfoQ16MBS
dim s,data as string
dim impp as new IMMagickPixelPacketQ16MBS

// empty string for nil picture
if p = nil then
Return ""
end if

// create a new picture info

imageinfo = im.NewImageInfo
imageinfo.ColorSpace=1
// only color space is needed. 1 for RGB.

// background color of image
```

```
impp.red = 0
impp.Green = 0
impp.Blue = 0

// creates a new image object
if not image.NewImage(imageinfo,p.Width,p.Height,impp) then
Return ""
end if

// copy RB picture into IM Image at position 0/0
image.ColorSpace = 1
image.SetPicture(p,0,0)

// set compression data
imageinfo.Magick = magick
imageinfo.Quality = quality

// and rendering intent: 2=PerceptualIntent
'image.RenderingIntent = 2

// create image data
data = image.ImageToBlob(imageinfo)

// release memory
image.DestroyImage
imageinfo.DestroyImageInfo

// return result
Return data

Exception
// in case of an exception return nothing
Return ""


End Function
```

Notes: It returns the image sequence as a string. The magick member of the ImageInfo structure determines the format of the returned blob (GIF, JPEG, PNG, etc.)

Note, some image formats do not permit multiple images to the same image stream (e.g. JPEG). in this instance, just the first image of the sequence is returned as a blob.

Sets the last exception property and returns "" on any error.
For more details please check the ImageMagick documentation.

5.19.76 Implode(factor as Double) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ImplodeImage creates a new image that is a copy of an existing one with the image pixels ”implode” by the specified percentage.
Notes: factor: A double value that defines the extent of the implosion.

Returns nil on any error.
Sets the last exception property.

5.19.77 IsBlobExempt as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is exempt.
Notes: For more details please check the ImageMagick documentation.

5.19.78 IsBlobSeekable as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is seekable.
Notes: For more details please check the ImageMagick documentation.

5.19.79 IsBlobTemporary as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is temporary.
Notes: For more details please check the ImageMagick documentation.

5.19.80 Magnify as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A convenience method that scales an image proportionally to twice its size.
Notes: Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.81 MedianFilter(radius as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a digital filter that improves the quality of a noisy image.
Notes: Each pixel is replaced by the median in a set of neighboring pixels as defined by radius.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.82 MergeImageLayers(ImageLayerMethod as Integer) as IMImageQ16MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: MergeImageLayers() composes all the image layers from the current given image onward to produce a single image of the merged layers.
Notes: The inital canvas's size depends on the given ImageLayerMethod, and is initialized using the first images images background color. The images are then compositied onto that image in sequence using the given composition that has been assigned to each individual image.

ImageLayerMethod:
the method of selecting the size of the initial canvas.

MergeLayer: Merge all layers onto a canvas just large enough to hold all the actual images. The virtual canvas of the first image is preserved but otherwise ignored.

FlattenLayer: Use the virtual canvas size of first image. Images which fall outside this canvas is clipped. This can be used to 'fill out' a given virtual canvas.

MosaicLayer: Start with the virtual canvas of the first image, enlarging left and right edges to contain all images. Images with negative offsets will be clipped.

Can raise an exception.

5.19.83 Minify as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A convenience method that scales an image proportionally to half its size.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.19.84 MosaicImages as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: MosaicImages inlays an image sequence to form a single coherent picture.
Notes: It returns a single image with each image in the sequence composited at the location defined by the page member of the image structure.
Returns nil on any error.
Sets the last exception property.

5.19.85 MotionBlur(radius as Double, sigma as Double, angle as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Simulates motion blur.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and MotionBlur selects a suitable radius for you. Angle gives the angle of the blurring motion.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.86 NegateImage(gray as boolean = false) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: Returns true on success or false on failure.
The grayscale option means that only grayscale values within the image are negated.

gray: If true, only negate grayscale pixels within the image.

5.19.87 NegateImageChannel(ChannelType as Integer, gray as boolean = false) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: Returns true on success or false on failure.
The grayscale option means that only grayscale values within the image are negated.

ChannelType: The channels to use.
gray: If true, only negate grayscale pixels within the image.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.19.88 NewImage(info as IMImageInfoQ16MBS, width as Integer, height as Integer, background as IMMagickPixelPacketQ16MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image.
Example:

```
dim im as ImageMagickQ16MBS // global
dim p as picture
dim imageinfo as IMImageInfoQ16MBS
dim image as IMImageQ16MBS
dim b as new IMMagickPixelPacketQ16MBS
b.Blue=65535
b.ColorSpace=1 // RGB
b.Depth=16
```

```
imageinfo = im.NewImageInfo
imageinfo.Depth=16
imageinfo.ColorSpace=1

//this should read any image IM understands
image = new IMImageQ16MBS
if image.NewImage(imageinfo,500,500,b) then
p=New Picture(300,300,32)
p.Graphics.ForeColor=Rgb(255,0,0)
p.Graphics.FillOval 0,0,300,300
image.SetPicture p,0,0
else
MsgBox "failed"
end if
```

Notes: Returns false on failure and true on success.

5.19.89 NormalizeImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: Returns true on success or false on failure.

5.19.90 NormalizeImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: Returns true on success or false on failure.

ChannelType: The channels to auto-level. If the special 'SyncChannels' flag is set the min/max/mean value of all given channels is used for all given channels, to all channels in the same way.

Constants for channel:

```
const UndefinedChannel    = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

5.19.91 OilPaint(radius as Double) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method OilPaintImage creates a new image that is a copy of an existing one with each pixel component replaced with the color of greatest frequency in a circular neighborhood.
Notes: radius parameter: radius of the circular neighborhood.
Returns nil on any error.
Sets the last exception property.

5.19.92 OptimizeImageLayers as IMImageQ16MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImageLayers() compares each image the GIF disposed forms of the previous image in the sequence.
Notes: From this it attempts to select the smallest cropped image to replace each frame, while preserving the results of the GIF animation.

Can raise an exception.

5.19.93 OptimizeImageTransparency

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImageTransparency() takes a frame optimized GIF animation, and compares the overlayed pixels against the disposal image resulting from all the previous frames in the animation.
Notes: Any pixel that does not change the disposal image (and thus does not effect the outcome of an

overlay) is made transparent.

WARNING: This modifies the current images directly, rather than generate a new image sequence.

Can raise an exception.

5.19.94 OptimizePlusImageLayers as IMImageQ16MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImagePlusLayers() is exactly as OptimizeImageLayers(), but may also add or even remove extra frames in the animation, if it improves the total number of pixels in the resulting GIF animation.
Notes: Can raise an exception.

5.19.95 ProfileImage(name as string, ProfileData as string) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds or removes a ICC, IPTC, or generic profile from an image.
Notes: If the ProfileData is "", it is removed from the image otherwise added. Use a name of '*' and a ProfileData of "" to remove all profiles from the image.

Returns false on any error and true on success.

5.19.96 RadialBlur(angle as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: RadialBlur applies a radial blur to the image.
Notes: angle: The angle of the radial blur.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.19.97 RaiseImage(x as Integer, y as Integer, width as Integer, height as Integer, raise as boolean) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a simulated three-dimensional button-like effect by lightening and darkening the edges of the image.
Notes: Width and height define the width of the vertical and horizontal edge of the effect.
raise: A value other than zero creates a 3-D raise effect, otherwise it has a lowered effect.

5.19.98 RandomThresholdChannel(channel as Integer, thresholds as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Canges the value of individual pixels based on the intensity of each pixel compared to a random threshold.
Notes: The result is a low-contrast, two color image.

channel: The channel or channels to be thresholded.
thresholds: a geometry string containing low,high thresholds. If the string contains 2x2, 3x3, or 4x4, an ordered dither of order 2, 3, or 4 is performed instead. (ASCII string)

Sets the last exception property.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.19.99 ReduceNoise(radius as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Smooths the contours of an image while still preserving edge information.
Notes: The algorithm works by replacing each pixel with its neighbor closest in value. A neighbor is defined by radius. Use a radius of 0 and ReduceNoise selects a suitable radius for you.

For more details please check the ImageMagick documentation.

5.19.100 RemoveDuplicateLayers

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that is exactly the same as the next image in the given image list.
Notes: Image size and virtual canvas offset must also match, though not the virtual canvas size itself.

No check is made with regards to image disposal setting, though it is the dispose setting of later image that is kept. Also any time delays are also added together. As such coalesced image animations should still produce the same result, though with duplicte frames merged into a single frame.

5.19.101 RemoveFirstImageFromList as IMImageQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes the first image from the image list and returns the image.
Notes: Returns nil on any error.
For more details please check the ImageMagick documentation.

5.19.102 RemoveImageProfile(name as string) as string

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes a profile from the image-map by its name.

5.19.103 RemoveZeroDelayLayers

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that as a zero delay time.
Notes: Such images generally represent intermediate or partial updates in GIF animations used for file optimization. They are not ment to be displayed to users of the animation. Viewable images in an animation should have a time delay of 3 or more centi-seconds (hundredths of a second).

However if all the frames have a zero time delay, then either the animation is as yet incomplete, or it is not a GIF animation. This is a non-sensible situation, so no image will be removed and a 'Zero Time Animation' warning (exception) given.

No warning will be given if no image was removed because all images had an appropriate non-zero time delay set.

Due to the special requirements of GIF disposal handling, GIF animations should be coalesced first, before calling this function, though that is not a requirement.

5.19.104 ResetImageAttributeIterator

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ResetImageAttributeIterator() resets the image attributes iterator.
Notes: Use it in conjunction with GetNextImageAttribute() to iterate over all the values associated with an image.

5.19.105 ResetImageProfileIterator

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image profile iterator.
Notes: Use it in conjunction with GetNextImageProfile() to iterate over all the profiles associated with an image.

5.19.106 Resize(width as Integer, height as Integer, FilterID as Integer, blur as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions.

Notes: Constants for the FilterID:

```
const PointFilter       =1
 const BoxFilter        =2
 const TriangleFilter   =3
 const HermiteFilter    =4
 const HanningFilter    =5
 const HammingFilter    =6
 const BlackmanFilter   =7
 const GaussianFilter   =8
 const QuadraticFilter  =9
 const CubicFilter      =10
 const CatromFilter     =11
 const MitchellFilter   =12
 const LanczosFilter    =13
 const BesselFilter     =14
 const SincFilter       =15
```

Most of the filters are FIR (finite impulse response), however, Bessel, Gaussian, and Sinc are IIR (infinite impulse response). Bessel and Sinc are windowed (brought down to zero) with the Blackman filter.
Sets the last exception property.

5.19.107 RGBTransformImage(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method RGBTransformImage converts the reference image from RGB to an alternate colorspace.
Notes: The transformation matrices are not the standard ones: the weights are rescaled to normalized the range of the transformed values to be [0..MaxRGB] .

colorspace: An integer value that indicates which colorspace to transform the image.

Returns false on any error and true on success.

constants:

5.19.108 Roll(x as Integer, y as Integer) as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll offsets an image as defined by x and y.

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

Notes: Returns nil on any error.
Sets the last exception property.

5.19.109 Rotate(degrees as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotation of an image.
Notes: Method RotateImage creates a new image that is a rotated copy of an existing one. Positive angles rotate counter-clockwise (right-hand rule), while negative angles rotate clockwise. Rotated images are usually larger than the originals and have 'empty' triangular corners. X axis. Empty triangles left over from shearing the image are filled with the color specified by the image background_color. RotateImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Method RotateImage is based on the paper "A Fast Algorithm for General Raster Rotatation" by Alan W. Paeth. RotateImage is adapted from a similar method based on the Paeth paper written by Michael Halle of the Spatial Imaging Group, MIT Media Lab.

degrees: Specifies the number of degrees to rotate the image.

Sets the lastexception property.
Returns nil on low memory.
For more details please check the ImageMagick documentation.

5.19.110 Sample(width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions with pixel sampling.
Notes: Unlike other scaling methods, this method does not introduce any additional color into the scaled image.
For more details please check the ImageMagick documentation.
Sets the last exception property.

5.19.111 Scale(width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Example:

```
dim image as IMImageQ16MBS // your image
image=Image.Scale(100,80)
```

Notes: This method was designed by Bob Friesenhahn as a low cost thumbnail generator.

columns: The number of columns in the scaled image.
rows: The number of rows in the scaled image.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.19.112 SetImageAttribute(key as string, value as string) as boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: SetImageAttribute searches the list of image attributes and replaces the attribute value.
Notes: If it is not found in the list, the attribute name and value is added to the list. If the attribute exists in the list, the value is concatenated to the attribute. SetImageAttribute returns True if the attribute is successfully concatenated or added to the list, otherwise False. If the value is "", the matching key is deleted from the list.

5.19.113 SetImageColorspace(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the colorspace member of the Image structure.
Notes: Returns false on any error and true on success.

5.19.114 SetImageProfile(name as string, ProfileData as string) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a named profile to the image.
Notes: If a profile with the same name already exists, it is replaced. This method differs from the ProfileImage() method in that it does not apply CMS color profiles.

name: The profile name.
profiledata: The binary data of the profile.

Returns false on any error and true on success.

5.19.115 SetPicture(pic as picture, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the pixels from a given Xojo picture into the Image Magick Image at the given location.
Example:

```
dim image as IMImageQ16MBS // your image
dim p as picture

p=New Picture(32,32,32)
p.Graphics.ForeColor=rgb(0,255,0)
p.Graphics.FillRect 0,0,32,32

image.SetPicture(p,30,30)
```

Notes: Sets the last exception property.
The method will do nothing on bad bounds.
This method works only for bitmap images.
x and y are zero based.

5.19.116 SetPictureMask(maskpic as picture, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the pixels from a given Xojo picture into the mask of the Image Magick Image at the given location.
Example:

```
dim i as IMImageQ16MBS // your image
dim p as picture

p=New Picture(32,32,32)
p.Graphics.ForeColor=rgb(0,255,0)
p.Graphics.FillRect 0,0,32,32

i.SetPictureMask(p,30,30)
```

Notes: Sets the last exception property.
The method will do nothing on bad bounds.
This method works only for bitmap images.
x and y are zero based.
You may need to set matte=True after this.

5.19.117 SetPixel(x as Integer, y as Integer, newPixel as IMColorQ16MBS)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets a pixel value.
Example:

```
dim image as IMImageQ16MBS // your image
dim co as IMColorQ16MBS

co=new IMColorQ16MBS
co.blue=65535 // max value
image.SetPixel 50,50,co // Makes Pixel 50/50 blue
```

Notes: The method will fail silently if the values are out of bounds or the image is not a bitmap image.
This method works only for bitmap images.
x and y are zero based.

5.19.118 Shade(gray as boolean, azimuth as Double, elevation as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shines a distant light on an image to create a three-dimensional effect.
Notes: You control the positioning of the light with azimuth and elevation; azimuth is measured in degrees off the x axis and elevation is measured in pixels above the Z axis.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.119 SharpenChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, radius should be larger than sigma. Use a radius of 0 and Sharpen selects a suitable radius for you.

channel: The channel type.
radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Laplacian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.120 Shave(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shave shaves pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.
Returns nil on any error.
Sets the last exception property.

5.19.121 Shear(Xshear as Double, Yshear as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ShearImage creates a new image that is a shear_image copy of an existing one.
Notes: Shearing slides one edge of an image along the X or Y axis, creating a parallelogram. An X direction shear slides an edge along the X axis, while a Y direction shear slides an edge along the Y axis. The amount of the shear is controlled by a shear angle. For X direction shears, x_shear is measured relative to the Y axis, and similarly, for Y direction shears y_shear is measured relative to the X axis. Empty triangles left over from shearing the image are filled with the color defined by the pixel at location (0,0). ShearImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Xshear and YYshear specify the number of degrees to shear the image.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.19.122 Solarize(factor as Double) as boolean

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SolarizeImage produces a 'solarization' effect seen when exposing a photographic film to light during the development process.
Notes: factor: An double value that defines the extent of the solarization.
Returns nil on any error.
Sets the last exception property.

5.19.123 Splice(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Splice splices a solid color into the image as defined by the geometry.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.124 Spread(radius as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: This is a special effects method that randomly displaces each pixel in a block defined by the radius parameter.
Notes: radius: Choose a random pixel in a neighborhood of this extent.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.19.125 Stegano(watermarkImage as IMImageQ16MBS) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SteganoImage hides a digital watermark within the image.
Notes: Returns nil on any error.
Sets the last exception property.

5.19.126 Stereo(otherImage as IMImageQ16MBS) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method StereoImage combines two images and produces a single image that is the composite of a left and right image of a stereo pair.
Notes: The left image is converted to gray scale and written to the red channel of the stereo image. The right image is converted to gray scale and written to the blue channel of the stereo image. View the composite image with red-blue glasses to create a stereo effect.

left image = self
right image = otherImage parameter

Returns nil on any error.
Sets the last exception property.

5.19.127 Swirl(degrees as Double) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SwirlImage creates a new image that is a copy of an existing one with the image pixels ”swirl” at a specified angle.
Notes: degrees: An double value that defines the tightness of the swirling.

Returns nil on any error.
Sets the last exception property.

5.19.128 Thumbnail(width as Integer, height as Integer) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Notes: Sets the last exception property.
This method was designed by Bob Friesenhahn as a low cost thumbnail generator.
For more details please check the ImageMagick documentation.

5.19.129 TransformImage(CropGeometry as string, ImageGeometry as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransformImage() is a convenience method that behaves like ResizeImage() or CropImage() but accepts scaling and/or cropping information as a region geometry specification. If the operation fails, the original image handle is left as is.
Notes: This should only be used for single images.

CropGeometry: A crop geometry string. This geometry defines a subregion of the image to crop.
ImageGeometry: An image geometry string. This geometry defines the final size of the image.

Returns true on success.

5.19.130 TransformImages(CropGeometry as string, ImageGeometry as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransformImages() calls TransformImage() on each image of a sequence.
Notes: TransformImage() is a convenience method that behaves like ResizeImage() or CropImage() but accepts scaling and/or cropping information as a region geometry specification. If the operation fails, the original image handle is left as is.

CropGeometry: A crop geometry string. This geometry defines a subregion of the image to crop.
ImageGeometry: An image geometry string. This geometry defines the final size of the image.

Returns true on success.

5.19.131 TransformRGBImage(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method TransformRGBImage converts the reference image from an alternate colorspace.
Notes: The transformation matrices are not the standard ones: the weights are rescaled to normalized the range of the transformed values to be [0..MaxRGB] .

colorspace: An integer value that indicates the colorspace the image is currently in. On return the image is in the RGB color space.

Returns false on any error and true on success.

constants:

5.19.132 TransposeImage as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransposeImage() creates a horizontal mirror image by reflecting the pixels around the central y-axis while rotating them by 90 degrees.

5.19.133 TransverseImage as IMImageQ16MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

Function: TransverseImage() creates a vertical mirror image by reflecting the pixels around the central x-axis while rotating them by 270 degrees.

5.19.134 Trim as IMImageQ16MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Trim trims pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Returns nil on any error.
Sets the last exception property.

5.19.135 UnsharpMaskChannel(channel as Integer, radius as Double, sigma as Double, amount as Double, threshold as Double) as IMImageQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and UnsharpMask selects a suitable radius for you.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.19.136 Wave(amplitude as Double, wavelength as Double) as IMImageQ16MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method Wave creates a new image that is a copy of an existing one with the image pixels altered along a sine wave.
Notes: Parameters are double values that indicates the amplitude and wavelength of the sine wave.
Returns nil on any error.
Sets the last exception property.

5.19.137 WhiteThreshold(threshold as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: WhiteThreshold is like Threshold but forces all pixels above the threshold into white while leaving all pixels below the threshold unchanged.
Notes: No exceptions are generated.
threshold: Define the threshold value. (ASCII string)
For more details please check the ImageMagick documentation.

5.19.138 WriteImage(info as IMImageInfoQ16MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method WriteImage writes an image to a file as defined by image.filename.
Notes: You can specify a particular image format by prefixing the file with the image type and a colon (i.e. ps:image) or specify the image type as the filename suffix (i.e. image.ps). The image may be modified to adapt it to the requirements of the image format. For example, DirectClass images must be color-reduced to PseudoClass if the format is GIF.

WriteImage returns True if the image is written. False is returned if there is a memory shortage or if the image file fails to write.

5.19.139 Properties

5.19.140 BackgroundColor as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Notes: (Read and Write property)

5.19.141 Bias as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.19.142 BlurFactor as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur factor to apply to the image when zooming. Default is 1.0 (no blur).
Notes: (Read and Write property)

5.19.143 BorderColor as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

5.19.144 Colors as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The desired number of colors.
Notes: Used by Quantize().
(Read and Write property)

5.19.145 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

(Read and Write property)

5.19.146 Compression as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.
Notes: useful constants:

```
const UndefinedCompression        = 0
 const NoCompression              = 1
 const BZipCompression            = 2
 const FaxCompression             = 3
 const Group4Compression          = 4
 const JPEGCompression            = 5
 const LosslessJPEGCompression    = 6
 const LZWCompression             = 7
 const RLECompression             = 8
 const ZipCompression             = 9
```

The default is the compression type of the specified image file.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.147 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: QuantumLeap must be defined before a depth of 16 is valid.
(Read and Write property)

5.19.148 Directory as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile names from within an image montage.
Notes: Only valid after calling MontageImages() or reading a MIFF file which contains a directory.
(Read and Write property)

5.19.149 Endian as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian setting to use.
Notes: constants:

UndefinedEndian	0	
LSBEndian	1	(Windows)
MSBEndian	2	(Mac)

e.g. tiff files support different endian settings.
(Read and Write property)

5.19.150 Filename as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path/name.
Notes: The string must be in the encoding of the library and is limited to 4000 bytes.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.151 Filter as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Filter to use when resizing image.
Notes: Constants:

The reduction filter employed has a sipngicant effect on the time required to resize an image and the resulting quality. The default filter is Lanczos which has been shown to produce high quality results when reducing most images.
(Read and Write property)

5.19.152 Fuzz as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this to match colors that are close to the target color in RGB space.
(Read and Write property)

const PointFilter	=1
const BoxFilter	=2
const TriangleFilter	=3
const HermiteFilter	=4
const HanningFilter	=5
const HammingFilter	=6
const BlackmanFilter	=7
const GaussianFilter	=8
const QuadraticFilter	=9
const CubicFilter	=10
const CatromFilter	=11
const MitchellFilter	=12
const LanczosFilter	=13
const BesselFilter	=14
const SincFilter	=15

5.19.153 Gamma as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma level of the image.
Notes: The same color image displayed on two different workstations may look different due to differences in the display monitor. Use gamma correction to adjust for this color difference.
(Read and Write property)

5.19.154 Geometry as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size of the image when encoding.
Notes: (Read and Write property)

5.19.155 Gravity as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.19.156 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to an Image structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.157 Height as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the image in pixels.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.158 Interlace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.

constants:

UndefinedInterlace	0	Unset value.
NoInterlace	1	Don't interlace image (RGBRGBRGBRGBRGBRGB...)
LineInterlace	2	Use scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...)
PlaneInterlace	3	Use plane interlacing (RRRRRR...GGGGGG...BBBBBB...)
PartitionInterlace	4	Similar to plane interlaing except that the different planes are saved to individual files (e.g. image.R, image.G, and image.B)

(Read and Write property)

5.19.159 LastError as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code reported.
Notes: If an exception is raised and it is not a warning exception, this exception code is saved in this property.
(Read and Write property)

5.19.160 LastException as IMExceptionQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception thrown by the Image Magick library.
Notes: You should check this value after every call to the library, process the error and set the property to nil.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.161 Magick as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image encoding format (e.g. "GIF").
Notes: (Read and Write property)

5.19.162 Matte as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether an alpha channel is used/present.
Notes: Set to true to enable masks.
(Read and Write property)

5.19.163 MatteColor as IMColorQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (transparent) color.
Notes: (Read and Write property)

5.19.164 Montage as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile size and offset within an image montage. Only valid for montage images.
Notes: (Read and Write property)

5.19.165 Offset as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of initial bytes to skip over when reading raw image.
Notes: (Read and Write property)

5.19.166 Orientation as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image orientation.
Notes: constants:

```
const UndefinedOrientation       = 0
 const TopLeftOrientation        = 1
 const TopRightOrientation       = 2
 const BottomRightOrientation    = 3
 const BottomLeftOrientation     = 4
 const LeftTopOrientation        = 5
 const RightTopOrientation       = 6
 const RightBottomOrientation    = 7
 const LeftBottomOrientation     = 8
```

For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.167 Quality as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level.
Example:

```
dim im as ImageMagickQ16MBS // global
```

```
Function TestJPEG(f as folderitem) As picture
// Reads an image, compresses in memory to JPEG, decompresses using JPEGlib and returns the image
// if quality setting works, you see it in the result.
// no error checking included!

// needs: im as ImageMagickQ16MBS ready initialized

dim image as IMImageQ16MBS
dim imageinfo as IMImageInfoQ16MBS
dim s,blob as string
dim p as Picture
dim i as Integer

if f = nil then
Return nil
end if

imageinfo = im.NewImageInfo

imageinfo.Filename = f.NativePath

//this should read any image IM understands
image = im.ReadImage(imageinfo)
//check for error
if im.lastexception <>nil and im.LastException.Severity >= 400 then
s = "LastError: "+Format(im.LastError,"-0")+" - Severity: "+str(im.LastException.Severity)+EndOfLine+im.LastException.Reason
MsgBox s
Return nil
elseif image = nil then
MsgBox "image=nil"
Return nil
end if

// Now lets convert to jpeg
imageinfo.Filename = "image.jpg"
imageinfo.Quality = 10 // 100 is max
blob = image.ImageToBlob(imageinfo)

// It may fail
if blob.lenb = 0 then
Return nil
end if
p = JPEGStringToPictureMBS(blob,true)

image.DestroyImage
imageinfo.DestroyImageInfo
```

```
Return p
Exception
Return nil
End Function
```

Notes: Default value is 75.
(Read and Write property)

5.19.168 Release as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: If true, the destructor will release the handle.
Notes: (Read and Write property)

5.19.169 RenderingIntent as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rendering intent to use.
Notes: constants:

UndefinedIntent	0
SaturationIntent	1
PerceptualIntent	2
AbsoluteIntent	3
RelativeIntent	4

(Read and Write property)

5.19.170 ResolutionUnits as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Notes: constants:

(Read and Write property)

UndefinedResolution	0	Unset value.
PixelsPerInchResolution	1	Density specifications are specified in units of pixels per inch (english units).
PixelsPerCentimeterResolution	2	Density specifications are specified in units of pixels per centimeter (metric units).

5.19.171 ResolutionX as Double

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The horizontal resolution of the image.
Notes: The unit for resolution must be specified.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.172 ResolutionY as Double

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The vertical resolution of the image.
Notes: The unit for resolution must be specified.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.173 Scene as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.19.174 StorageClass as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image storage class.
Notes: If DirectClass then the image packets contain valid RGB or CMYK colors. If PseudoClass then the image has a colormap referenced by pixel's index member.

constants:

UndefinedClass	0	Unset value.
DirectClass	1	Image is composed of pixels which represent literal color values.
PseudoClass	2	Image is composed of pixels which specify an index in a color palette.

(Read and Write property)

5.19.175 Taint as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set to True if the image pixels have been modified.
Notes: (Read and Write property)

5.19.176 Width as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the image in pixels.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.19.177 Constants

Constants

Constant	Value	Description
kBackgroundDispose	2	One of the Image layer Dispose Types.
kCoalesceLayer	1	One of the Image layer method constants.
kCompareAnyLayer	2	One of the Image layer method constants.
kCompareClearLayer	3	One of the Image layer method constants.
kCompareOverlayLayer	4	One of the Image layer method constants.
kCompositeLayer	12	One of the Image layer method constants.
kDisposeLayer	5	One of the Image layer method constants.
kFlattenLayer	14	One of the Image layer method constants.
kMergeLayer	13	One of the Image layer method constants.
kMosaicLayer	15	One of the Image layer method constants.
kNoneDispose	1	One of the Image layer Dispose Types.
kOptimizeImageLayer	7	One of the Image layer method constants.
kOptimizeLayer	6	One of the Image layer method constants.
kOptimizePlusLayer	8	One of the Image layer method constants.
kOptimizeTransLayer	9	One of the Image layer method constants.
kPreviousDispose	3	One of the Image layer Dispose Types.
kRemoveDupsLayer	10	One of the Image layer method constants.
kRemoveZeroLayer	11	One of the Image layer method constants.
kUndefinedDispose	0	One of the Image layer Dispose Types.
kUndefinedLayer	0	One of the Image layer method constants.
kUnrecognizedDispose	0	One of the Image layer Dispose Types.

Distortion Effects

Constant	Value	Description
kAffineDistortion	1	
kAffineDistortion	1	
kAffineProjectionDistortion	2	
kAffineProjectionDistortion	2	
kArcDistortion	9	
kArcDistortion	9	
kBarrelDistortion	14	
kBarrelDistortion	14	
kBarrelInverseDistortion	15	
kBarrelInverseDistortion	15	
kBilinearDistortion	6	
kBilinearDistortion	6	
kBilinearForwardDistortion	6	
kBilinearForwardDistortion	6	
kBilinearReverseDistortion	7	
kBilinearReverseDistortion	7	
kCylinder2PlaneDistortion	12	
kCylinder2PlaneDistortion	12	
kDePolarDistortion	11	
kDePolarDistortion	11	
kPerspectiveDistortion	4	
kPerspectiveDistortion	4	
kPerspectiveProjectionDistortion	5	
kPerspectiveProjectionDistortion	5	
kPlane2CylinderDistortion	13	
kPlane2CylinderDistortion	13	
kPolarDistortion	10	
kPolarDistortion	10	
kPolynomialDistortion	8	
kPolynomialDistortion	8	
kResizeDistortion	17	
kResizeDistortion	17	
kScaleRotateTranslateDistortion	3	
kScaleRotateTranslateDistortion	3	
kSentinelDistortion	18	
kSentinelDistortion	18	
kShepardsDistortion	16	
kShepardsDistortion	16	
kUndefinedDistortion	0	
kUndefinedDistortion	0	

Interpolate Modes

Constant	Value	Description
kBarycentricColorInterpolate	1	
kBarycentricColorInterpolate	1	
kBilinearColorInterpolate	7	
kBilinearColorInterpolate	7	
kInverseColorInterpolate	19	
kInverseColorInterpolate	19	
kPolynomialColorInterpolate	8	
kPolynomialColorInterpolate	8	
kShepardsColorInterpolate	16	
kShepardsColorInterpolate	16	
kUndefinedColorInterpolate	0	
kUndefinedColorInterpolate	0	
kVoronoiColorInterpolate	18	
kVoronoiColorInterpolate	18	

5.20 class IMImageQ32MBS

5.20.1 class IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for an Image Magick Image in memory.
Notes: Can exist with or without pixel data.

For more details please check the ImageMagick documentation.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.1pr4

5.20.2 Methods

5.20.3 AdaptiveThreshold(width as Integer, height as Integer, offset as Integer) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: AdaptiveThreshold selects an individual threshold for each pixel based on the range of intensity values in its local neighborhood.
Notes: This allows for thresholding of an image whose global intensity histogram doesn't contain distinctive peaks.
Sets the last exception property.

width: The width of the local neighborhood.
height: The height of the local neighborhood.
offset: The mean offset.

For more details please check the ImageMagick documentation.

5.20.4 AddNoise(NoiseType as Integer) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds random noise to the image.
Notes: Constants

For more details please check the ImageMagick documentation.
Sets the last exception property.

UndefinedNoise	=0
UniformNoise	=1
GaussianNoise	=2
MultiplicativeGaussianNoise	=3
ImpulseNoise	=4
LaplacianNoise	=5
PoissonNoise	=6

5.20.5 AffineTransformImage(matrix as IMImageAffineMatrixQ32MBS) as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transforms an image as dictated by the affine matrix.

5.20.6 AppendImageToList(img as IMImageQ32MBS)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds an image to the image list.
Notes: For more details please check the ImageMagick documentation.

5.20.7 AutoGammaImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoGammaImage extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.
Notes: Returns true on success or false on failure.

5.20.8 AutoGammaImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoGammaImage extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.
Notes: Returns true on success or false on failure.

channelType: The channels to auto-level. If the special 'SyncChannels' flag is set all given channels is adjusted in the same way using the mean average of those channels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

5.20.9 AutoLevelImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoLevelImage adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: Returns true on success or false on failure.

5.20.10 AutoLevelImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoLevelImage adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: Returns true on success or false on failure.

ChannelType: The channels to auto-level. If the special 'SyncChannels' flag is set the min/max/mean value of all given channels is used for all given channels, to all channels in the same way.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

5.20.11 Average as IMImageQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Average() method takes a set of images and averages them together.
Notes: Each image in the set must have the same width and height. Average() returns a single image with each corresponding pixel component of each image averaged. On failure, a nil image is returned and exception describes the reason for the failure.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.12 BilevelChannel(channel as Integer, threshold as Double) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the value of individual pixels based on the intensity of each pixel channel.
Notes: The result is a high-contrast image.

channel: The channel type.
threshold: define the threshold values.

Constants for channel:

For more details please check the ImageMagick documentation.

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.20.13 BlackThreshold(threshold as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: BlackThreshold is like Threshold but forces all pixels below the threshold into black while leaving all pixels above the threshold unchanged.
Notes: No exceptions are generated.
threshold: Define the threshold value. (ASCII string)
For more details please check the ImageMagick documentation.

5.20.14 BlobSize as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The expected size for this image written to a file.
Notes: For more details please check the ImageMagick documentation.

5.20.15 Blur(radius as Double, sigma as Double) as IMImageQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and BlurImage selects a suitable radius for you.

radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Gaussian, in pixels.

For more details please check the ImageMagick documentation.

5.20.16 BlurImageChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and BlurImageChannel selects a suitable radius for you.

channel: The channel type.
radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Gaussian, in pixels.

Constants for channel:

```
 const UndefinedChannel   = 0
  const RedChannel         = &h0001
  const GrayChannel        = &h0001
  const CyanChannel        = &h0001
  const GreenChannel       = &h0002
  const MagentaChannel     = &h0002
  const BlueChannel        = &h0004
  const YellowChannel      = &h0004
  const AlphaChannel       = &h0008
  const OpacityChannel     = &h0008
  const BlackChannel       = &h0020
  const IndexChannel       = &h0020
  const AllChannels        = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.20.17 BorderImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Surrounds the image with a border of the color defined by the bordercolor member of the image.
Notes: The width and height of the border are defined by the corresponding parameters.

5.20.18 BrightnessContrastImage(brightness as Double, contrast as Double) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image. It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.
Notes: Returns true on success or false on failure.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).

5.20.19 BrightnessContrastImageChannel(ChannelType as Integer, brightness as Double, contrast as Double) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image. It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.
Notes: Returns true on success or false on failure.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).
ChannelType: The channels to use.

Constants for channel:

5.20.20 Charcoal(radius as Double, sigma as Double) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Charcoal creates a new image that is a copy of an existing one with the edge highlighted.
Notes: radius: the radius of the pixel neighborhood.
sigma: The standard deviation of the Gaussian, in pixels.

Returns nil on any error.
Sets the last exception property.

```
const UndefinedChannel    = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

5.20.21 Chop(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chop removes a region of an image and collapses the image to occupy the removed portion.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.22 ClipPath(path as string, inside as boolean) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the image clip mask based any clipping path information if it exists.
Notes:

pathname: name of clipping path resource. If name is preceded by #, use clipping path numbered by name.

inside: if true, later operations take effect inside clipping path. Otherwise later operations take effect outside clipping path.

Returns true on success and false on any error.

5.20.23 Clone as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of this image object.
Notes: For more details please check the ImageMagick documentation.

5.20.24 CloneImageAttributes(image as IMImageAttributeQ32MBS) as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: CloneImageAttributes() clones one or more image attributes.
Notes: Returns false on any error.

5.20.25 CloneImageProfiles(SourceImage as IMImageQ32MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones one or more image profiles.
Notes: Returns false on any error and true on success.

5.20.26 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.20.27 ClutImage(clutImage as IMImageQ32MBS) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either

for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

ClutImage: the color lookup table image for replacement color values.

Returns true on success or false on failure.

5.20.28 ClutImageChannel(ChannelType as Integer, clutImage as IMImageQ32MBS) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

ClutImage: the color lookup table image for replacement color values.
ChannelType: The channels to use.

Returns true on success or false on failure.

Constants for channel:

5.20.29 CoalesceImages as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: CoalesceImages composites a set of images while respecting any page offsets and disposal meth-

```
const UndefinedChannel    = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

ods.
Notes: GIF, MIFF, and MNG animation sequences typically start with an image background and each subsequent image varies in size and offset. CoalesceImages() returns a new sequence where each image in the sequence is the same size as the first and composited with the next image in the sequence.

Returns nil on any error.
Sets the last exception property.

5.20.30 Colorize(opacity as string, PenColorRed as Integer, PenColorGreen as Integer, PenColorBlue as Integer, PenColorOpacity as Integer) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ColorizeImage creates a new image that is a copy of an existing one with the image pixels colorized.
Notes: The colorization is controlled with the pen color and the opacity levels.

opacity: A character string indicating the level of opacity as a percentage (0-100).
PenColorRed, PenColorGreen, PenColorBlue and PenColorOpacity define the pen color used.

Returns nil on any error.
Sets the last exception property.

5.20.31 Combine(channel as Integer) as IMImageQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Combines one or more images into a single image.
Notes: The grayscale value of the pixels of each image in the sequence is assigned in order to the specified channels of the combined image. The typical ordering would be image 1 =>Red, 2 =>Green, 3 =>Blue, etc.

The lastexception property is set.

5.20.32 CompareImageLayers(ImageLayerMethod as Integer) as IMImageQ32MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: CompareImageLayers() compares each image with the next in a sequence and returns the minimum bounding region of all the pixel differences (of the mageLayerMethod specified) it discovers.
Notes: Images do NOT have to be the same size, though it is best that all the images are 'coalesced' (images are all the same size, on a flattened canvas, so as to represent exactly how an specific frame should look).

No GIF dispose methods are applied, so GIF animations must be coalesced before applying this image operator to find differences to them.

ImageLayerMethod:
the layers type to compare images with. Must be one of... CompareAnyLayer, CompareClearLayer, CompareOverlayLayer.

Can raise an exception.

5.20.33 Composite(ComposeOperator as Integer, Image as IMImageQ32MBS, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the second image composited onto the first at the specified offsets.
Notes: compose: Specifies an image composite operator.
Image: The second image.
x: An integer that specifies the column offset of the composited image.
y: An integer that specifies the row offset of the composited image.

No error code and exception!

5.20.34 ConsolidateCMYKImages as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Consolidates a sequence of CMYK images.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.35 ContrastImage(sharpen as boolean) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the intensity differences between the lighter and darker elements of the image.
Notes: Returns true on success or false on failure.
Set sharpen to true to increase the image contrast otherwise the contrast is reduced.

5.20.36 CopyPicture as picture

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ32MBS // your image
Canvas1.Backdrop=image.CopyPicture
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
See also:

- 5.20.37 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture 751

5.20.37 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a portion of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ32MBS // your image
Canvas1.Backdrop=image.CopyPicture(0,0,image.Width,image.Height)
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.
See also:

- 5.20.36 CopyPicture as picture 751

5.20.38 CopyPictureMask as picture

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the mask of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ32MBS // your image
Canvas1.Backdrop=image.CopyPictureMask
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
See also:

- 5.20.39 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture 752

5.20.39 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a portion of the mask of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ32MBS // your image
Canvas1.Backdrop=image.CopyPictureMask(0,0,image.Width,image.Height)
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.
See also:

- 5.20.38 CopyPictureMask as picture 752

5.20.40 CopyPixel(x as Integer, y as Integer) as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a pixel.
Notes: Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.

5.20.41 CreateHBITMAP as Ptr

Plugin Version: 15.1, Platform: Windows, Targets: All.

Function: Creates a HBITMAP for the image for use with Windows Declares.
Notes: The HBITMAP returned needs to be freed when you are done with it or you risk having a memory leak.

5.20.42 Crop(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crop extracts a region of the image starting at the offset defined by geometry.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.43 CropImageToTiles(CropGeometry as string) as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crops a single image, into a possible list of tiles.
Notes: This may include a single sub-region of the image. This basically applies all the normal geometry flags for Crop.

5.20.44 CycleColormap(displace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Displaces an image's colormap by a given number of positions.
Notes: If you cycle the colormap a number of times you can produce a psychodelic effect.
Returns true on success.
displace: displace the colormap this amount.

5.20.45 DecipherImage(passkey as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts cipher pixels to plain pixels.
Notes: Passkey: decipher cipher pixels with this passphrase.
Returns true on success.

5.20.46 DeconstructImages as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: DeconstructImages() compares each image with the next in a sequence and returns the minimum bounding region of all differences from the first image.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.47 DeleteImageAttribute(key as string) as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: DeleteImageAttribute() deletes an attribute from the image.
Notes: Returns false on any error.

5.20.48 Despeckle() as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduces the speckle noise in an image while perserving the edges of the original image.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.20.49 DestroyImage

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases the memory used for this image and sets handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.20.50 DestroyImageAttributes

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Deallocates memory associated with the image attribute list.

5.20.51 DestroyImageList

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys the image list and sets the handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.20.52 DestroyImageProfiles

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases memory associated with an image profile map.

5.20.53 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ32MBS

Plugin Version: 12.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: DistortImage() distorts an image using various distortion methods, by mapping color lookups of the source image to a new destination image usally of the same size as the source image, unless 'bestfit' is set to true.
Notes: If 'bestfit' is enabled, and distortion allows it, the destination image is adjusted to ensure the whole source 'image' will just fit within the final destination image, which will be sized and offset accordingly. Also in many cases the virtual offset of the source image will be taken into account in the mapping.

If the ’-verbose’ control option has been set print to standard error the equicelent ’-fx’ formula with coefficients for the function, if practical.

A description of each parameter follows:
self: the image to be distorted.
m: the method of image distortion. ArcDistortion always ignores source image offset, and always ’bestfit’ the destination image with the top left corner offset relative to the polar mapping center. Affine, Perspective, and Bilinear, do least squares fitting of the distrotion when more than the minimum number of control point pairs are provided. Perspective, and Bilinear, fall back to a Affine distortion when less than 4 control point pairs are provided. While Affine distortions let you use any number of control point pairs, that is Zero pairs is a No-Op (viewport only) distortion, one pair is a translation and two pairs of control points do a scale-rotate-translate, without any shearing.
values: arguments given.
bestfit: Attempt to ’bestfit’ the size of the resulting image. This also forces the resulting image to be a ’layered’ virtual canvas image. Can be overridden using ’distort:viewport’ setting.

Extra Controls from Image meta-data (artifacts)...

- ”verbose” Output to stderr alternatives, internal coefficents, and FX equivalents for the distortion operation (if feasible). This forms an extra check of the distortion method, and allows users access to the internal constants IM calculates for the distortion.
- ”distort:viewport” Directly set the output image canvas area and offest to use for the resulting image, rather than use the original images canvas, or a calculated ’bestfit’ canvas.
- ”distort:scale” Scale the size of the output canvas by this amount to provide a method of Zooming, and for super-sampling the results.

Other settings that can effect results include

- ’interpolate’ For source image lookups (scale enlargements)
- ’filter’ Set filter to use for area-resampling (scale shrinking). Set to ’point’ to turn off and use ’interpolate’ lookup instead

See also:

- 5.20.54 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ32MBS 756

5.20.54 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ32MBS

Plugin Version: 12.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: DistortImage() distorts an image using various distortion methods, by mapping color lookups of the source image to a new destination image usally of the same size as the source image, unless 'bestfit' is set to true.
Notes: If 'bestfit' is enabled, and distortion allows it, the destination image is adjusted to ensure the whole source 'image' will just fit within the final destination image, which will be sized and offset accordingly. Also in many cases the virtual offset of the source image will be taken into account in the mapping.

If the '-verbose' control option has been set print to standard error the equicelent '-fx' formula with coefficients for the function, if practical.

A description of each parameter follows:
self: the image to be distorted.
m: the method of image distortion. ArcDistortion always ignores source image offset, and always 'bestfit' the destination image with the top left corner offset relative to the polar mapping center. Affine, Perspective, and Bilinear, do least squares fitting of the distrotion when more than the minimum number of control point pairs are provided. Perspective, and Bilinear, fall back to a Affine distortion when less than 4 control point pairs are provided. While Affine distortions let you use any number of control point pairs, that is Zero pairs is a No-Op (viewport only) distortion, one pair is a translation and two pairs of control points do a scale-rotate-translate, without any shearing.
values: arguments given.
bestfit: Attempt to 'bestfit' the size of the resulting image. This also forces the resulting image to be a 'layered' virtual canvas image. Can be overridden using 'distort:viewport' setting.

Extra Controls from Image meta-data (artifacts)...

- "verbose" Output to stderr alternatives, internal coefficents, and FX equivalents for the distortion operation (if feasible). This forms an extra check of the distortion method, and allows users access to the internal constants IM calculates for the distortion.
- "distort:viewport" Directly set the output image canvas area and offest to use for the resulting image, rather than use the original images canvas, or a calculated 'bestfit' canvas.
- "distort:scale" Scale the size of the output canvas by this amount to provide a method of Zooming, and for super-sampling the results.

Other settings that can effect results include

- 'interpolate' For source image lookups (scale enlargements)
- 'filter' Set filter to use for area-resampling (scale shrinking). Set to 'point' to turn off and use 'interpolate' lookup instead

See also:

- 5.20.53 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ32MBS 755

5.20.55 Edge(radius as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Finds edges in an image.
Notes: Radius defines the radius of the convolution filter. Use a radius of 0 and Edge selects a suitable radius for you.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.56 Emboss(radius as Double, sigma as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a grayscale image with a three-dimensional effect.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, radius should be larger than sigma. Use a radius of 0 and Emboss selects a suitable radius for you.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.57 EncipherImage(passkey as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts pixels to cipher-pixels.
Notes: passkey: encipher pixels with this passphrase.
Returns true on success.

5.20.58 EqualizeImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: Returns true on success or false on failure.

ChannelType: The channels to use.

5.20.59 EqualizeImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: Returns true on success or false on failure.

ChannelType: The channels to use.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.20.60 ExcerptImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a excerpt of the image as defined by the geometry.
Notes: Define the region of the image to extend with x, y, width, and height.

5.20.61 ExtentImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends the image as defined by the geometry, gravity, and image background color.
Notes: Define the region of the image to extend with x, y, width, and height.

Set the (x,y) offset of the geometry to move the original image relative to the extended image.

5.20.62 FlattenImages as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flatten composites all images from the current image pointer to the end of the image list and returns a single flattened image.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.63 Flip as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flip creates a vertical mirror image by reflecting the pixels around the central x-axis.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.64 Flop as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flop creates a horizontal mirror image by reflecting the pixels around the central y-axis.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.65 FrameImage(x as Integer, y as Integer, width as Integer, height as Integer, innerBevel as Integer, OuterBevel as Integer) as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a simulated three-dimensional border around the image.
Notes: The color of the border is defined by the MatteColor of image. Width and height specify the border width of the vertical and horizontal sides of the frame. innerBevel and OuterBevel indicate the width of the inner and outer shadows of the frame.

5.20.66 FxImage(expression as string) as IMImageQ32MBS

Plugin Version: 8.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: FxImage() applies a mathematical expression to the specified image.
Notes: Can raise an exception.

5.20.67 GaussianBlurChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and GaussianBlur selects a suitable radius for you.
Sets the last exception property.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
channel: The channel type.
sigma: the standard deviation of the Gaussian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.20.68 GetImageAttribute(key as string) as IMImageAttributeQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetImageAttribute searches the list of image attributes and returns a reference to the attribute if it exists otherwise nil.

5.20.69 GetImageClippingPathAttribute as IMImageAttributeQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetImageClippingPathAttribute searches the list of image attributes and returns a reference to a clipping path if it exists otherwise nil.

5.20.70 GetImageProfile(name as string) as string

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets a profile associated with an image by name.
Notes: Returns ”” on any error.

5.20.71 GetNextImageAttribute as IMImageAttributeQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetNextImageAttribute() gets the next image attribute.
Notes: Returns nil on any error.

5.20.72 GetNextImageProfile as string

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets the next profile name for an image.
Notes: Returns ”” on any error.

5.20.73 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole Image structure copied into a memoryblock.
Notes: Returns nil on any error.

5.20.74 ImagesToBlob(info as IMImageInfoQ32MBS) as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImagesToBlob implements direct to memory image formats.
Notes: It returns the image sequence as a string. The magick member of the ImageInfo structure determines the format of the returned blob (GIF, JPEG, PNG, etc.)

Note, some image formats do not permit multiple images to the same image stream (e.g. JPEG). in this instance, just the first image of the sequence is returned as a blob.

Sets the last exception property and returns "" on any error.
For more details please check the ImageMagick documentation.

5.20.75 ImageToBlob(info as IMImageInfoQ32MBS) as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImagesToBlob implements direct to memory image formats.
Example:

```
dim im as ImageMagickQ32MBS // global

Function IMPictureToString(p as picture, magick as string, quality as Integer) As string
dim image as new IMImageQ32MBS
dim imageinfo as IMImageInfoQ32MBS
dim s,data as string
dim impp as new IMMagickPixelPacketQ32MBS

// empty string for nil picture
if p = nil then
Return ""
end if

// create a new picture info

imageinfo = im.NewImageInfo
imageinfo.ColorSpace=1
// only color space is needed. 1 for RGB.

// background color of image
```

```
impp.red = 0
impp.Green = 0
impp.Blue = 0

// creates a new image object
if not image.NewImage(imageinfo,p.Width,p.Height,impp) then
Return ""
end if

// copy RB picture into IM Image at position 0/0
image.ColorSpace = 1
image.SetPicture(p,0,0)

// set compression data
imageinfo.Magick = magick
imageinfo.Quality = quality

// and rendering intent: 2=PerceptualIntent
'image.RenderingIntent = 2

// create image data
data = image.ImageToBlob(imageinfo)

// release memory
image.DestroyImage
imageinfo.DestroyImageInfo

// return result
Return data

Exception
// in case of an exception return nothing
Return ""


End Function
```

Notes: It returns the image sequence as a string. The magick member of the ImageInfo structure determines the format of the returned blob (GIF, JPEG, PNG, etc.)

Note, some image formats do not permit multiple images to the same image stream (e.g. JPEG). in this instance, just the first image of the sequence is returned as a blob.

Sets the last exception property and returns "" on any error.
For more details please check the ImageMagick documentation.

5.20.76 Implode(factor as Double) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ImplodeImage creates a new image that is a copy of an existing one with the image pixels ”implode” by the specified percentage.
Notes: factor: A double value that defines the extent of the implosion.

Returns nil on any error.
Sets the last exception property.

5.20.77 IsBlobExempt as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is exempt.
Notes: For more details please check the ImageMagick documentation.

5.20.78 IsBlobSeekable as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is seekable.
Notes: For more details please check the ImageMagick documentation.

5.20.79 IsBlobTemporary as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is temporary.
Notes: For more details please check the ImageMagick documentation.

5.20.80 Magnify as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A convenience method that scales an image proportionally to twice its size.
Notes: Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.81 MedianFilter(radius as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a digital filter that improves the quality of a noisy image.
Notes: Each pixel is replaced by the median in a set of neighboring pixels as defined by radius.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.82 MergeImageLayers(ImageLayerMethod as Integer) as IMImageQ32MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: MergeImageLayers() composes all the image layers from the current given image onward to produce a single image of the merged layers.
Notes: The inital canvas's size depends on the given ImageLayerMethod, and is initialized using the first images images background color. The images are then compositied onto that image in sequence using the given composition that has been assigned to each individual image.

ImageLayerMethod:
the method of selecting the size of the initial canvas.

MergeLayer: Merge all layers onto a canvas just large enough to hold all the actual images. The virtual canvas of the first image is preserved but otherwise ignored.

FlattenLayer: Use the virtual canvas size of first image. Images which fall outside this canvas is clipped. This can be used to 'fill out' a given virtual canvas.

MosaicLayer: Start with the virtual canvas of the first image, enlarging left and right edges to contain all images. Images with negative offsets will be clipped.

Can raise an exception.

5.20.83 Minify as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A convenience method that scales an image proportionally to half its size.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.20.84 MosaicImages as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: MosaicImages inlays an image sequence to form a single coherent picture.
Notes: It returns a single image with each image in the sequence composited at the location defined by the page member of the image structure.
Returns nil on any error.
Sets the last exception property.

5.20.85 MotionBlur(radius as Double, sigma as Double, angle as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Simulates motion blur.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and MotionBlur selects a suitable radius for you. Angle gives the angle of the blurring motion.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.86 NegateImage(gray as boolean = false) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: Returns true on success or false on failure.
The grayscale option means that only grayscale values within the image are negated.

gray: If true, only negate grayscale pixels within the image.

5.20.87 NegateImageChannel(ChannelType as Integer, gray as boolean = false) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: Returns true on success or false on failure.
The grayscale option means that only grayscale values within the image are negated.

ChannelType: The channels to use.
gray: If true, only negate grayscale pixels within the image.

Constants for channel:

```
 const UndefinedChannel    = 0
  const RedChannel         = &h0001
  const GrayChannel        = &h0001
  const CyanChannel        = &h0001
  const GreenChannel       = &h0002
  const MagentaChannel     = &h0002
  const BlueChannel        = &h0004
  const YellowChannel      = &h0004
  const AlphaChannel       = &h0008
  const OpacityChannel     = &h0008
  const BlackChannel       = &h0020
  const IndexChannel       = &h0020
  const AllChannels        = &h7fffffff
```

5.20.88 NewImage(info as IMImageInfoQ32MBS, width as Integer, height as Integer, background as IMMagickPixelPacketQ32MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image.
Example:

```
dim im as ImageMagickQ32MBS // global
dim p as picture
dim imageinfo as IMImageInfoQ32MBS
dim image as IMImageQ32MBS
dim b as new IMMagickPixelPacketQ32MBS
b.Blue=65535
b.ColorSpace=1 // RGB
b.Depth=16
```

```
imageinfo = im.NewImageInfo
imageinfo.Depth=16
imageinfo.ColorSpace=1

//this should read any image IM understands
image = new IMImageQ32MBS
if image.NewImage(imageinfo,500,500,b) then
p=New Picture(300,300,32)
p.Graphics.ForeColor=Rgb(255,0,0)
p.Graphics.FillOval 0,0,300,300
image.SetPicture p,0,0
else
MsgBox "failed"
end if
```

Notes: Returns false on failure and true on success.

5.20.89 NormalizeImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: Returns true on success or false on failure.

5.20.90 NormalizeImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: Returns true on success or false on failure.

ChannelType: The channels to auto-level. If the special 'SyncChannels' flag is set the min/max/mean value of all given channels is used for all given channels, to all channels in the same way.

Constants for channel:

```
const UndefinedChannel  = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.20.91 OilPaint(radius as Double) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method OilPaintImage creates a new image that is a copy of an existing one with each pixel component replaced with the color of greatest frequency in a circular neighborhood.
Notes: radius parameter: radius of the circular neighborhood.
Returns nil on any error.
Sets the last exception property.

5.20.92 OptimizeImageLayers as IMImageQ32MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImageLayers() compares each image the GIF disposed forms of the previous image in the sequence.
Notes: From this it attempts to select the smallest cropped image to replace each frame, while preserving the results of the GIF animation.

Can raise an exception.

5.20.93 OptimizeImageTransparency

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImageTransparency() takes a frame optimized GIF animation, and compares the overlayed pixels against the disposal image resulting from all the previous frames in the animation.
Notes: Any pixel that does not change the disposal image (and thus does not effect the outcome of an

overlay) is made transparent.

WARNING: This modifies the current images directly, rather than generate a new image sequence.

Can raise an exception.

5.20.94 OptimizePlusImageLayers as IMImageQ32MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImagePlusLayers() is exactly as OptimizeImageLayers(), but may also add or even remove extra frames in the animation, if it improves the total number of pixels in the resulting GIF animation.
Notes: Can raise an exception.

5.20.95 ProfileImage(name as string, ProfileData as string) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds or removes a ICC, IPTC, or generic profile from an image.
Notes: If the ProfileData is "", it is removed from the image otherwise added. Use a name of '*' and a ProfileData of "" to remove all profiles from the image.

Returns false on any error and true on success.

5.20.96 RadialBlur(angle as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: RadialBlur applies a radial blur to the image.
Notes: angle: The angle of the radial blur.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.20.97 RaiseImage(x as Integer, y as Integer, width as Integer, height as Integer, raise as boolean) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a simulated three-dimensional button-like effect by lightening and darkening the edges of the image.
Notes: Width and height define the width of the vertical and horizontal edge of the effect.
raise: A value other than zero creates a 3-D raise effect, otherwise it has a lowered effect.

5.20.98 RandomThresholdChannel(channel as Integer, thresholds as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Canges the value of individual pixels based on the intensity of each pixel compared to a random threshold.
Notes: The result is a low-contrast, two color image.

channel: The channel or channels to be thresholded.
thresholds: a geometry string containing low,high thresholds. If the string contains 2x2, 3x3, or 4x4, an ordered dither of order 2, 3, or 4 is performed instead. (ASCII string)

Sets the last exception property.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.20.99 ReduceNoise(radius as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Smooths the contours of an image while still preserving edge information.
Notes: The algorithm works by replacing each pixel with its neighbor closest in value. A neighbor is defined by radius. Use a radius of 0 and ReduceNoise selects a suitable radius for you.

For more details please check the ImageMagick documentation.

5.20.100 RemoveDuplicateLayers

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that is exactly the same as the next image in the given image list.
Notes: Image size and virtual canvas offset must also match, though not the virtual canvas size itself.

No check is made with regards to image disposal setting, though it is the dispose setting of later image that is kept. Also any time delays are also added together. As such coalesced image animations should still produce the same result, though with duplicte frames merged into a single frame.

5.20.101 RemoveFirstImageFromList as IMImageQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes the first image from the image list and returns the image.
Notes: Returns nil on any error.
For more details please check the ImageMagick documentation.

5.20.102 RemoveImageProfile(name as string) as string

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes a profile from the image-map by its name.

5.20.103 RemoveZeroDelayLayers

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that as a zero delay time.
Notes: Such images generally represent intermediate or partial updates in GIF animations used for file optimization. They are not ment to be displayed to users of the animation. Viewable images in an animation should have a time delay of 3 or more centi-seconds (hundredths of a second).

However if all the frames have a zero time delay, then either the animation is as yet incomplete, or it is not a GIF animation. This is a non-sensible situation, so no image will be removed and a 'Zero Time Animation' warning (exception) given.

No warning will be given if no image was removed because all images had an appropriate non-zero time delay set.

Due to the special requirements of GIF disposal handling, GIF animations should be coalesced first, before calling this function, though that is not a requirement.

5.20.104 ResetImageAttributeIterator

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ResetImageAttributeIterator() resets the image attributes iterator.
Notes: Use it in conjunction with GetNextImageAttribute() to iterate over all the values associated with an image.

5.20.105 ResetImageProfileIterator

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image profile iterator.
Notes: Use it in conjunction with GetNextImageProfile() to iterate over all the profiles associated with an image.

5.20.106 Resize(width as Integer, height as Integer, FilterID as Integer, blur as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions.

Notes: Constants for the FilterID:

```
const PointFilter       =1
 const BoxFilter        =2
 const TriangleFilter   =3
 const HermiteFilter    =4
 const HanningFilter    =5
 const HammingFilter    =6
 const BlackmanFilter   =7
 const GaussianFilter   =8
 const QuadraticFilter  =9
 const CubicFilter      =10
 const CatromFilter     =11
 const MitchellFilter   =12
 const LanczosFilter    =13
 const BesselFilter     =14
 const SincFilter       =15
```

Most of the filters are FIR (finite impulse response), however, Bessel, Gaussian, and Sinc are IIR (infinite impulse response). Bessel and Sinc are windowed (brought down to zero) with the Blackman filter.
Sets the last exception property.

5.20.107 RGBTransformImage(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method RGBTransformImage converts the reference image from RGB to an alternate colorspace.
Notes: The transformation matrices are not the standard ones: the weights are rescaled to normalized the range of the transformed values to be [0..MaxRGB] .

colorspace: An integer value that indicates which colorspace to transform the image.

Returns false on any error and true on success.

constants:

5.20.108 Roll(x as Integer, y as Integer) as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll offsets an image as defined by x and y.

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

Notes: Returns nil on any error.
Sets the last exception property.

5.20.109 Rotate(degrees as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotation of an image.
Notes: Method RotateImage creates a new image that is a rotated copy of an existing one. Positive angles rotate counter-clockwise (right-hand rule), while negative angles rotate clockwise. Rotated images are usually larger than the originals and have 'empty' triangular corners. X axis. Empty triangles left over from shearing the image are filled with the color specified by the image background_color. RotateImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Method RotateImage is based on the paper "A Fast Algorithm for General Raster Rotatation" by Alan W. Paeth. RotateImage is adapted from a similar method based on the Paeth paper written by Michael Halle of the Spatial Imaging Group, MIT Media Lab.

degrees: Specifies the number of degrees to rotate the image.

Sets the lastexception property.
Returns nil on low memory.
For more details please check the ImageMagick documentation.

5.20.110 Sample(width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions with pixel sampling.
Notes: Unlike other scaling methods, this method does not introduce any additional color into the scaled image.
For more details please check the ImageMagick documentation.
Sets the last exception property.

5.20.111 Scale(width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Example:

```
dim image as IMImageQ32MBS // your image
image=Image.Scale(100,80)
```

Notes: This method was designed by Bob Friesenhahn as a low cost thumbnail generator.

columns: The number of columns in the scaled image.
rows: The number of rows in the scaled image.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.20.112 SetImageAttribute(key as string, value as string) as boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: SetImageAttribute searches the list of image attributes and replaces the attribute value.
Notes: If it is not found in the list, the attribute name and value is added to the list. If the attribute exists in the list, the value is concatenated to the attribute. SetImageAttribute returns True if the attribute is successfully concatenated or added to the list, otherwise False. If the value is "", the matching key is deleted from the list.

5.20.113 SetImageColorspace(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the colorspace member of the Image structure.
Notes: Returns false on any error and true on success.

5.20.114 SetImageProfile(name as string, ProfileData as string) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a named profile to the image.
Notes: If a profile with the same name already exists, it is replaced. This method differs from the ProfileImage() method in that it does not apply CMS color profiles.

name: The profile name.
profiledata: The binary data of the profile.

Returns false on any error and true on success.

5.20.115 SetPicture(pic as picture, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the pixels from a given Xojo picture into the Image Magick Image at the given location.
Example:

```
dim image as IMImageQ32MBS // your image
dim p as picture

p=New Picture(32,32,32)
p.Graphics.ForeColor=rgb(0,255,0)
p.Graphics.FillRect 0,0,32,32

image.SetPicture(p,30,30)
```

Notes: Sets the last exception property.
The method will do nothing on bad bounds.
This method works only for bitmap images.
x and y are zero based.

5.20.116 SetPictureMask(maskpic as picture, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the pixels from a given Xojo picture into the mask of the Image Magick Image at the given location.
Example:

```
dim i as IMImageQ32MBS // your image
dim p as picture

p=New Picture(32,32,32)
p.Graphics.ForeColor=rgb(0,255,0)
p.Graphics.FillRect 0,0,32,32

i.SetPictureMask(p,30,30)
```

Notes: Sets the last exception property.
The method will do nothing on bad bounds.
This method works only for bitmap images.
x and y are zero based.
You may need to set matte=True after this.

5.20.117 SetPixel(x as Integer, y as Integer, newPixel as IMColorQ32MBS)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets a pixel value.
Example:

```
dim image as IMImageQ32MBS // your image
dim co as IMColorQ32MBS

co=new IMColorQ32MBS
co.blue=65535 // max value
image.SetPixel 50,50,co // Makes Pixel 50/50 blue
```

Notes: The method will fail silently if the values are out of bounds or the image is not a bitmap image.
This method works only for bitmap images.
x and y are zero based.

5.20.118 Shade(gray as boolean, azimuth as Double, elevation as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shines a distant light on an image to create a three-dimensional effect.
Notes: You control the positioning of the light with azimuth and elevation; azimuth is measured in degrees off the x axis and elevation is measured in pixels above the Z axis.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.119 SharpenChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, radius should be larger than sigma. Use a radius of 0 and Sharpen selects a suitable radius for you.

channel: The channel type.
radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Laplacian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.120 Shave(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shave shaves pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.
Returns nil on any error.
Sets the last exception property.

5.20.121 Shear(Xshear as Double, Yshear as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ShearImage creates a new image that is a shear_image copy of an existing one.
Notes: Shearing slides one edge of an image along the X or Y axis, creating a parallelogram. An X direction shear slides an edge along the X axis, while a Y direction shear slides an edge along the Y axis. The amount of the shear is controlled by a shear angle. For X direction shears, x_shear is measured relative to the Y axis, and similarly, for Y direction shears y_shear is measured relative to the X axis. Empty triangles left over from shearing the image are filled with the color defined by the pixel at location (0,0). ShearImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Xshear and YYshear specify the number of degrees to shear the image.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.20.122 Solarize(factor as Double) as boolean

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SolarizeImage produces a 'solarization' effect seen when exposing a photographic film to light during the development process.
Notes: factor: An double value that defines the extent of the solarization.
Returns nil on any error.
Sets the last exception property.

5.20.123 Splice(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Splice splices a solid color into the image as defined by the geometry.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.124 Spread(radius as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: This is a special effects method that randomly displaces each pixel in a block defined by the radius parameter.
Notes: radius: Choose a random pixel in a neighborhood of this extent.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.20.125 Stegano(watermarkImage as IMImageQ32MBS) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SteganoImage hides a digital watermark within the image.
Notes: Returns nil on any error.
Sets the last exception property.

5.20.126 Stereo(otherImage as IMImageQ32MBS) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method StereoImage combines two images and produces a single image that is the composite of a left and right image of a stereo pair.
Notes: The left image is converted to gray scale and written to the red channel of the stereo image. The right image is converted to gray scale and written to the blue channel of the stereo image. View the composite image with red-blue glasses to create a stereo effect.

left image = self
right image = otherImage parameter

Returns nil on any error.
Sets the last exception property.

5.20.127 Swirl(degrees as Double) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SwirlImage creates a new image that is a copy of an existing one with the image pixels ”swirl” at a specified angle.
Notes: degrees: An double value that defines the tightness of the swirling.

Returns nil on any error.
Sets the last exception property.

5.20.128 Thumbnail(width as Integer, height as Integer) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Notes: Sets the last exception property.
This method was designed by Bob Friesenhahn as a low cost thumbnail generator.
For more details please check the ImageMagick documentation.

5.20.129 TransformImage(CropGeometry as string, ImageGeometry as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransformImage() is a convenience method that behaves like ResizeImage() or CropImage() but accepts scaling and/or cropping information as a region geometry specification. If the operation fails, the original image handle is left as is.
Notes: This should only be used for single images.

CropGeometry: A crop geometry string. This geometry defines a subregion of the image to crop.
ImageGeometry: An image geometry string. This geometry defines the final size of the image.

Returns true on success.

5.20.130 TransformImages(CropGeometry as string, ImageGeometry as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransformImages() calls TransformImage() on each image of a sequence.
Notes: TransformImage() is a convenience method that behaves like ResizeImage() or CropImage() but accepts scaling and/or cropping information as a region geometry specification. If the operation fails, the original image handle is left as is.

CropGeometry: A crop geometry string. This geometry defines a subregion of the image to crop.
ImageGeometry: An image geometry string. This geometry defines the final size of the image.

Returns true on success.

5.20.131 TransformRGBImage(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method TransformRGBImage converts the reference image from an alternate colorspace.
Notes: The transformation matrices are not the standard ones: the weights are rescaled to normalized the range of the transformed values to be [0..MaxRGB] .

colorspace: An integer value that indicates the colorspace the image is currently in. On return the image is in the RGB color space.

Returns false on any error and true on success.

constants:

5.20.132 TransposeImage as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransposeImage() creates a horizontal mirror image by reflecting the pixels around the central y-axis while rotating them by 90 degrees.

5.20.133 TransverseImage as IMImageQ32MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

Function: TransverseImage() creates a vertical mirror image by reflecting the pixels around the central x-axis while rotating them by 270 degrees.

5.20.134 Trim as IMImageQ32MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Trim trims pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Returns nil on any error.
Sets the last exception property.

5.20.135 UnsharpMaskChannel(channel as Integer, radius as Double, sigma as Double, amount as Double, threshold as Double) as IMImageQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and UnsharpMask selects a suitable radius for you.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel          = &h0001
 const GrayChannel         = &h0001
 const CyanChannel         = &h0001
 const GreenChannel        = &h0002
 const MagentaChannel      = &h0002
 const BlueChannel         = &h0004
 const YellowChannel       = &h0004
 const AlphaChannel        = &h0008
 const OpacityChannel      = &h0008
 const BlackChannel        = &h0020
 const IndexChannel        = &h0020
 const AllChannels         = &h7fffffff
```

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.20.136 Wave(amplitude as Double, wavelength as Double) as IMImageQ32MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method Wave creates a new image that is a copy of an existing one with the image pixels altered along a sine wave.
Notes: Parameters are double values that indicates the amplitude and wavelength of the sine wave.
Returns nil on any error.
Sets the last exception property.

5.20.137 WhiteThreshold(threshold as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: WhiteThreshold is like Threshold but forces all pixels above the threshold into white while leaving all pixels below the threshold unchanged.
Notes: No exceptions are generated.
threshold: Define the threshold value. (ASCII string)
For more details please check the ImageMagick documentation.

5.20.138 WriteImage(info as IMImageInfoQ32MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method WriteImage writes an image to a file as defined by image.filename.
Notes: You can specify a particular image format by prefixing the file with the image type and a colon (i.e. ps:image) or specify the image type as the filename suffix (i.e. image.ps). The image may be modified to adapt it to the requirements of the image format. For example, DirectClass images must be color-reduced to PseudoClass if the format is GIF.

WriteImage returns True if the image is written. False is returned if there is a memory shortage or if the image file fails to write.

5.20.139 Properties

5.20.140 BackgroundColor as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Notes: (Read and Write property)

5.20.141 Bias as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.20.142 BlurFactor as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur factor to apply to the image when zooming. Default is 1.0 (no blur).
Notes: (Read and Write property)

5.20.143 BorderColor as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

5.20.144 Colors as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The desired number of colors.
Notes: Used by Quantize().
(Read and Write property)

5.20.145 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

(Read and Write property)

5.20.146 Compression as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.
Notes: useful constants:

```
const UndefinedCompression        = 0
 const NoCompression              = 1
 const BZipCompression            = 2
 const FaxCompression             = 3
 const Group4Compression          = 4
 const JPEGCompression            = 5
 const LosslessJPEGCompression    = 6
 const LZWCompression             = 7
 const RLECompression             = 8
 const ZipCompression             = 9
```

The default is the compression type of the specified image file.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.147 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: QuantumLeap must be defined before a depth of 16 is valid.
(Read and Write property)

5.20.148 Directory as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile names from within an image montage.
Notes: Only valid after calling MontageImages() or reading a MIFF file which contains a directory.
(Read and Write property)

5.20.149 Endian as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian setting to use.
Notes: constants:

UndefinedEndian	0	
LSBEndian	1	(Windows)
MSBEndian	2	(Mac)

e.g. tiff files support different endian settings.
(Read and Write property)

5.20.150 Filename as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path/name.
Notes: The string must be in the encoding of the library and is limited to 4000 bytes.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.151 Filter as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Filter to use when resizing image.
Notes: Constants:

The reduction filter employed has a sipngicant effect on the time required to resize an image and the resulting quality. The default filter is Lanczos which has been shown to produce high quality results when reducing most images.
(Read and Write property)

5.20.152 Fuzz as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this to match colors that are close to the target color in RGB space.
(Read and Write property)

const PointFilter	=1
const BoxFilter	=2
const TriangleFilter	=3
const HermiteFilter	=4
const HanningFilter	=5
const HammingFilter	=6
const BlackmanFilter	=7
const GaussianFilter	=8
const QuadraticFilter	=9
const CubicFilter	=10
const CatromFilter	=11
const MitchellFilter	=12
const LanczosFilter	=13
const BesselFilter	=14
const SincFilter	=15

5.20.153 Gamma as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma level of the image.
Notes: The same color image displayed on two different workstations may look different due to differences in the display monitor. Use gamma correction to adjust for this color difference.
(Read and Write property)

5.20.154 Geometry as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size of the image when encoding.
Notes: (Read and Write property)

5.20.155 Gravity as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.20.156 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to an Image structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.157 Height as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the image in pixels.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.158 Interlace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.

constants:

UndefinedInterlace	0	Unset value.
NoInterlace	1	Don't interlace image (RGBRGBRGBRGBRGBRGB...)
LineInterlace	2	Use scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...)
PlaneInterlace	3	Use plane interlacing (RRRRRR...GGGGGG...BBBBBB...)
PartitionInterlace	4	Similar to plane interlaing except that the different planes are saved to individual files (e.g. image.R, image.G, and image.B)

(Read and Write property)

5.20.159 LastError as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code reported.
Notes: If an exception is raised and it is not a warning exception, this exception code is saved in this property.
(Read and Write property)

5.20.160 LastException as IMExceptionQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception thrown by the Image Magick library.
Notes: You should check this value after every call to the library, process the error and set the property to nil.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.161 Magick as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image encoding format (e.g. ”GIF”).
Notes: (Read and Write property)

5.20.162 Matte as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether an alpha channel is used/present.
Notes: Set to true to enable masks.
(Read and Write property)

5.20.163 MatteColor as IMColorQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (transparent) color.
Notes: (Read and Write property)

5.20.164 Montage as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile size and offset within an image montage. Only valid for montage images.
Notes: (Read and Write property)

5.20.165 Offset as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of initial bytes to skip over when reading raw image.
Notes: (Read and Write property)

5.20.166 Orientation as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image orientation.
Notes: constants:

```
const UndefinedOrientation       = 0
 const TopLeftOrientation        = 1
 const TopRightOrientation       = 2
 const BottomRightOrientation    = 3
 const BottomLeftOrientation     = 4
 const LeftTopOrientation        = 5
 const RightTopOrientation       = 6
 const RightBottomOrientation    = 7
 const LeftBottomOrientation     = 8
```

For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.167 Quality as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level.
Example:

```
dim im as ImageMagickQ32MBS // global
```

```
Function TestJPEG(f as folderitem) As picture
// Reads an image, compresses in memory to JPEG, decompresses using JPEGlib and returns the image
// if quality setting works, you see it in the result.
// no error checking included!

// needs: im as ImageMagickQ32MBS ready initialized

dim image as IMImageQ32MBS
dim imageinfo as IMImageInfoQ32MBS
dim s,blob as string
dim p as Picture
dim i as Integer

if f = nil then
Return nil
end if

imageinfo = im.NewImageInfo

imageinfo.Filename = f.NativePath

//this should read any image IM understands
image = im.ReadImage(imageinfo)
//check for error
if im.lastexception <>nil and im.LastException.Severity >= 400 then
s = "LastError: "+Format(im.LastError,"-0")+" - Severity: "+str(im.LastException.Severity)+EndOfLine+im.LastException.Reason
MsgBox s
Return nil
elseif image = nil then
MsgBox "image=nil"
Return nil
end if

// Now lets convert to jpeg
imageinfo.Filename = "image.jpg"
imageinfo.Quality = 10 // 100 is max
blob = image.ImageToBlob(imageinfo)

// It may fail
if blob.lenb = 0 then
Return nil
end if
p = JPEGStringToPictureMBS(blob,true)

image.DestroyImage
imageinfo.DestroyImageInfo
```

```
Return p
Exception
Return nil
End Function
```

Notes: Default value is 75.
(Read and Write property)

5.20.168 Release as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: If true, the destructor will release the handle.
Notes: (Read and Write property)

5.20.169 RenderingIntent as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rendering intent to use.
Notes: constants:

UndefinedIntent	0
SaturationIntent	1
PerceptualIntent	2
AbsoluteIntent	3
RelativeIntent	4

(Read and Write property)

5.20.170 ResolutionUnits as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Notes: constants:

(Read and Write property)

UndefinedResolution	0	Unset value.
PixelsPerInchResolution	1	Density specifications are specified in units of pixels per inch (english units).
PixelsPerCentimeterResolution	2	Density specifications are specified in units of pixels per centimeter (metric units).

5.20.171 ResolutionX as Double

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The horizontal resolution of the image.
Notes: The unit for resolution must be specified.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.172 ResolutionY as Double

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The vertical resolution of the image.
Notes: The unit for resolution must be specified.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.173 Scene as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.20.174 StorageClass as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image storage class.
Notes: If DirectClass then the image packets contain valid RGB or CMYK colors. If PseudoClass then the image has a colormap referenced by pixel's index member.

constants:

UndefinedClass	0	Unset value.
DirectClass	1	Image is composed of pixels which represent literal color values.
PseudoClass	2	Image is composed of pixels which specify an index in a color palette.

(Read and Write property)

5.20.175 Taint as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set to True if the image pixels have been modified.
Notes: (Read and Write property)

5.20.176 Width as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the image in pixels.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.20.177 Constants

Constants

Constant	Value	Description
kBackgroundDispose	2	One of the Image layer Dispose Types.
kCoalesceLayer	1	One of the Image layer method constants.
kCompareAnyLayer	2	One of the Image layer method constants.
kCompareClearLayer	3	One of the Image layer method constants.
kCompareOverlayLayer	4	One of the Image layer method constants.
kCompositeLayer	12	One of the Image layer method constants.
kDisposeLayer	5	One of the Image layer method constants.
kFlattenLayer	14	One of the Image layer method constants.
kMergeLayer	13	One of the Image layer method constants.
kMosaicLayer	15	One of the Image layer method constants.
kNoneDispose	1	One of the Image layer Dispose Types.
kOptimizeImageLayer	7	One of the Image layer method constants.
kOptimizeLayer	6	One of the Image layer method constants.
kOptimizePlusLayer	8	One of the Image layer method constants.
kOptimizeTransLayer	9	One of the Image layer method constants.
kPreviousDispose	3	One of the Image layer Dispose Types.
kRemoveDupsLayer	10	One of the Image layer method constants.
kRemoveZeroLayer	11	One of the Image layer method constants.
kUndefinedDispose	0	One of the Image layer Dispose Types.
kUndefinedLayer	0	One of the Image layer method constants.
kUnrecognizedDispose	0	One of the Image layer Dispose Types.

Distortion Effects

Constant	Value	Description
kAffineDistortion	1	
kAffineDistortion	1	
kAffineProjectionDistortion	2	
kAffineProjectionDistortion	2	
kArcDistortion	9	
kArcDistortion	9	
kBarrelDistortion	14	
kBarrelDistortion	14	
kBarrelInverseDistortion	15	
kBarrelInverseDistortion	15	
kBilinearDistortion	6	
kBilinearDistortion	6	
kBilinearForwardDistortion	6	
kBilinearForwardDistortion	6	
kBilinearReverseDistortion	7	
kBilinearReverseDistortion	7	
kCylinder2PlaneDistortion	12	
kCylinder2PlaneDistortion	12	
kDePolarDistortion	11	
kDePolarDistortion	11	
kPerspectiveDistortion	4	
kPerspectiveDistortion	4	
kPerspectiveProjectionDistortion	5	
kPerspectiveProjectionDistortion	5	
kPlane2CylinderDistortion	13	
kPlane2CylinderDistortion	13	
kPolarDistortion	10	
kPolarDistortion	10	
kPolynomialDistortion	8	
kPolynomialDistortion	8	
kResizeDistortion	17	
kResizeDistortion	17	
kScaleRotateTranslateDistortion	3	
kScaleRotateTranslateDistortion	3	
kSentinelDistortion	18	
kSentinelDistortion	18	
kShepardsDistortion	16	
kShepardsDistortion	16	
kUndefinedDistortion	0	
kUndefinedDistortion	0	

Interpolate Modes

Constant	Value	Description
kBarycentricColorInterpolate	1	
kBarycentricColorInterpolate	1	
kBilinearColorInterpolate	7	
kBilinearColorInterpolate	7	
kInverseColorInterpolate	19	
kInverseColorInterpolate	19	
kPolynomialColorInterpolate	8	
kPolynomialColorInterpolate	8	
kShepardsColorInterpolate	16	
kShepardsColorInterpolate	16	
kUndefinedColorInterpolate	0	
kUndefinedColorInterpolate	0	
kVoronoiColorInterpolate	18	
kVoronoiColorInterpolate	18	

5.21 class IMImageQ8MBS

5.21.1 class IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for an Image Magick Image in memory.
Notes: Can exist with or without pixel data.

For more details please check the ImageMagick documentation.
Blog Entries

- MBS Xojo / Real Studio Plugins, version 15.1pr4

5.21.2 Methods

5.21.3 AdaptiveThreshold(width as Integer, height as Integer, offset as Integer) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: AdaptiveThreshold selects an individual threshold for each pixel based on the range of intensity values in its local neighborhood.
Notes: This allows for thresholding of an image whose global intensity histogram doesn't contain distinctive peaks.
Sets the last exception property.

width: The width of the local neighborhood.
height: The height of the local neighborhood.
offset: The mean offset.

For more details please check the ImageMagick documentation.

5.21.4 AddNoise(NoiseType as Integer) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds random noise to the image.
Notes: Constants

For more details please check the ImageMagick documentation.
Sets the last exception property.

UndefinedNoise	=0
UniformNoise	=1
GaussianNoise	=2
MultiplicativeGaussianNoise	=3
ImpulseNoise	=4
LaplacianNoise	=5
PoissonNoise	=6

5.21.5 AffineTransformImage(matrix as IMImageAffineMatrixQ8MBS) as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transforms an image as dictated by the affine matrix.

5.21.6 AppendImageToList(img as IMImageQ8MBS)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds an image to the image list.
Notes: For more details please check the ImageMagick documentation.

5.21.7 AutoGammaImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoGammaImage extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.
Notes: Returns true on success or false on failure.

5.21.8 AutoGammaImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoGammaImage extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.
Notes: Returns true on success or false on failure.

channelType: The channels to auto-level. If the special 'SyncChannels' flag is set all given channels is adjusted in the same way using the mean average of those channels.

Constants for channel:

```
 const UndefinedChannel    = 0
  const RedChannel          = &h0001
  const GrayChannel         = &h0001
  const CyanChannel         = &h0001
  const GreenChannel        = &h0002
  const MagentaChannel      = &h0002
  const BlueChannel         = &h0004
  const YellowChannel       = &h0004
  const AlphaChannel        = &h0008
  const OpacityChannel      = &h0008
  const BlackChannel        = &h0020
  const IndexChannel        = &h0020
  const AllChannels         = &h7ffffff
```

5.21.9 AutoLevelImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoLevelImage adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: Returns true on success or false on failure.

5.21.10 AutoLevelImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: AutoLevelImage adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: Returns true on success or false on failure.

ChannelType: The channels to auto-level. If the special 'SyncChannels' flag is set the min/max/mean value of all given channels is used for all given channels, to all channels in the same way.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

5.21.11 Average as IMImageQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Average() method takes a set of images and averages them together.
Notes: Each image in the set must have the same width and height. Average() returns a single image with each corresponding pixel component of each image averaged. On failure, a nil image is returned and exception describes the reason for the failure.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.12 BilevelChannel(channel as Integer, threshold as Double) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the value of individual pixels based on the intensity of each pixel channel.
Notes: The result is a high-contrast image.

channel: The channel type.
threshold: define the threshold values.

Constants for channel:

For more details please check the ImageMagick documentation.

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

5.21.13 BlackThreshold(threshold as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: BlackThreshold is like Threshold but forces all pixels below the threshold into black while leaving all pixels above the threshold unchanged.
Notes: No exceptions are generated.
threshold: Define the threshold value. (ASCII string)
For more details please check the ImageMagick documentation.

5.21.14 BlobSize as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The expected size for this image written to a file.
Notes: For more details please check the ImageMagick documentation.

5.21.15 Blur(radius as Double, sigma as Double) as IMImageQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and BlurImage selects a suitable radius for you.

radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Gaussian, in pixels.

For more details please check the ImageMagick documentation.

5.21.16 BlurImageChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and BlurImageChannel selects a suitable radius for you.

channel: The channel type.
radius: The radius of the Gaussian, in pixels, not counting the center pixel.
sigma: The standard deviation of the Gaussian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.21.17 BorderImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Surrounds the image with a border of the color defined by the bordercolor member of the image.
Notes: The width and height of the border are defined by the corresponding parameters.

5.21.18 BrightnessContrastImage(brightness as Double, contrast as Double) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image. It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.
Notes: Returns true on success or false on failure.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).

5.21.19 BrightnessContrastImageChannel(ChannelType as Integer, brightness as Double, contrast as Double) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image. It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.
Notes: Returns true on success or false on failure.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).
ChannelType: The channels to use.

Constants for channel:

5.21.20 Charcoal(radius as Double, sigma as Double) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Charcoal creates a new image that is a copy of an existing one with the edge highlighted.
Notes: radius: the radius of the pixel neighborhood.
sigma: The standard deviation of the Gaussian, in pixels.

Returns nil on any error.
Sets the last exception property.

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.21.21 Chop(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Chop removes a region of an image and collapses the image to occupy the removed portion.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.22 ClipPath(path as string, inside as boolean) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the image clip mask based any clipping path information if it exists.
Notes:

pathname: name of clipping path resource. If name is preceded by #, use clipping path numbered by name.

inside: if true, later operations take effect inside clipping path. Otherwise later operations take effect outside clipping path.

Returns true on success and false on any error.

5.21.23 Clone as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of this image object.
Notes: For more details please check the ImageMagick documentation.

5.21.24 CloneImageAttributes(image as IMImageAttributeQ8MBS) as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: CloneImageAttributes() clones one or more image attributes.
Notes: Returns false on any error.

5.21.25 CloneImageProfiles(SourceImage as IMImageQ8MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones one or more image profiles.
Notes: Returns false on any error and true on success.

5.21.26 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.21.27 ClutImage(clutImage as IMImageQ8MBS) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either

for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

ClutImage: the color lookup table image for replacement color values.

Returns true on success or false on failure.

5.21.28 ClutImageChannel(ChannelType as Integer, clutImage as IMImageQ8MBS) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

ClutImage: the color lookup table image for replacement color values.
ChannelType: The channels to use.

Returns true on success or false on failure.

Constants for channel:

5.21.29 CoalesceImages as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: CoalesceImages composites a set of images while respecting any page offsets and disposal meth-

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

ods.
Notes: GIF, MIFF, and MNG animation sequences typically start with an image background and each subsequent image varies in size and offset. CoalesceImages() returns a new sequence where each image in the sequence is the same size as the first and composited with the next image in the sequence.

Returns nil on any error.
Sets the last exception property.

5.21.30 Colorize(opacity as string, PenColorRed as Integer, PenColorGreen as Integer, PenColorBlue as Integer, PenColorOpacity as Integer) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ColorizeImage creates a new image that is a copy of an existing one with the image pixels colorized.
Notes: The colorization is controlled with the pen color and the opacity levels.

opacity: A character string indicating the level of opacity as a percentage (0-100).
PenColorRed, PenColorGreen, PenColorBlue and PenColorOpacity define the pen color used.

Returns nil on any error.
Sets the last exception property.

5.21.31 Combine(channel as Integer) as IMImageQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Combines one or more images into a single image.
Notes: The grayscale value of the pixels of each image in the sequence is assigned in order to the specified channels of the combined image. The typical ordering would be image 1 =>Red, 2 =>Green, 3 =>Blue, etc.

The lastexception property is set.

5.21.32 CompareImageLayers(ImageLayerMethod as Integer) as IMImageQ8MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: CompareImageLayers() compares each image with the next in a sequence and returns the minimum bounding region of all the pixel differences (of the mageLayerMethod specified) it discovers.
Notes: Images do NOT have to be the same size, though it is best that all the images are 'coalesced' (images are all the same size, on a flattened canvas, so as to represent exactly how an specific frame should look).

No GIF dispose methods are applied, so GIF animations must be coalesced before applying this image operator to find differences to them.

ImageLayerMethod:
the layers type to compare images with. Must be one of... CompareAnyLayer, CompareClearLayer, CompareOverlayLayer.

Can raise an exception.

5.21.33 Composite(ComposeOperator as Integer, Image as IMImageQ8MBS, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the second image composited onto the first at the specified offsets.
Notes: compose: Specifies an image composite operator.
Image: The second image.
x: An integer that specifies the column offset of the composited image.
y: An integer that specifies the row offset of the composited image.

No error code and exception!

5.21.34 ConsolidateCMYKImages as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Consolidates a sequence of CMYK images.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.35 ContrastImage(sharpen as boolean) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the intensity differences between the lighter and darker elements of the image.
Notes: Returns true on success or false on failure.
Set sharpen to true to increase the image contrast otherwise the contrast is reduced.

5.21.36 CopyPicture as picture

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ8MBS // your image
Canvas1.Backdrop=image.CopyPicture
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
See also:

- 5.21.37 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture 814

5.21.37 CopyPicture(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a portion of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ8MBS // your image
Canvas1.Backdrop=image.CopyPicture(0,0,image.Width,image.Height)
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.
See also:

- 5.21.36 CopyPicture as picture 814

5.21.38 CopyPictureMask as picture

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the mask of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ8MBS // your image
Canvas1.Backdrop=image.CopyPictureMask
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
See also:

- 5.21.39 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture 815

5.21.39 CopyPictureMask(x as Integer, y as Integer, width as Integer, height as Integer) as picture

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a portion of the mask of the Image Magick Image and returns a Xojo picture.
Example:

```
dim image as IMImageQ8MBS // your image
Canvas1.Backdrop=image.CopyPictureMask(0,0,image.Width,image.Height)
```

Notes: Sets the last exception property.
Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.
See also:

- 5.21.38 CopyPictureMask as picture 815

5.21.40 CopyPixel(x as Integer, y as Integer) as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies a pixel.
Notes: Returns nil on any error.
This method works only for bitmap images.
x and y are zero based.

5.21.41 CreateHBITMAP as Ptr

Plugin Version: 15.1, Platform: Windows, Targets: All.

Function: Creates a HBITMAP for the image for use with Windows Declares.
Notes: The HBITMAP returned needs to be freed when you are done with it or you risk having a memory leak.

5.21.42 Crop(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crop extracts a region of the image starting at the offset defined by geometry.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.43 CropImageToTiles(CropGeometry as string) as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crops a single image, into a possible list of tiles.
Notes: This may include a single sub-region of the image. This basically applies all the normal geometry flags for Crop.

5.21.44 CycleColormap(displace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Displaces an image's colormap by a given number of positions.
Notes: If you cycle the colormap a number of times you can produce a psychodelic effect.
Returns true on success.
displace: displace the colormap this amount.

5.21.45 DecipherImage(passkey as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts cipher pixels to plain pixels.
Notes: Passkey: decipher cipher pixels with this passphrase.
Returns true on success.

5.21.46 DeconstructImages as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: DeconstructImages() compares each image with the next in a sequence and returns the minimum bounding region of all differences from the first image.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.47 DeleteImageAttribute(key as string) as Boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: DeleteImageAttribute() deletes an attribute from the image.
Notes: Returns false on any error.

5.21.48 Despeckle() as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduces the speckle noise in an image while perserving the edges of the original image.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.49 DestroyImage

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases the memory used for this image and sets handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.21.50 DestroyImageAttributes

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Deallocates memory associated with the image attribute list.

5.21.51 DestroyImageList

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys the image list and sets the handle to 0.
Notes: For more details please check the ImageMagick documentation.
The destructor will call this for you if release=true.

5.21.52 DestroyImageProfiles

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases memory associated with an image profile map.

5.21.53 DistortImage(DistortImageMethod as Integer, values() as Double, bestfit as boolean) as IMImageQ8MBS

Plugin Version: 12.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: DistortImage() distorts an image using various distortion methods, by mapping color lookups of the source image to a new destination image usally of the same size as the source image, unless 'bestfit' is set to true.
Notes: If 'bestfit' is enabled, and distortion allows it, the destination image is adjusted to ensure the whole source 'image' will just fit within the final destination image, which will be sized and offset accordingly. Also in many cases the virtual offset of the source image will be taken into account in the mapping.

If the '-verbose' control option has been set print to standard error the equicelent '-fx' formula with coefficients for the function, if practical.

A description of each parameter follows:
self: the image to be distorted.
m: the method of image distortion. ArcDistortion always ignores source image offset, and always 'bestfit' the destination image with the top left corner offset relative to the polar mapping center. Affine, Perspective, and Bilinear, do least squares fitting of the distrotion when more than the minimum number of control point pairs are provided. Perspective, and Bilinear, fall back to a Affine distortion when less than 4 control point pairs are provided. While Affine distortions let you use any number of control point pairs, that is Zero pairs is a No-Op (viewport only) distortion, one pair is a translation and two pairs of control points do a scale-rotate-translate, without any shearing.
values: arguments given.
bestfit: Attempt to 'bestfit' the size of the resulting image. This also forces the resulting image to be a 'layered' virtual canvas image. Can be overridden using 'distort:viewport' setting.

Extra Controls from Image meta-data (artifacts)...

- "verbose" Output to stderr alternatives, internal coefficents, and FX equivalents for the distortion operation (if feasible). This forms an extra check of the distortion method, and allows users access to the internal constants IM calculates for the distortion.
- "distort:viewport" Directly set the output image canvas area and offest to use for the resulting image, rather than use the original images canvas, or a calculated 'bestfit' canvas.
- "distort:scale" Scale the size of the output canvas by this amount to provide a method of Zooming, and for super-sampling the results.

Other settings that can effect results include

- 'interpolate' For source image lookups (scale enlargements)
- 'filter' Set filter to use for area-resampling (scale shrinking). Set to 'point' to turn off and use 'interpolate' lookup instead

5.21.54 Edge(radius as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Finds edges in an image.
Notes: Radius defines the radius of the convolution filter. Use a radius of 0 and Edge selects a suitable radius for you.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.55 Emboss(radius as Double, sigma as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a grayscale image with a three-dimensional effect.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, radius should be larger than sigma. Use a radius of 0 and Emboss selects a suitable radius for you.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.56 EncipherImage(passkey as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts pixels to cipher-pixels.
Notes: passkey: encipher pixels with this passphrase.
Returns true on success.

5.21.57 EqualizeImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: Returns true on success or false on failure.

ChannelType: The channels to use.

5.21.58 EqualizeImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: Returns true on success or false on failure.

ChannelType: The channels to use.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.21.59 ExcerptImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a excerpt of the image as defined by the geometry.
Notes: Define the region of the image to extend with x, y, width, and height.

5.21.60 ExtentImage(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends the image as defined by the geometry, gravity, and image background color.
Notes: Define the region of the image to extend with x, y, width, and height.

Set the (x,y) offset of the geometry to move the original image relative to the extended image.

5.21.61 FlattenImages as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flatten composites all images from the current image pointer to the end of the image list and returns a single flattened image.
Notes: Returns nil on any error.

Sets the last exception property.

5.21.62 Flip as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flip creates a vertical mirror image by reflecting the pixels around the central x-axis.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.63 Flop as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Flop creates a horizontal mirror image by reflecting the pixels around the central y-axis.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.64 FrameImage(x as Integer, y as Integer, width as Integer, height as Integer, innerBevel as Integer, OuterBevel as Integer) as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a simulated three-dimensional border around the image.
Notes: The color of the border is defined by the MatteColor of image. Width and height specify the border width of the vertical and horizontal sides of the frame. innerBevel and OuterBevel indicate the width of the inner and outer shadows of the frame.

5.21.65 FxImage(expression as string) as IMImageQ8MBS

Plugin Version: 8.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: FxImage() applies a mathematical expression to the specified image.
Notes: Can raise an exception.

5.21.66 GaussianBlurChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, the radius should be larger than sigma. Use a radius of 0 and GaussianBlur selects a suitable radius for you.
Sets the last exception property.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
channel: The channel type.
sigma: the standard deviation of the Gaussian, in pixels.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

For more details please check the ImageMagick documentation.

5.21.67 GetImageAttribute(key as string) as IMImageAttributeQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetImageAttribute searches the list of image attributes and returns a reference to the attribute if it exists otherwise nil.

5.21.68 GetImageClippingPathAttribute as IMImageAttributeQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetImageClippingPathAttribute searches the list of image attributes and returns a reference to a clipping path if it exists otherwise nil.

5.21.69 GetImageProfile(name as string) as string

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets a profile associated with an image by name.
Notes: Returns "" on any error.

5.21.70 GetNextImageAttribute as IMImageAttributeQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: GetNextImageAttribute() gets the next image attribute.
Notes: Returns nil on any error.

5.21.71 GetNextImageProfile as string

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets the next profile name for an image.
Notes: Returns "" on any error.

5.21.72 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole Image structure copied into a memoryblock.
Notes: Returns nil on any error.

5.21.73 ImagesToBlob(info as IMImageInfoQ8MBS) as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImagesToBlob implements direct to memory image formats.
Notes: It returns the image sequence as a string. The magick member of the ImageInfo structure determines the format of the returned blob (GIF, JPEG, PNG, etc.)

Note, some image formats do not permit multiple images to the same image stream (e.g. JPEG). in this instance, just the first image of the sequence is returned as a blob.

Sets the last exception property and returns "" on any error.

For more details please check the ImageMagick documentation.

5.21.74 ImageToBlob(info as IMImageInfoQ8MBS) as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImagesToBlob implements direct to memory image formats.
Example:

```
dim im as ImageMagickQ8MBS // global

Function IMPictureToString(p as picture, magick as string, quality as Integer) As string
dim image as new IMImageQ8MBS
dim imageinfo as IMImageInfoQ8MBS
dim s,data as string
dim impp as new IMMagickPixelPacketQ8MBS

// empty string for nil picture
if p = nil then
Return ""
end if

// create a new picture info

imageinfo = im.NewImageInfo
imageinfo.ColorSpace=1
// only color space is needed. 1 for RGB.

// background color of image
impp.red = 0
impp.Green = 0
impp.Blue = 0

// creates a new image object
if not image.NewImage(imageinfo,p.Width,p.Height,impp) then
Return ""
end if

// copy RB picture into IM Image at position 0/0
image.ColorSpace = 1
image.SetPicture(p,0,0)

// set compression data
imageinfo.Magick = magick
imageinfo.Quality = quality

// and rendering intent: 2=PerceptualIntent
```

```
'image.RenderingIntent = 2

// create image data
data = image.ImageToBlob(imageinfo)

// release memory
image.DestroyImage
imageinfo.DestroyImageInfo

// return result
Return data

Exception
// in case of an exception return nothing
Return ""


End Function
```

Notes: It returns the image sequence as a string. The magick member of the ImageInfo structure determines the format of the returned blob (GIF, JPEG, PNG, etc.)

Note, some image formats do not permit multiple images to the same image stream (e.g. JPEG). in this instance, just the first image of the sequence is returned as a blob.

Sets the last exception property and returns "" on any error.
For more details please check the ImageMagick documentation.

5.21.75 Implode(factor as Double) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ImplodeImage creates a new image that is a copy of an existing one with the image pixels "implode" by the specified percentage.
Notes: factor: A double value that defines the extent of the implosion.

Returns nil on any error.
Sets the last exception property.

5.21.76 IsBlobExempt as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is exempt.
Notes: For more details please check the ImageMagick documentation.

5.21.77 IsBlobSeekable as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is seekable.
Notes: For more details please check the ImageMagick documentation.

5.21.78 IsBlobTemporary as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is temporary.
Notes: For more details please check the ImageMagick documentation.

5.21.79 Magnify as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A convenience method that scales an image proportionally to twice its size.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.80 MedianFilter(radius as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a digital filter that improves the quality of a noisy image.
Notes: Each pixel is replaced by the median in a set of neighboring pixels as defined by radius.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.81 MergeImageLayers(ImageLayerMethod as Integer) as IMImageQ8MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: MergeImageLayers() composes all the image layers from the current given image onward to produce a single image of the merged layers.
Notes: The inital canvas's size depends on the given ImageLayerMethod, and is initialized using the first images images background color. The images are then compositied onto that image in sequence using the given composition that has been assigned to each individual image.

ImageLayerMethod:
the method of selecting the size of the initial canvas.

MergeLayer: Merge all layers onto a canvas just large enough to hold all the actual images. The virtual canvas of the first image is preserved but otherwise ignored.

FlattenLayer: Use the virtual canvas size of first image. Images which fall outside this canvas is clipped. This can be used to 'fill out' a given virtual canvas.

MosaicLayer: Start with the virtual canvas of the first image, enlarging left and right edges to contain all images. Images with negative offsets will be clipped.

Can raise an exception.

5.21.82 Minify as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A convenience method that scales an image proportionally to half its size.
Notes: Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.83 MosaicImages as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: MosaicImages inlays an image sequence to form a single coherent picture.
Notes: It returns a single image with each image in the sequence composited at the location defined by the page member of the image structure.
Returns nil on any error.
Sets the last exception property.

5.21.84 MotionBlur(radius as Double, sigma as Double, angle as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Simulates motion blur.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and MotionBlur selects a suitable radius for you. Angle gives the angle of the blurring motion.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.85 NegateImage(gray as boolean = false) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: Returns true on success or false on failure.
The grayscale option means that only grayscale values within the image are negated.

gray: If true, only negate grayscale pixels within the image.

5.21.86 NegateImageChannel(ChannelType as Integer, gray as boolean = false) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: Returns true on success or false on failure.
The grayscale option means that only grayscale values within the image are negated.

ChannelType: The channels to use.
gray: If true, only negate grayscale pixels within the image.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7fffffff
```

5.21.87 NewImage(info as IMImageInfoQ8MBS, width as Integer, height as Integer, background as IMMagickPixelPacketQ8MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image.
Example:

```
dim im as ImageMagickQ8MBS // global
dim p as picture
dim imageinfo as IMImageInfoQ8MBS
dim image as IMImageQ8MBS
dim b as new IMMagickPixelPacketQ8MBS
b.Blue=65535
b.ColorSpace=1 // RGB
b.Depth=16

imageinfo = im.NewImageInfo
imageinfo.Depth=16
imageinfo.ColorSpace=1

//this should read any image IM understands
image = new IMImageQ8MBS
if image.NewImage(imageinfo,500,500,b) then
p=New Picture(300,300,32)
p.Graphics.ForeColor=Rgb(255,0,0)
p.Graphics.FillOval 0,0,300,300
image.SetPicture p,0,0
else
MsgBox "failed"
end if
```

Notes: Returns false on failure and true on success.

5.21.88 NormalizeImage as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: Returns true on success or false on failure.

5.21.89 NormalizeImageChannel(ChannelType as Integer) as Boolean

Plugin Version: 15.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: Returns true on success or false on failure.

ChannelType: The channels to auto-level. If the special 'SyncChannels' flag is set the min/max/mean value of all given channels is used for all given channels, to all channels in the same way.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7fffffff
```

5.21.90 OilPaint(radius as Double) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method OilPaintImage creates a new image that is a copy of an existing one with each pixel component replaced with the color of greatest frequency in a circular neighborhood.
Notes: radius parameter: radius of the circular neighborhood.
Returns nil on any error.
Sets the last exception property.

5.21.91 OptimizeImageLayers as IMImageQ8MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImageLayers() compares each image the GIF disposed forms of the previous image in the sequence.
Notes: From this it attempts to select the smallest cropped image to replace each frame, while preserving the results of the GIF animation.

Can raise an exception.

5.21.92 OptimizeImageTransparency

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImageTransparency() takes a frame optimized GIF animation, and compares the overlayed pixels against the disposal image resulting from all the previous frames in the animation.
Notes: Any pixel that does not change the disposal image (and thus does not effect the outcome of an overlay) is made transparent.

WARNING: This modifies the current images directly, rather than generate a new image sequence.

Can raise an exception.

5.21.93 OptimizePlusImageLayers as IMImageQ8MBS

Plugin Version: 8.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizeImagePlusLayers() is exactly as OptimizeImageLayers(), but may also add or even remove extra frames in the animation, if it improves the total number of pixels in the resulting GIF animation.
Notes: Can raise an exception.

5.21.94 ProfileImage(name as string, ProfileData as string) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds or removes a ICC, IPTC, or generic profile from an image.
Notes: If the ProfileData is "", it is removed from the image otherwise added. Use a name of '*' and a ProfileData of "" to remove all profiles from the image.

Returns false on any error and true on success.

5.21.95 RadialBlur(angle as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: RadialBlur applies a radial blur to the image.
Notes: angle: The angle of the radial blur.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.96 RaiseImage(x as Integer, y as Integer, width as Integer, height as Integer, raise as boolean) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a simulated three-dimensional button-like effect by lightening and darkening the edges of the image.
Notes: Width and height define the width of the vertical and horizontal edge of the effect.
raise: A value other than zero creates a 3-D raise effect, otherwise it has a lowered effect.

5.21.97 RandomThresholdChannel(channel as Integer, thresholds as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Canges the value of individual pixels based on the intensity of each pixel compared to a random threshold.
Notes: The result is a low-contrast, two color image.

channel: The channel or channels to be thresholded.
thresholds: a geometry string containing low,high thresholds. If the string contains 2x2, 3x3, or 4x4, an

ordered dither of order 2, 3, or 4 is performed instead. (ASCII string)

Sets the last exception property.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel        = &h0001
 const GrayChannel       = &h0001
 const CyanChannel       = &h0001
 const GreenChannel      = &h0002
 const MagentaChannel    = &h0002
 const BlueChannel       = &h0004
 const YellowChannel     = &h0004
 const AlphaChannel      = &h0008
 const OpacityChannel    = &h0008
 const BlackChannel      = &h0020
 const IndexChannel      = &h0020
 const AllChannels       = &h7ffffff
```

For more details please check the ImageMagick documentation.

5.21.98 ReduceNoise(radius as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Smooths the contours of an image while still preserving edge information.
Notes: The algorithm works by replacing each pixel with its neighbor closest in value. A neighbor is defined by radius. Use a radius of 0 and ReduceNoise selects a suitable radius for you.

For more details please check the ImageMagick documentation.

5.21.99 RemoveDuplicateLayers

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that is exactly the same as the next image in the given image list.
Notes: Image size and virtual canvas offset must also match, though not the virtual canvas size itself.

No check is made with regards to image disposal setting, though it is the dispose setting of later image that is kept. Also any time delays are also added together. As such coalesced image animations should still produce the same result, though with duplicte frames merged into a single frame.

5.21.100 RemoveFirstImageFromList as IMImageQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes the first image from the image list and returns the image.
Notes: Returns nil on any error.
For more details please check the ImageMagick documentation.

5.21.101 RemoveImageProfile(name as string) as string

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes a profile from the image-map by its name.

5.21.102 RemoveZeroDelayLayers

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that as a zero delay time.
Notes: Such images generally represent intermediate or partial updates in GIF animations used for file optimization. They are not ment to be displayed to users of the animation. Viewable images in an animation should have a time delay of 3 or more centi-seconds (hundredths of a second).

However if all the frames have a zero time delay, then either the animation is as yet incomplete, or it is not a GIF animation. This is a non-sensible situation, so no image will be removed and a 'Zero Time Animation' warning (exception) given.

No warning will be given if no image was removed because all images had an appropriate non-zero time delay set.

Due to the special requirements of GIF disposal handling, GIF animations should be coalesced first, before calling this function, though that is not a requirement.

5.21.103 ResetImageAttributeIterator

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: ResetImageAttributeIterator() resets the image attributes iterator.
Notes: Use it in conjunction with GetNextImageAttribute() to iterate over all the values associated with an image.

5.21.104 ResetImageProfileIterator

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image profile iterator.
Notes: Use it in conjunction with GetNextImageProfile() to iterate over all the profiles associated with an image.

5.21.105 Resize(width as Integer, height as Integer, FilterID as Integer, blur as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions.
Notes: Constants for the FilterID:

```
const PointFilter      =1
 const BoxFilter        =2
 const TriangleFilter   =3
 const HermiteFilter    =4
 const HanningFilter    =5
 const HammingFilter    =6
 const BlackmanFilter   =7
 const GaussianFilter   =8
 const QuadraticFilter  =9
 const CubicFilter      =10
 const CatromFilter     =11
 const MitchellFilter   =12
 const LanczosFilter    =13
 const BesselFilter     =14
 const SincFilter       =15
```

Most of the filters are FIR (finite impulse response), however, Bessel, Gaussian, and Sinc are IIR (infinite impulse response). Bessel and Sinc are windowed (brought down to zero) with the Blackman filter.

Sets the last exception property.

5.21.106 RGBTransformImage(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method RGBTransformImage converts the reference image from RGB to an alternate colorspace.
Notes: The transformation matrices are not the standard ones: the weights are rescaled to normalized the range of the transformed values to be [0..MaxRGB] .

colorspace: An integer value that indicates which colorspace to transform the image.

Returns false on any error and true on success.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

5.21.107 Roll(x as Integer, y as Integer) as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Roll offsets an image as defined by x and y.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.108 Rotate(degrees as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotation of an image.
Notes: Method RotateImage creates a new image that is a rotated copy of an existing one. Positive angles rotate counter-clockwise (right-hand rule), while negative angles rotate clockwise. Rotated images are usually larger than the originals and have 'empty' triangular corners. X axis. Empty triangles left over from shearing the image are filled with the color specified by the image background_color. RotateImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Method RotateImage is based on the paper "A Fast Algorithm for General Raster Rotatation" by Alan W. Paeth. RotateImage is adapted from a similar method based on the Paeth paper written by Michael Halle of the Spatial Imaging Group, MIT Media Lab.

degrees: Specifies the number of degrees to rotate the image.

Sets the lastexception property.
Returns nil on low memory.
For more details please check the ImageMagick documentation.

5.21.109 Sample(width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions with pixel sampling.
Notes: Unlike other scaling methods, this method does not introduce any additional color into the scaled image.
For more details please check the ImageMagick documentation.
Sets the last exception property.

5.21.110 Scale(width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Example:

```
dim image as IMImageQ8MBS // your image
image=Image.Scale(100,80)
```

Notes: This method was designed by Bob Friesenhahn as a low cost thumbnail generator.

columns: The number of columns in the scaled image.
rows: The number of rows in the scaled image.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.111 SetImageAttribute(key as string, value as string) as boolean

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: SetImageAttribute searches the list of image attributes and replaces the attribute value.
Notes: If it is not found in the list, the attribute name and value is added to the list. If the attribute exists in the list, the value is concatenated to the attribute. SetImageAttribute returns True if the attribute is successfully concatenated or added to the list, otherwise False. If the value is "", the matching key is deleted from the list.

5.21.112 SetImageColorspace(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the colorspace member of the Image structure.
Notes: Returns false on any error and true on success.

5.21.113 SetImageProfile(name as string, ProfileData as string) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a named profile to the image.
Notes: If a profile with the same name already exists, it is replaced. This method differs from the ProfileImage() method in that it does not apply CMS color profiles.

name: The profile name.
profiledata: The binary data of the profile.

Returns false on any error and true on success.

5.21.114 SetPicture(pic as picture, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the pixels from a given Xojo picture into the Image Magick Image at the given location.
Example:

```
dim image as IMImageQ8MBS // your image
dim p as picture

p=New Picture(32,32,32)
p.Graphics.ForeColor=rgb(0,255,0)
p.Graphics.FillRect 0,0,32,32

image.SetPicture(p,30,30)
```

Notes: Sets the last exception property.
The method will do nothing on bad bounds.
This method works only for bitmap images.
x and y are zero based.

5.21.115 SetPictureMask(maskpic as picture, x as Integer, y as Integer)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies the pixels from a given Xojo picture into the mask of the Image Magick Image at the given location.
Example:

```
dim i as IMImageQ8MBS // your image
dim p as picture

p=New Picture(32,32,32)
p.Graphics.ForeColor=rgb(0,255,0)
p.Graphics.FillRect 0,0,32,32

i.SetPictureMask(p,30,30)
```

Notes: Sets the last exception property.
The method will do nothing on bad bounds.
This method works only for bitmap images.
x and y are zero based.
You may need to set matte=True after this.

5.21.116 SetPixel(x as Integer, y as Integer, newPixel as IMColorQ8MBS)

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets a pixel value.
Example:

```
dim image as IMImageQ8MBS // your image
dim co as IMColorQ8MBS

co=new IMColorQ8MBS
co.blue=65535 // max value
image.SetPixel 50,50,co // Makes Pixel 50/50 blue
```

Notes: The method will fail silently if the values are out of bounds or the image is not a bitmap image.
This method works only for bitmap images.
x and y are zero based.

5.21.117 Shade(gray as boolean, azimuth as Double, elevation as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shines a distant light on an image to create a three-dimensional effect.
Notes: You control the positioning of the light with azimuth and elevation; azimuth is measured in degrees off the x axis and elevation is measured in pixels above the Z axis.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.118 SharpenChannel(channel as Integer, radius as Double, sigma as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma) . For reasonable results, radius should be larger than sigma. Use a radius of 0 and Sharpen selects a suitable radius for you.

channel: The channel type.
radius: The radius of the Gaussian, in pixels, not counting the center pixel.

sigma: The standard deviation of the Laplacian, in pixels.

Constants for channel:

```
const UndefinedChannel    = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7ffffff
```

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.119 Shave(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shave shaves pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.
Returns nil on any error.
Sets the last exception property.

5.21.120 Shear(Xshear as Double, Yshear as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method ShearImage creates a new image that is a shear_image copy of an existing one.
Notes: Shearing slides one edge of an image along the X or Y axis, creating a parallelogram. An X direction shear slides an edge along the X axis, while a Y direction shear slides an edge along the Y axis. The amount of the shear is controlled by a shear angle. For X direction shears, x_shear is measured relative to the Y axis, and similarly, for Y direction shears y_shear is measured relative to the X axis. Empty triangles left over from shearing the image are filled with the color defined by the pixel at location (0,0). ShearImage

allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Xshear and YYshear specify the number of degrees to shear the image.

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.121 Solarize(factor as Double) as boolean

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SolarizeImage produces a 'solarization' effect seen when exposing a photographic film to light during the development process.
Notes: factor: An double value that defines the extent of the solarization.
Returns nil on any error.
Sets the last exception property.

5.21.122 Splice(x as Integer, y as Integer, width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Splice splices a solid color into the image as defined by the geometry.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.123 Spread(radius as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: This is a special effects method that randomly displaces each pixel in a block defined by the radius parameter.
Notes: radius: Choose a random pixel in a neighborhood of this extent.
Sets the last exception property.

For more details please check the ImageMagick documentation.

5.21.124 Stegano(watermarkImage as IMImageQ8MBS) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SteganoImage hides a digital watermark within the image.
Notes: Returns nil on any error.
Sets the last exception property.

5.21.125 Stereo(otherImage as IMImageQ8MBS) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method StereoImage combines two images and produces a single image that is the composite of a left and right image of a stereo pair.
Notes: The left image is converted to gray scale and written to the red channel of the stereo image. The right image is converted to gray scale and written to the blue channel of the stereo image. View the composite image with red-blue glasses to create a stereo effect.

left image = self
right image = otherImage parameter

Returns nil on any error.
Sets the last exception property.

5.21.126 Swirl(degrees as Double) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method SwirlImage creates a new image that is a copy of an existing one with the image pixels ”swirl” at a specified angle.
Notes: degrees: An double value that defines the tightness of the swirling.

Returns nil on any error.
Sets the last exception property.

5.21.127 Thumbnail(width as Integer, height as Integer) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Notes: Sets the last exception property.

This method was designed by Bob Friesenhahn as a low cost thumbnail generator.
For more details please check the ImageMagick documentation.

5.21.128 TransformImage(CropGeometry as string, ImageGeometry as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransformImage() is a convenience method that behaves like ResizeImage() or CropImage() but accepts scaling and/or cropping information as a region geometry specification. If the operation fails, the original image handle is left as is.
Notes: This should only be used for single images.

CropGeometry: A crop geometry string. This geometry defines a subregion of the image to crop.
ImageGeometry: An image geometry string. This geometry defines the final size of the image.

Returns true on success.

5.21.129 TransformImages(CropGeometry as string, ImageGeometry as string) as boolean

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransformImages() calls TransformImage() on each image of a sequence.
Notes: TransformImage() is a convenience method that behaves like ResizeImage() or CropImage() but accepts scaling and/or cropping information as a region geometry specification. If the operation fails, the original image handle is left as is.

CropGeometry: A crop geometry string. This geometry defines a subregion of the image to crop.
ImageGeometry: An image geometry string. This geometry defines the final size of the image.

Returns true on success.

5.21.130 TransformRGBImage(Colorspace as Integer) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method TransformRGBImage converts the reference image from an alternate colorspace.
Notes: The transformation matrices are not the standard ones: the weights are rescaled to normalized the range of the transformed values to be [0..MaxRGB] .

colorspace: An integer value that indicates the colorspace the image is currently in. On return the image is in the RGB color space.

Returns false on any error and true on success.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

5.21.131 TransposeImage as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransposeImage() creates a horizontal mirror image by reflecting the pixels around the central y-axis while rotating them by 90 degrees.

5.21.132 TransverseImage as IMImageQ8MBS

Plugin Version: 11.3, Platforms: macOS, Linux, Windows, Targets: All.

Function: TransverseImage() creates a vertical mirror image by reflecting the pixels around the central x-axis while rotating them by 270 degrees.

5.21.133 Trim as IMImageQ8MBS

Plugin Version: 6.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Trim trims pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Returns nil on any error.
Sets the last exception property.

5.21.134 UnsharpMaskChannel(channel as Integer, radius as Double, sigma as Double, amount as Double, threshold as Double) as IMImageQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and UnsharpMask selects a suitable radius for you.

Constants for channel:

```
const UndefinedChannel   = 0
 const RedChannel         = &h0001
 const GrayChannel        = &h0001
 const CyanChannel        = &h0001
 const GreenChannel       = &h0002
 const MagentaChannel     = &h0002
 const BlueChannel        = &h0004
 const YellowChannel      = &h0004
 const AlphaChannel       = &h0008
 const OpacityChannel     = &h0008
 const BlackChannel       = &h0020
 const IndexChannel       = &h0020
 const AllChannels        = &h7ffffff
```

Sets the last exception property.
For more details please check the ImageMagick documentation.

5.21.135 Wave(amplitude as Double, wavelength as Double) as IMImageQ8MBS

Plugin Version: 5.4, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method Wave creates a new image that is a copy of an existing one with the image pixels altered along a sine wave.
Notes: Parameters are double values that indicates the amplitude and wavelength of the sine wave.
Returns nil on any error.
Sets the last exception property.

5.21.136 WhiteThreshold(threshold as string) as boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: WhiteThreshold is like Threshold but forces all pixels above the threshold into white while leaving all pixels below the threshold unchanged.
Notes: No exceptions are generated.
threshold: Define the threshold value. (ASCII string)
For more details please check the ImageMagick documentation.

5.21.137 WriteImage(info as IMImageInfoQ8MBS) as boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method WriteImage writes an image to a file as defined by image.filename.
Notes: You can specify a particular image format by prefixing the file with the image type and a colon (i.e. ps:image) or specify the image type as the filename suffix (i.e. image.ps). The image may be modified to adapt it to the requirements of the image format. For example, DirectClass images must be color-reduced to PseudoClass if the format is GIF.

WriteImage returns True if the image is written. False is returned if there is a memory shortage or if the image file fails to write.

5.21.138 Properties

5.21.139 BackgroundColor as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image background color.
Notes: (Read and Write property)

5.21.140 Bias as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.21.141 BlurFactor as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blur factor to apply to the image when zooming. Default is 1.0 (no blur).
Notes: (Read and Write property)

5.21.142 BorderColor as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image border color.
Notes: (Read and Write property)

5.21.143 Colors as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The desired number of colors.
Notes: Used by Quantize().
(Read and Write property)

5.21.144 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

(Read and Write property)

5.21.145 Compression as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image compresion type.
Notes: useful constants:

const UndefinedCompression	= 0
const NoCompression	= 1
const BZipCompression	= 2
const FaxCompression	= 3
const Group4Compression	= 4
const JPEGCompression	= 5
const LosslessJPEGCompression	= 6
const LZWCompression	= 7
const RLECompression	= 8
const ZipCompression	= 9

The default is the compression type of the specified image file.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.146 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: QuantumLeap must be defined before a depth of 16 is valid.
(Read and Write property)

5.21.147 Directory as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile names from within an image montage.
Notes: Only valid after calling MontageImages() or reading a MIFF file which contains a directory.
(Read and Write property)

5.21.148 Endian as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The endian setting to use.
Notes: constants:

UndefinedEndian	0	
LSBEndian	1	(Windows)
MSBEndian	2	(Mac)

e.g. tiff files support different endian settings.
(Read and Write property)

5.21.149 Filename as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path/name.
Notes: The string must be in the encoding of the library and is limited to 4000 bytes.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.150 Filter as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Filter to use when resizing image.
Notes: Constants:

```
const PointFilter        =1
 const BoxFilter         =2
 const TriangleFilter    =3
 const HermiteFilter     =4
 const HanningFilter     =5
 const HammingFilter     =6
 const BlackmanFilter    =7
 const GaussianFilter    =8
 const QuadraticFilter   =9
 const CubicFilter       =10
 const CatromFilter      =11
 const MitchellFilter    =12
 const LanczosFilter     =13
 const BesselFilter      =14
 const SincFilter        =15
```

The reduction filter employed has a sipngicant effect on the time required to resize an image and the resulting quality. The default filter is Lanczos which has been shown to produce high quality results when reducing most images.
(Read and Write property)

5.21.151 Fuzz as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this to match colors that are close to the target color in RGB space.
(Read and Write property)

5.21.152 Gamma as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma level of the image.
Notes: The same color image displayed on two different workstations may look different due to differences

in the display monitor. Use gamma correction to adjust for this color difference.
(Read and Write property)

5.21.153 Geometry as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Preferred size of the image when encoding.
Notes: (Read and Write property)

5.21.154 Gravity as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.21.155 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to an Image structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.156 Height as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the image in pixels.
Notes: For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.157 Interlace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of interlacing scheme (default NoInterlace).
Notes: This option is used to specify the type of interlacing scheme for raw image formats such as RGB

or YUV. NoInterlace means do not interlace, LineInterlace uses scanline interlacing, and PlaneInterlace uses plane interlacing. PartitionInterlace is like PlaneInterlace except the different planes are saved to individual files (e.g. image.R, image.G, and image.B). Use LineInterlace or PlaneInterlace to create an interlaced GIF or progressive JPEG image.

constants:

UndefinedInterlace	0	Unset value.
NoInterlace	1	Don't interlace image (RGBRGBRGBRGBRGBRGB...)
LineInterlace	2	Use scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...)
PlaneInterlace	3	Use plane interlacing (RRRRRR...GGGGGG...BBBBBB...)
PartitionInterlace	4	Similar to plane interlaing except that the different planes are saved to individual files (e.g. image.R, image.G, and image.B)

(Read and Write property)

5.21.158 LastError as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code reported.
Notes: If an exception is raised and it is not a warning exception, this exception code is saved in this property.
(Read and Write property)

5.21.159 LastException as IMExceptionQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception thrown by the Image Magick library.
Notes: You should check this value after every call to the library, process the error and set the property to nil.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.160 Magick as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image encoding format (e.g. "GIF").
Notes: (Read and Write property)

5.21.161 Matte as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether an alpha channel is used/present.
Notes: Set to true to enable masks.
(Read and Write property)

5.21.162 MatteColor as IMColorQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image matte (transparent) color.
Notes: (Read and Write property)

5.21.163 Montage as String

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tile size and offset within an image montage. Only valid for montage images.
Notes: (Read and Write property)

5.21.164 Offset as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of initial bytes to skip over when reading raw image.
Notes: (Read and Write property)

5.21.165 Orientation as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image orientation.
Notes: constants:

```
const UndefinedOrientation      = 0
 const TopLeftOrientation       = 1
 const TopRightOrientation      = 2
 const BottomRightOrientation   = 3
 const BottomLeftOrientation    = 4
 const LeftTopOrientation       = 5
 const RightTopOrientation      = 6
 const RightBottomOrientation   = 7
 const LeftBottomOrientation    = 8
```

For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.166 Quality as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG/MIFF/PNG compression level.
Example:

```
dim im as ImageMagickQ8MBS // global

Function TestJPEG(f as folderitem) As picture
// Reads an image, compresses in memory to JPEG, decompresses using JPEGlib and returns the image
// if quality setting works, you see it in the result.
// no error checking included!

// needs: im as ImageMagickQ8MBS ready initialized

dim image as IMImageQ8MBS
dim imageinfo as IMImageInfoQ8MBS
dim s,blob as string
dim p as Picture
dim i as Integer

if f = nil then
Return nil
end if

imageinfo = im.NewImageInfo

imageinfo.Filename = f.NativePath

//this should read any image IM understands
image = im.ReadImage(imageinfo)
//check for error
```

```
if im.lastexception <>nil and im.LastException.Severity >= 400 then
s = "LastError: "+Format(im.LastError,"-0")+" - Severity: "+str(im.LastException.Severity)+EndOfLine+im.LastException.Reason
MsgBox s
Return nil
elseif image = nil then
MsgBox "image=nil"
Return nil
end if

// Now lets convert to jpeg
imageinfo.Filename = "image.jpg"
imageinfo.Quality = 10 // 100 is max
blob = image.ImageToBlob(imageinfo)

// It may fail
if blob.lenb = 0 then
Return nil
end if
p = JPEGStringToPictureMBS(blob,true)

image.DestroyImage
imageinfo.DestroyImageInfo

Return p
Exception
Return nil
End Function
```

Notes: Default value is 75.
(Read and Write property)

5.21.167 Release as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: If true, the destructor will release the handle.
Notes: (Read and Write property)

5.21.168 RenderingIntent as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rendering intent to use.
Notes: constants:

UndefinedIntent	0
SaturationIntent	1
PerceptualIntent	2
AbsoluteIntent	3
RelativeIntent	4

(Read and Write property)

5.21.169 ResolutionUnits as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units of image resolution.
Notes: constants:

UndefinedResolution	0	Unset value.
PixelsPerInchResolution	1	Density specifications are specified in units of pixels per inch (english units).
PixelsPerCentimeterResolution	2	Density specifications are specified in units of pixels per centimeter (metric units).

(Read and Write property)

5.21.170 ResolutionX as Double

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The horizontal resolution of the image.
Notes: The unit for resolution must be specified.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.171 ResolutionY as Double

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The vertical resolution of the image.
Notes: The unit for resolution must be specified.

For more details please check the ImageMagick documentation.
(Read and Write property)

5.21.172 Scene as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: An undocumented property.
Notes: (Read and Write property)

5.21.173 StorageClass as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image storage class.
Notes: If DirectClass then the image packets contain valid RGB or CMYK colors. If PseudoClass then the image has a colormap referenced by pixel's index member.

constants:

UndefinedClass	0	Unset value.
DirectClass	1	Image is composed of pixels which represent literal color values.
PseudoClass	2	Image is composed of pixels which specify an index in a color palette.

(Read and Write property)

5.21.174 Taint as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set to True if the image pixels have been modified.
Notes: (Read and Write property)

5.21.175 Width as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the image in pixels.
Notes: For more details please check the ImageMagick documentation.

(Read and Write property)

5.21.176 Constants

Constants

Constant	Value	Description
kBackgroundDispose	2	One of the Image layer Dispose Types.
kCoalesceLayer	1	One of the Image layer method constants.
kCompareAnyLayer	2	One of the Image layer method constants.
kCompareClearLayer	3	One of the Image layer method constants.
kCompareOverlayLayer	4	One of the Image layer method constants.
kCompositeLayer	12	One of the Image layer method constants.
kDisposeLayer	5	One of the Image layer method constants.
kFlattenLayer	14	One of the Image layer method constants.
kMergeLayer	13	One of the Image layer method constants.
kMosaicLayer	15	One of the Image layer method constants.
kNoneDispose	1	One of the Image layer Dispose Types.
kOptimizeImageLayer	7	One of the Image layer method constants.
kOptimizeLayer	6	One of the Image layer method constants.
kOptimizePlusLayer	8	One of the Image layer method constants.
kOptimizeTransLayer	9	One of the Image layer method constants.
kPreviousDispose	3	One of the Image layer Dispose Types.
kRemoveDupsLayer	10	One of the Image layer method constants.
kRemoveZeroLayer	11	One of the Image layer method constants.
kUndefinedDispose	0	One of the Image layer Dispose Types.
kUndefinedLayer	0	One of the Image layer method constants.
kUnrecognizedDispose	0	One of the Image layer Dispose Types.

Distortion Effects

Constant	Value	Description
kAffineDistortion	1	
kAffineProjectionDistortion	2	
kArcDistortion	9	
kBarrelDistortion	14	
kBarrelInverseDistortion	15	
kBilinearDistortion	6	
kBilinearForwardDistortion	6	
kBilinearReverseDistortion	7	
kCylinder2PlaneDistortion	12	
kDePolarDistortion	11	
kPerspectiveDistortion	4	
kPerspectiveProjectionDistortion	5	
kPlane2CylinderDistortion	13	
kPolarDistortion	10	
kPolynomialDistortion	8	
kResizeDistortion	17	
kScaleRotateTranslateDistortion	3	
kSentinelDistortion	18	
kShepardsDistortion	16	
kUndefinedDistortion	0	

Interpolate Modes

Constant	Value	Description
kBarycentricColorInterpolate	1	
kBilinearColorInterpolate	7	
kInverseColorInterpolate	19	
kPolynomialColorInterpolate	8	
kShepardsColorInterpolate	16	
kUndefinedColorInterpolate	0	
kVoronoiColorInterpolate	18	

5.22 class IMMagickInfoListQ16MBS

5.22.1 class IMMagickInfoListQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class with the list of the image formats supported in Image Magick.
Notes: For more details please check the ImageMagick documentation.

5.22.2 Methods

5.22.3 Item(index as Integer) as IMMagickInfoQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The items inside this list.
Notes: Index goes from 0 to count-1.
Returns nil on invalid index.

5.22.4 Properties

5.22.5 Count as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of items.
Notes: Index goes from 0 to count-1.
(Read only property)

5.22.6 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a MagickInfo list.
For more details please check the ImageMagick documentation.
(Read only property)

5.23 class IMMagickInfoListQ32MBS

5.23.1 class IMMagickInfoListQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class with the list of the image formats supported in Image Magick.
Notes: For more details please check the ImageMagick documentation.

5.23.2 Methods

5.23.3 Item(index as Integer) as IMMagickInfoQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The items inside this list.
Notes: Index goes from 0 to count-1.
Returns nil on invalid index.

5.23.4 Properties

5.23.5 Count as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of items.
Notes: Index goes from 0 to count-1.
(Read only property)

5.23.6 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a MagickInfo list.
For more details please check the ImageMagick documentation.
(Read only property)

5.24 class IMMagickInfoListQ8MBS

5.24.1 class IMMagickInfoListQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class with the list of the image formats supported in Image Magick.
Notes: For more details please check the ImageMagick documentation.

5.24.2 Methods

5.24.3 Item(index as Integer) as IMMagickInfoQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The items inside this list.
Notes: Index goes from 0 to count-1.
Returns nil on invalid index.

5.24.4 Properties

5.24.5 Count as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of items.
Notes: Index goes from 0 to count-1.
(Read only property)

5.24.6 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a MagickInfo list.
For more details please check the ImageMagick documentation.
(Read only property)

5.25 class IMMagickInfoQ16MBS

5.25.1 class IMMagickInfoQ16MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for information about a file import/export format Image Magick can handle.
Notes: For more details please check the ImageMagick documentation.

5.25.2 Methods

5.25.3 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.25.4 Properties

5.25.5 Adjoin as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if this file format supports multi-frame images.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.25.6 BlobSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if the encoder and decoder for this format supports operating on arbitrary BLOBs (rather than only disk files).
Notes: As currently disc read/write does not work with the 5.1 plugins, we really need that to use the classes.
Returns false for an invalid MagickInfo (handle=0).

For more details please check the ImageMagick documentation.

(Read only property)

5.25.7 Description as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Long form image format description (e.g. ”CompuServe graphics interchange format”).
Notes: For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.25.8 EndianSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether endian support is available.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.25.9 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a MagickInfo structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.25.10 ModuleName as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Name of module (e.g. ”GIF”) which registered this format.
Notes: Value is ”” if format is not registered by a module.
For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.25.11 Name as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Magick string (e.g. ”GIF”) which identifies this format.
Notes: For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.25.12 Note as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Additional notes for this format.
Notes: e.g. compilation parameters or copyright notices.
Returns ”” for an invalid MagickInfo (handle=0).

For more details please check the ImageMagick documentation.
(Read only property)

5.25.13 Raw as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if Image format does not contain size (must be specified in ImageInfo).
Notes: Returns false for an invalid MagickInfo (handle=0).
For more details please check the ImageMagick documentation.
(Read only property)

5.25.14 SeekableStream as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns True if the magick supports a seekable stream.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.25.15 Stealth as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.25.16 ThreadSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if threading is supported.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.25.17 Version as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Version string.
Notes: For more details please check the ImageMagick documentation.
Returns "" for an invalid MagickInfo (handle=0).
(Read only property)

5.26 class IMMagickInfoQ32MBS

5.26.1 class IMMagickInfoQ32MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for information about a file import/export format Image Magick can handle.
Notes: For more details please check the ImageMagick documentation.

5.26.2 Methods

5.26.3 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.26.4 Properties

5.26.5 Adjoin as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if this file format supports multi-frame images.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.26.6 BlobSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if the encoder and decoder for this format supports operating on arbitrary BLOBs (rather than only disk files).
Notes: As currently disc read/write does not work with the 5.1 plugins, we really need that to use the classes.
Returns false for an invalid MagickInfo (handle=0).

For more details please check the ImageMagick documentation.

(Read only property)

5.26.7 Description as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Long form image format description (e.g. "CompuServe graphics interchange format").
Notes: For more details please check the ImageMagick documentation.
Returns "" for an invalid MagickInfo (handle=0).
(Read only property)

5.26.8 EndianSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether endian support is available.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.26.9 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a MagickInfo structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.26.10 ModuleName as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Name of module (e.g. "GIF") which registered this format.
Notes: Value is "" if format is not registered by a module.
For more details please check the ImageMagick documentation.
Returns "" for an invalid MagickInfo (handle=0).
(Read only property)

5.26.11 Name as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Magick string (e.g. "GIF") which identifies this format.
Notes: For more details please check the ImageMagick documentation.
Returns "" for an invalid MagickInfo (handle=0).
(Read only property)

5.26.12 Note as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Additional notes for this format.
Notes: e.g. compilation parameters or copyright notices.
Returns "" for an invalid MagickInfo (handle=0).

For more details please check the ImageMagick documentation.
(Read only property)

5.26.13 Raw as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if Image format does not contain size (must be specified in ImageInfo).
Notes: Returns false for an invalid MagickInfo (handle=0).
For more details please check the ImageMagick documentation.
(Read only property)

5.26.14 SeekableStream as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns True if the magick supports a seekable stream.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.26.15 Stealth as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.26.16 ThreadSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if threading is supported.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.26.17 Version as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Version string.
Notes: For more details please check the ImageMagick documentation.
Returns "" for an invalid MagickInfo (handle=0).
(Read only property)

5.27 class IMMagickInfoQ8MBS

5.27.1 class IMMagickInfoQ8MBS

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: A class for information about a file import/export format Image Magick can handle.
Notes: For more details please check the ImageMagick documentation.

5.27.2 Methods

5.27.3 Close

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The destructor.
Notes: There is no need to call this method except you want to free all resources used by this object now without waiting for Xojo to do it for you.

5.27.4 Properties

5.27.5 Adjoin as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if this file format supports multi-frame images.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.27.6 BlobSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if the encoder and decoder for this format supports operating on arbitrary BLOBs (rather than only disk files).
Notes: As currently disc read/write does not work with the 5.1 plugins, we really need that to use the classes.
Returns false for an invalid MagickInfo (handle=0).

For more details please check the ImageMagick documentation.

(Read only property)

5.27.7 Description as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Long form image format description (e.g. ”CompuServe graphics interchange format”).
Notes: For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.27.8 EndianSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether endian support is available.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.27.9 Handle as Integer

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to a MagickInfo structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.27.10 ModuleName as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Name of module (e.g. ”GIF”) which registered this format.
Notes: Value is ”” if format is not registered by a module.
For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.27.11 Name as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Magick string (e.g. ”GIF”) which identifies this format.
Notes: For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.27.12 Note as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Additional notes for this format.
Notes: e.g. compilation parameters or copyright notices.
Returns ”” for an invalid MagickInfo (handle=0).

For more details please check the ImageMagick documentation.
(Read only property)

5.27.13 Raw as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if Image format does not contain size (must be specified in ImageInfo).
Notes: Returns false for an invalid MagickInfo (handle=0).
For more details please check the ImageMagick documentation.
(Read only property)

5.27.14 SeekableStream as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns True if the magick supports a seekable stream.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.27.15 Stealth as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unknown.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.27.16 ThreadSupport as Boolean

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: True if threading is supported.
Notes: For more details please check the ImageMagick documentation.
Returns false for an invalid MagickInfo (handle=0).
(Read only property)

5.27.17 Version as String

Plugin Version: 5.1, Platforms: macOS, Linux, Windows, Targets: All.

Function: Version string.
Notes: For more details please check the ImageMagick documentation.
Returns ”” for an invalid MagickInfo (handle=0).
(Read only property)

5.28 class IMMagickPixelPacketQ16MBS

5.28.1 class IMMagickPixelPacketQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class to describe a picture background.
Notes: Needed for IMImageQ16MBS.NewImage function.

5.28.2 Methods

5.28.3 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole ImageInfo structure copied into a memoryblock.
Notes: Returns nil on any error.

5.28.4 Properties

5.28.5 Blue as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color value.
Notes: (Read and Write property)

5.28.6 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

(Read and Write property)

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

5.28.7 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: (Read and Write property)

5.28.8 Fuzz as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this to match colors that are close to the target color in RGB space.
(Read and Write property)

5.28.9 Green as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color value.
Notes: (Read and Write property)

5.28.10 Handle as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to an MagickPixelPacket structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.28.11 Index as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The index color value.
Notes: Only for indexed color spaces.
(Read and Write property)

5.28.12 Matte as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether an alpha channel is used/present.
Notes: Set to true to enable masks.
(Read and Write property)

5.28.13 Opacity as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity part of the color value.
Notes: (Read and Write property)

5.28.14 Red as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color value.
Notes: (Read and Write property)

5.29 class IMMagickPixelPacketQ32MBS

5.29.1 class IMMagickPixelPacketQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class to describe a picture background.
Notes: Needed for IMImageQ32MBS.NewImage function.

5.29.2 Methods

5.29.3 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole ImageInfo structure copied into a memoryblock.
Notes: Returns nil on any error.

5.29.4 Properties

5.29.5 Blue as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color value.
Notes: (Read and Write property)

5.29.6 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

(Read and Write property)

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

5.29.7 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: (Read and Write property)

5.29.8 Fuzz as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this to match colors that are close to the target color in RGB space.
(Read and Write property)

5.29.9 Green as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color value.
Notes: (Read and Write property)

5.29.10 Handle as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to an MagickPixelPacket structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.29.11 Index as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The index color value.
Notes: Only for indexed color spaces.
(Read and Write property)

5.29.12 Matte as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether an alpha channel is used/present.
Notes: Set to true to enable masks.
(Read and Write property)

5.29.13 Opacity as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity part of the color value.
Notes: (Read and Write property)

5.29.14 Red as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color value.
Notes: (Read and Write property)

5.30 class IMMagickPixelPacketQ8MBS

5.30.1 class IMMagickPixelPacketQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class to describe a picture background.
Notes: Needed for IMImageQ8MBS.NewImage function.

5.30.2 Methods

5.30.3 HandleMemory as memoryblock

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The content of the whole ImageInfo structure copied into a memoryblock.
Notes: Returns nil on any error.

5.30.4 Properties

5.30.5 Blue as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color value.
Notes: (Read and Write property)

5.30.6 ColorSpace as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image pixel interpretation.
Notes: If the colorspace is RGB the pixels are red, green, blue. If matte is true, then red, green, blue, and index. If it is CMYK, the pixels are cyan, yellow, magenta, black. Otherwise the colorspace is ignored.

constants:

(Read and Write property)

UndefinedColorspace	0
RGBColorspace	1
GRAYColorspace	2
TransparentColorspace	3
OHTAColorspace	4
LABColorspace	5
XYZColorspace	6
YCbCrColorspace	7
YCCColorspace	8
YIQColorspace	9
YPbPrColorspace	10
YUVColorspace	11
CMYKColorspace	12
sRGBColorspace	13
HSBColorspace	14
HSLColorspace	15
HWBColorspace	16

5.30.7 Depth as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image depth (8 or 16).
Notes: (Read and Write property)

5.30.8 Fuzz as Double

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colors within this distance are considered equal.
Notes: A number of algorithms search for a target color. By default the color must be exact. Use this to match colors that are close to the target color in RGB space.
(Read and Write property)

5.30.9 Green as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color value.
Notes: (Read and Write property)

5.30.10 Handle as Integer

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The handle used internally by the plugin.
Notes: A pointer to an MagickPixelPacket structure.
For more details please check the ImageMagick documentation.
(Read and Write property)

5.30.11 Index as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The index color value.
Notes: Only for indexed color spaces.
(Read and Write property)

5.30.12 Matte as Boolean

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether an alpha channel is used/present.
Notes: Set to true to enable masks.
(Read and Write property)

5.30.13 Opacity as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The opacity part of the color value.
Notes: (Read and Write property)

5.30.14 Red as Single

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color value.
Notes: (Read and Write property)

5.31 class IMMissingFunctionExceptionQ16MBS

5.31.1 class IMMissingFunctionExceptionQ16MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: A class for an exception in Image Magick.
Notes: This exception is raised on every IM function if the library function behind is not available.
(this can be a plugin bug or a bad compiled library or simply a too old library.)
Subclass of the RuntimeException class.

5.32 class IMMissingFunctionExceptionQ32MBS

5.32.1 class IMMissingFunctionExceptionQ32MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: A class for an exception in Image Magick.
Notes: This exception is raised on every IM function if the library function behind is not available.
(this can be a plugin bug or a bad compiled library or simply a too old library.)
Subclass of the RuntimeException class.

5.33 class IMMissingFunctionExceptionQ8MBS

5.33.1 class IMMissingFunctionExceptionQ8MBS

Plugin Version: 5.2, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: A class for an exception in Image Magick.
Notes: This exception is raised on every IM function if the library function behind is not available.
(this can be a plugin bug or a bad compiled library or simply a too old library.)
Subclass of the RuntimeException class.

Chapter 6

ImageMagick7

6.1 module ImageMagick7MBS

6.1.1 module ImageMagick7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The module for global functions.
Notes: The MBS Plugin can load 8, 16, 32 or 64 bit ImageMagick libraries with or without HDRI support. We detect what library is loaded and adjust in various functions. Check QuantumDepth, QuantumSize and HDRI properties.

All class names have currently 7 in the class name to give users of the older classes time for the transition. Currently we use ImageMagick 7.0.9. The plugin may work with older/newer versions of 7.x.

We do have functions to use Xojo pictures with CopyPicture and SetPicture. Those move pixels as they are, so it is up to you to do proper color management. As Xojo pictures are just 8bit RGB, you may loose information on conversion. ExportPixels and ImportPixels functions can move pixel data into/from memory blocks.

Best to avoid Xojo pictures and load image here, modify it and save it again.
Blog Entries

- MBS Xojo Plugins, version 19.6pr1

6.1.2 Methods

6.1.3 ClampToQuantum(value as Double) as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clamps given value to the range for quantum.
Notes: Limits range between 0 and QuantumRange and for non-HDRI rounds value.

6.1.4 InitializeMagick(path as string = "")

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the MagickCore environment.
Notes: path: the execution path of the current ImageMagick client.

See MagickCoreGenesis function in ImageMagick documentation.

6.1.5 LoadLibrary(path as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the MagickCore library file.
Example:

```
#If TargetMacOS Then

// installed via homebrew
Dim f As New FolderItem("/opt/homebrew/Cellar/imagemagick/7.1.1-6/lib/libMagickCore-7.Q16HDRI.10.dylib",
FolderItem.PathModes.Native)

If ImageMagick7MBS.LoadLibraryFile(f) Then

'MsgBox "loaded"

Else

MsgBox "failed to load: "+ImageMagick7MBS.LoadErrorString

End If

#Else
If ImageMagick7MBS.LoadLibrary("CORE_RL_MagickCore_.dll") Then
```

```
'MsgBox "loaded"

Else

MsgBox "failed to load: "+ImageMagick7MBS.LoadErrorString

End If
#EndIf
```

Notes: Returns true on success and false on failure.
LoadErrorString is set with error string if available.

6.1.6 LoadLibraryFile(path as folderitem) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Loads the MagickCore library file.
Example:

```
Dim f As FolderItem = SpecialFolder.Desktop.Child("libMagickCore-7.Q16HDRI.7.dylib")

If ImageMagick7MBS.LoadLibraryFile(f) Then

MsgBox "loaded"

Else

MsgBox "failed to load: "+ImageMagick7MBS.LoadErrorString

End If
```

Notes: Returns true on success and false on failure.
LoadErrorString is set with error string if available.

6.1.7 MagickInfoList as IMMagickInfoList7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries list of MagickInfo objects.
Notes: See GetMagickInfoList function in ImageMagick documentation.

6.1.8 MagickToMime(name as string) as string

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the officially registered (or de facto) MIME media-type corresponding to a magick string.
Notes: If there is no registered media-type, then the string ”image/x-magick” (all lower case) is returned.

See MagickToMime function in ImageMagick documentation.

6.1.9 NewImageInfo as IMImageInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates new image info object.
Notes: See CloneImageInfo function in ImageMagick documentation.

6.1.10 NewImageList as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty image list.
Notes: See NewImageList function in ImageMagick documentation.

6.1.11 PrintMagickVersion

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Identifies the ImageMagick version by printing its attributes to the file. Attributes include the copyright, features, and delegates.
Notes: See ListMagickVersion function in ImageMagick documentation.

6.1.12 ScaleQuantumToChar(value as Double) as UInt8

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales a quantum value to 8 bit unsigned integer.
Notes: Depending on which library is loaded a quantum is an UInt8, UInt16, UInt32, Single or Double value.
This function normalizes and rounds values to nearest 8 bit integer value.

6.1.13 SetCurrentDirectory(path as folderitem) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the current working directory.
Notes: This is needed for most installations to point to the folder with the libraries in order for LoadLibrary to find the dependencies.

6.1.14 Properties

6.1.15 Copyright as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick API copyright as a string.
Notes: See GetMagickCopyright function in ImageMagick documentation.
(Read only property)

6.1.16 Delegates as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick delegate libraries.
Notes: See GetMagickDelegates function in ImageMagick documentation.
(Read only property)

6.1.17 Epsilon as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Small value.
Notes: Two values are considered equal if the difference is smaller than epsilon.
(Read only property)

6.1.18 Features as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick features.
Example:

```
MsgBox ImageMagick7MBS.Features
```

Notes: See GetMagickFeatures function in ImageMagick documentation.
e.g. "Cipher DPC Modules OpenCL OpenMP(2.0)".
(Read only property)

6.1.19 HDRI as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether HDRI is used.
Notes: If the plugin detects HDRI mode for the library, we use floats (8 and 16 bit depth) or doubles (32 and 64 bit depth).
(Read only property)

6.1.20 HomeURL as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick home URL.
Notes: See GetMagickHomeURL function in ImageMagick documentation.
(Read only property)

6.1.21 Huge as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Huge value.
Notes: Values bigger than Huge are far over the range of a quantum.
(Read only property)

6.1.22 License as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick API license as a string.
Notes: See GetMagickLicense function in ImageMagick documentation.
(Read only property)

6.1.23 LoadErrorString as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error from LoadLibrary call.
Notes: (Read only property)

6.1.24 MagickPrecision as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the maximum number of significant digits to be printed.
Notes: See GetMagickPrecision function in ImageMagick documentation.
(Read and Write property)

6.1.25 MagickSignature as UInt32

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a signature that uniquely encodes the MagickCore libary version, quantum depth, HDRI status, OS word size, and endianness.
Notes: See GetMagickSignature function in ImageMagick documentation.
(Read only property)

6.1.26 MaxColormapSize as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The maximum size of a color map.
Notes: Value is 256 for 8 bit or 65536 for higher bit depths.
(Read only property)

6.1.27 MaxMap as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The maximum index in a color map.
Notes: Either 255 for 8 bit or 65535 for 16 bit or higher.
(Read only property)

6.1.28 PackageName as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick package name.
Notes: See GetMagickPackageName function in ImageMagick documentation.
(Read only property)

6.1.29 QuantumDepth as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The quantum depth.
Notes: Can be 8, 16, 32 or 64 bit depending on the loaded image magick library.
(Read only property)

6.1.30 QuantumDepthString as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick quantum depth.
Notes: Returns the quantum depth is returned as a number.
See GetMagickQuantumDepth function in ImageMagick documentation.
(Read only property)

6.1.31 QuantumRange as UInt32

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The maximum value for the range of quantums.
Notes: Depending on quantum depth ranges from 255 to &hFFFFFFFF.
(Read only property)

6.1.32 QuantumRangeString as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick quantum range.
Notes: See GetMagickQuantumRange function in ImageMagick documentation.
e.g. "((Quantum) 65535)"
(Read only property)

6.1.33 QuantumSize as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The size of a quantum value in bytes.
Notes: Used to allocate memoryblocks of right size.
e.g. Width * Height * QuantumSize * ChannelCount.
(Read only property)

6.1.34 ReleaseDate as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick release date.
Notes: See GetMagickReleaseDate function in ImageMagick documentation.
(Read only property)

6.1.35 Version as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick API version as a string.
Example:

```
MsgBox ImageMagick7MBS.Version
// e.g. "ImageMagick 7.0.9-2 Q16 x86 2019-10-30 http://www.imagemagick.org"
```

Notes: See GetMagickVersion function in ImageMagick documentation.
(Read only property)

6.1.36 VersionNumber as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the ImageMagick API version as a number.
Notes: See GetMagickVersion function in ImageMagick documentation.
e.g. &h709 for 7.0.9.
(Read only property)

6.2 class IMChannelStatistics7MBS

6.2.1 class IMChannelStatistics7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for channel statistics.
Notes: Obtain image statistics. Statistics are normalized to the range of 0.0 to 1.0 and are output into this class.
This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

6.2.2 Methods

6.2.3 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The private constructor.

6.2.4 Properties

6.2.5 Area as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Area.
Notes: (Read only property)

6.2.6 Depth as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The depth.
Notes: (Read only property)

6.2.7 Entropy as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Entropy.
Notes: (Read only property)

6.2.8 Kurtosis as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Kurtosis
Notes: (Read only property)

6.2.9 Maxima as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Maximum value observed.
Notes: (Read only property)

6.2.10 Mean as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Average (mean) value observed.
Notes: (Read only property)

6.2.11 Minima as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Minimum value observed.
Notes: (Read only property)

6.2.12 Skewness as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Skewness.
Notes: (Read only property)

6.2.13 StandardDeviation as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Standard deviation, sqrt(variance)
Notes: (Read only property)

6.2.14 Sum as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sum.
Notes: (Read only property)

6.2.15 SumCubed as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sum cubed
Notes: (Read only property)

6.2.16 SumFourthPower as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sum fourth power.
Notes: (Read only property)

6.2.17 SumSquared as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sum squared.
Notes: (Read only property)

6.2.18 Variance as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Variance.
Notes: (Read only property)

6.3 class IMException7MBS

6.3.1 class IMException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an exception.
Notes: Subclass of the RuntimeException class.

6.3.2 Properties

6.3.3 Description as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The description text.
Notes: (Read and Write property)

6.3.4 Reason as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The reason text.
Notes: (Read and Write property)

6.3.5 Severity as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The error code for exception.
Notes: (Read and Write property)

6.3.6 Constants

Error Types

Constant	Value	Description
BlobError	435	
BlobFatalError	735	
BlobWarning	335	
CacheError	445	
CacheFatalError	745	
CacheWarning	345	
CoderError	450	
CoderFatalError	750	
CoderWarning	350	
ConfigureError	495	
ConfigureFatalError	795	
ConfigureWarning	395	
CorruptImageError	425	
CorruptImageFatalError	725	
CorruptImageWarning	325	
DelegateError	415	
DelegateFatalError	715	
DelegateWarning	315	
DrawError	460	
DrawFatalError	760	
DrawWarning	360	
ErrorException	400	
FatalErrorException	700	
FileOpenError	430	
FileOpenFatalError	730	
FileOpenWarning	330	
FilterError	452	
FilterFatalError	752	
FilterWarning	352	
ImageError	465	
ImageFatalError	765	
ImageWarning	365	
MissingDelegateError	420	
MissingDelegateFatalError	720	
MissingDelegateWarning	320	
ModuleError	455	
ModuleFatalError	755	
ModuleWarning	355	
MonitorError	485	
MonitorFatalError	785	
MonitorWarning	385	
OptionError	410	
OptionFatalError	710	
OptionWarning	310	
PolicyError	499	
PolicyFatalError	799	
PolicyWarning	399	
RandomError	475	
RandomFatalError	775	
RandomWarning	375	
RegistryError	490	
RegistryFatalError	790	
RegistryWarning	390	
ResourceLimitError	400	
ResourceLimitFatalError	700	
ResourceLimitWarning	300	
StreamError	440	
StreamFatalError	740	

6.4 class IMFrameInfo7MBS

6.4.1 class IMFrameInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for a frame.

6.4.2 Methods

6.4.3 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the point object with zero values.
See also:

- 6.4.4 Constructor(X as Integer, Y as Integer, Width as Integer, Heigth as Integer) 904

6.4.4 Constructor(X as Integer, Y as Integer, Width as Integer, Heigth as Integer)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the point object with given values.
See also:

- 6.4.3 Constructor 904

6.4.5 Properties

6.4.6 Height as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the rectangle.
Notes: (Read and Write property)

6.4.7 InnerBevel as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The inner bevel size.
Notes: (Read and Write property)

6.4.8 OuterBevel as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The outer bevel size.
Notes: (Read and Write property)

6.4.9 Width as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the rectangle.
Notes: (Read and Write property)

6.4.10 X as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x coordinate of the rectangle.
Notes: (Read and Write property)

6.4.11 Y as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y coordinate of the rectangle.
Notes: (Read and Write property)

6.5 class IMGeometryInfo7MBS

6.5.1 class IMGeometryInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for geometry info.

6.5.2 Methods

6.5.3 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

6.5.4 Properties

6.5.5 chi as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The chi value.
Notes: (Read and Write property)

6.5.6 psi as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The psi value.
Notes: (Read and Write property)

6.5.7 rho as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rho value.
Notes: (Read and Write property)

6.5.8 sigma as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The sigma value.
Notes: (Read and Write property)

6.5.9 xi as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The xi value.
Notes: (Read and Write property)

6.6 class IMImage7MBS

6.6.1 class IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an ImageMagick image.
Notes: See xx function in ImageMagick documentation.

6.6.2 Methods

6.6.3 AcquireImageColormap(count as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Allocates an image colormap and initializes it to a linear gray colorspace.
Notes: If the image already has a colormap, it is replaced. AcquireImageColormap() returns true if successful, otherwise false if there is not enough memory.

colors: the number of colors in the image colormap.

See AcquireImageColormap function in ImageMagick documentation.

6.6.4 AdaptiveBlur(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adaptively blurs the image by blurring less intensely near image edges and more intensely far from edges.
Notes: We blur the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and AdaptiveBlur selects a suitable radius for you.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Laplacian, in pixels.

See AdaptiveBlurImage function in ImageMagick documentation.

6.6.5 AdaptiveResize(columns as Integer, Rows as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adaptively resize image with pixel resampling.
Notes: This is shortcut function for a fast interpolative resize using mesh interpolation. It works well for small resizes of less than +/- 50 of the original image size. For larger resizing on images a full filtered and slower resize function should be used instead.

columns: the number of columns in the resized image.
rows: the number of rows in the resized image.

See AdaptiveResizeImage function in ImageMagick documentation.

6.6.6 AdaptiveSharpen(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adaptively sharpens the image by sharpening more intensely near image edges and less intensely far from edges.
Notes: We sharpen the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and AdaptiveSharpen selects a suitable radius for you.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Laplacian, in pixels.

See AdaptiveSharpenImage function in ImageMagick documentation.

6.6.7 AdaptiveThreshold(width as Integer, height as integer, bias as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Selects an individual threshold for each pixel based on the range of intensity values in its local neighborhood.
Notes: This allows for thresholding of an image whose global intensity histogram doesn't contain distinctive peaks.

width: the width of the local neighborhood.
height: the height of the local neighborhood.

bias: the mean bias.

See AdaptiveThresholdImage function in ImageMagick documentation.

6.6.8 AddNoise(NoiseType as integer, value as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds random noise to the image.
Notes: channel: the channel type.
noise_type: The type of noise: Uniform, Gaussian, Multiplicative, Impulse, Laplacian, or Poisson.
attenuate: attenuate the random distribution.

See AddNoiseImage function in ImageMagick documentation.

6.6.9 AffineTransform(matrix as IMImageAffineMatrix7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transforms an image as dictated by the affine matrix.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

See xx function in ImageMagick documentation.

6.6.10 AppendImageToList(img as IMImage7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Appends the second image list to the end of the first list. The given image list pointer is left unchanged, unless it was empty.
Notes: See xx function in ImageMagick documentation.

6.6.11 AuthenticPixels(X as Integer, Y as Integer, Width as Integer, Height as Integer) as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Obtains a pixel region for read/write access.

Notes: If the region is successfully accessed, a pointer to a Quantum array representing the region is returned, otherwise NULL is returned.

The returned pointer may point to a temporary working copy of the pixels or it may point to the original pixels in memory. Performance is maximized if the selected region is part of one row, or one or more full rows, since then there is opportunity to access the pixels in-place (without a copy) if the image is in memory, or in a memory-mapped file. The returned pointer must *never* be deallocated by the user.

Pixels accessed via the returned pointer represent a simple array of type Quantum. If the image has corresponding metacontent,call GetAuthenticMetacontent() after invoking GetAuthenticPixels() to obtain the meta-content corresponding to the region. Once the Quantum array has been updated, the changes must be saved back to the underlying image using SyncAuthenticPixels() or they may be lost.

See xx function in ImageMagick documentation.

6.6.12 AutoGamma as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extract the 'mean' from the image and adjust the image to try make set its gamma appropriatally.
Notes: See AutoGammaImage function in ImageMagick documentation.

6.6.13 AutoLevel as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adjusts the levels of a particular image channel by scaling the minimum and maximum values to the full quantum range.
Notes: See AutoLevelImage function in ImageMagick documentation.

6.6.14 AutoOrient(OrientationType as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adjusts an image so that its orientation is suitable for viewing (i.e. top-left orientation).
Notes: orientation: Current image orientation.

See AutoOrientImage function in ImageMagick documentation.

6.6.15 AutoThreshold(autoThresholdMethod as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Automatically performs image thresholding dependent on which method you specify.
Notes: method: choose from Kapur, OTSU, or Triangle.

See AutoThresholdImage function in ImageMagick documentation.

6.6.16 Bilevel(threshold as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the value of individual pixels based on the intensity of each pixel channel.
Notes: The result is a high-contrast image.

More precisely each channel value of the image is ’thresholded’ so that if it is equal to or less than the given value it is set to zero, while any value greater than that give is set to it maximum or QuantumRange.

This function is what is used to implement the ”-threshold” operator for the command line API.

If the default channel setting is given the image is thresholded using just the gray ’intensity’ of the image, rather than the individual channels.

threshold: define the threshold values.

See BilevelImage function in ImageMagick documentation.

6.6.17 BlackThreshold(thresholds as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Is like Threshold but forces all pixels below the threshold into black while leaving all pixels at or above the threshold unchanged.
Notes: threshold: define the threshold value.

See BlackThresholdImage function in ImageMagick documentation.

6.6.18 BlueShift(factor as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Mutes the colors of the image to simulate a scene at nighttime in the moonlight.
Notes: factor: the shift factor.

See BlueShiftImage function in ImageMagick documentation.

6.6.19 Blur(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, the radius should be larger than sigma. Use a radius of 0 and Blur selects a suitable radius for you.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.

See BlurImage function in ImageMagick documentation.

6.6.20 Border(rectangle as IMRectangleInfo7MBS, CompositeOperator as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Surrounds the image with a border of the color defined by the bordercolor member of the image structure.
Notes: The width and height of the border are defined by the corresponding members of the rectangle.

rectangle: define the width and height of the border.
CompositeOperator: the composite operator.

See BorderImage function in ImageMagick documentation.

6.6.21 BrightnessContrast(brightness as double, contrast as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the brightness and/or contrast of an image.
Notes: It converts the brightness and contrast parameters into slope and intercept and calls a polynomical function to apply to the image.

brightness: the brightness percent (-100 .. 100).
contrast: the contrast percent (-100 .. 100).

See BrightnessContrastImage function in ImageMagick documentation.

6.6.22 Charcoal(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image that is a copy of an existing one with the edge highlighted.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

radius: the radius of the pixel neighborhood.
sigma: the standard deviation of the Gaussian, in pixels.

See CharcoalImage function in ImageMagick documentation.

6.6.23 Chop(rect as IMRectangleInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes a region of an image and collapses the image to occupy the removed portion.
Notes: rect: Define the region of the image to chop.

See ChopImage function in ImageMagick documentation.

6.6.24 CLAHE(Width as Integer, Height as Integer, NumberBins as Integer, ClipLimit as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Is a variant of adaptive histogram equalization in which the contrast amplification is limited, so as to reduce this problem of noise amplification.
Notes: width: the width of the tile divisions to use in horizontal direction.
height: the height of the tile divisions to use in vertical direction.

NumberBins: number of bins for histogram (”dynamic range”).
ClipLimit: contrast limit for localised changes in contrast. A limit less than 1 results in standard non-contrast limited AHE.

See CLAHEImage function in ImageMagick documentation.

6.6.25 Clamp as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clamps image pixels.
Notes: Set each pixel whose value is below zero to zero and any the pixel whose value is above the quantum range to the quantum range (e.g. 65535) otherwise the pixel value remains unchanged.

See xx function in ImageMagick documentation.

6.6.26 ClampToQuantum(value as Double) as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clamps given value to the range for quantum.
Notes: Limits range between 0 and QuantumRange and for non-HDRI rounds value.

6.6.27 Clip as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clips along the first path from the 8BIM profile, if present.
Notes: See ClipImage function in ImageMagick documentation.

6.6.28 Clone as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies an image and returns the copy as a new image object.
Notes: See CloneImage function in ImageMagick documentation.

6.6.29 CloneImageProfiles(SourceImage as IMImage7MBS) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones one or more image profiles.
Notes: See xx function in ImageMagick documentation.

6.6.30 CloneProperties(Source as IMImage7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones all the image properties from source image to current image.
Notes: See CloneImageProperties function in ImageMagick documentation.

6.6.31 Clut(clutImage as IMImage7MBS, pixelInterpolateMethod as integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces each color value in the given image, by using it as an index to lookup a replacement color value in a Color Look UP Table in the form of an image.
Notes: The values are extracted along a diagonal of the CLUT image so either a horizontal or vertial gradient image can be used.

Typically this is used to either re-color a gray-scale image according to a color gradient in the CLUT image, or to perform a freeform histogram (level) adjustment according to the (typically gray-scale) gradient in the CLUT image.

When the 'channel' mask includes the matte/alpha transparency channel but one image has no such channel it is assumed that that image is a simple gray-scale image that will effect the alpha channel values, either for gray-scale coloring (with transparent or semi-transparent colors), or a histogram adjustment of existing alpha channel values. If both images have matte channels, direct and normal indexing is applied, which is rarely used.

clutImage: the color lookup table image for replacement color values.
method: the pixel interpolation method.

See ClutImage function in ImageMagick documentation.

6.6.32 CoalesceImages as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composites a set of images while respecting any page offsets and disposal methods.
Notes: GIF, MIFF, and MNG animation sequences typically start with an image background and each subsequent image varies in size and offset. A new image sequence is returned with all images the same size as the first images virtual canvas and composited with the next image in the sequence.

See CoalesceImages function in ImageMagick documentation.

6.6.33 ColorDecisionList(ColorCorrectionCollection as string) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Accepts a lightweight Color Correction Collection (CCC) file which solely contains one or more color corrections and applies the correction to the image.
Notes: Here is a sample CCC file:

```
<ColorCorrectionCollection xmlns=”urn:ASC:CDL:v1.2”>
<ColorCorrection id=”cc03345”>
<SOPNode>
<Slope>0.9 1.2 0.5 </Slope>
<Offset>0.4 -0.5 0.6 </Offset>
<Power>1.0 0.8 1.5 </Power>
</SOPNode>
<SATNode>
<Saturation>0.85 </Saturation>
</SATNode>
</ColorCorrection>
</ColorCorrectionCollection>
```

which includes the slop, offset, and power for each of the RGB channels as well as the saturation.

ColorCorrectionCollection: the color correction collection in XML.

See ColorDecisionListImage function in ImageMagick documentation.

6.6.34 Colorize(opacity as string, pixelInfo as IMPixelInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blends the fill color with each pixel in the image.
Notes: A percentage blend is specified with opacity. Control the application of different color components by specifying a different percentage for each component (e.g. 90/100/10 is 90 red, 100 green, and 10 blue).

blend: A character string indicating the level of blending as a percentage.
colorize: A color value.

See ColorizeImage function in ImageMagick documentation.

6.6.35 ColorspaceType as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the potential type of image: sRGBColorspaceType, RGBColorspaceType, GRAYColorspaceType, etc.
Notes: See xx function in ImageMagick documentation.

6.6.36 Combine(Colorspace as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Combines one or more images into a single image.
Notes: The grayscale value of the pixels of each image in the sequence is assigned in order to the specified channels of the combined image. The typical ordering would be image 1 =>Red, 2 =>Green, 3 =>Blue, etc.

See CombineImages function in ImageMagick documentation.

6.6.37 CompareImagesLayers(ImageLayerMethod as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compares each image with the next in a sequence and returns the minimum bounding region of all the pixel differences (of the LayerMethod specified) it discovers.
Notes: Images do NOT have to be the same size, though it is best that all the images are 'coalesced' (images are all the same size, on a flattened canvas, so as to represent exactly how an specific frame should look).

No GIF dispose methods are applied, so GIF animations must be coalesced before applying this image operator to find differences to them.

method: the layers type to compare images with. Must be one of... CompareAnyLayer, CompareClearLayer, CompareOverlayLayer.

See xx function in ImageMagick documentation.

6.6.38 ComplexImages(ComplexOperator as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Performs complex mathematics on an image sequence.
Notes: See ComplexImages function in ImageMagick documentation.

6.6.39 Composite(ComposeOperator as integer, Image as IMImage7MBS, ClipToSelf as boolean, xOffset as integer, yOffset as integer)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the second image composited onto the first at the specified offset, using the specified composite method.
Notes: See xx function in ImageMagick documentation.

6.6.40 CompositeLayers(CompositeOperator as Integer, Source as IMImage7MBS, XOffset as Integer, YOffset as Integer)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compose the source image sequence over the destination image sequence, starting with the current image in both lists.
Notes: Each layer from the two image lists are composted together until the end of one of the image lists is reached. The offset of each composition is also adjusted to match the virtual canvas offsets of each layer. As such the given offset is relative to the virtual canvas, and not the actual image.

Composition uses given x and y offsets, as the 'origin' location of the source images virtual canvas (not the real image) allowing you to compose a list of 'layer images' into the destiantioni images. This makes it well sutiable for directly composing 'Clears Frame Animations' or 'Coaleased Animations' onto a static or other 'Coaleased Animation' destination image list. GIF disposal handling is not looked at.

Special case:- If one of the image sequences is the last image (just a single image remaining), that image is repeatally composed with all the images in the other image list. Either the source or destination lists may be the single image, for this situation.

In the case of a single destination image (or last image given), that image will ve cloned to match the number of images remaining in the source image list.

This is equivelent to the ”-layer Composite” Shell API operator.

See xx function in ImageMagick documentation.

6.6.41 CompressColormap as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: compresses an image colormap by removing any duplicate or unused color entries.
Notes: Returns true on success.
See CompressImageColormap function in ImageMagick documentation.

6.6.42 ConsolidateCMYKImages as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Consolidates separate C, M, Y, and K planes into a single image.
Notes: See ConsolidateCMYKImages function in ImageMagick documentation.

6.6.43 Constructor(columns as Integer, Rows as Integer, map as String, StorageType as Integer, Pixels as MemoryBlock)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns an image from the pixel data you supply.
Notes: The pixel data must be in scanline order top-to-bottom. The data can be char, short int, int, float, or double. Float and double require the pixels to be normalized [0..1] , otherwise [0..QuantumRange] . For example, to create a 640x480 image from unsigned red-green-blue character data, use:

image = new IMImage7MBS(640,480,”RGB”, kCharPixel, pixels)

columns: width in pixels of the image.
rows: height in pixels of the image.
map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (0 is transparent), O = opacity (0 is opaque), C = cyan, Y = yellow, M = magenta, K = black, I = intensity (for grayscale), P = pad.
storage: Define the data type of the pixels. Float and double types are expected to be normalized [0..1]

otherwise [0..QuantumRange] . Choose from these types: CharPixel, DoublePixel, FloatPixel, IntegerPixel, LongPixel, QuantumPixel, or ShortPixel.
pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.

See ConstituteImage function in ImageMagick documentation.
See also:

- 6.6.44 Constructor(info as IMImageInfo7MBS) 921
- 6.6.45 Constructor(info as IMImageInfo7MBS, width as integer, height as integer, background as IMPixelInfo7MBS) 921
- 6.6.46 Constructor(Pic as Picture) 922

6.6.44 Constructor(info as IMImageInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image.
Notes: Returns a pointer to an image initialized to default values.
Many of the image default values are set from the image info. For example, filename, compression, depth, background color, and others.

See AcquireImage function in ImageMagick documentation.
See also:

- 6.6.43 Constructor(columns as Integer, Rows as Integer, map as String, StorageType as Integer, Pixels as MemoryBlock) 920
- 6.6.45 Constructor(info as IMImageInfo7MBS, width as integer, height as integer, background as IMPixelInfo7MBS) 921
- 6.6.46 Constructor(Pic as Picture) 922

6.6.45 Constructor(info as IMImageInfo7MBS, width as integer, height as integer, background as IMPixelInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a blank image canvas of the specified size and background color.
Notes: info: the image details.
width: the image width.
height: the image height.
background: the image color.

See NewMagickImage function in ImageMagick documentation.
See also:

- 6.6.43 Constructor(columns as Integer, Rows as Integer, map as String, StorageType as Integer, Pixels as MemoryBlock) 920
- 6.6.44 Constructor(info as IMImageInfo7MBS) 921
- 6.6.46 Constructor(Pic as Picture) 922

6.6.46 Constructor(Pic as Picture)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new picture based on Xojo picture.
Notes: Creates a new image and uses SetPicture method to fill it.

See NewMagickImage function in ImageMagick documentation.
See also:

- 6.6.43 Constructor(columns as Integer, Rows as Integer, map as String, StorageType as Integer, Pixels as MemoryBlock) 920
- 6.6.44 Constructor(info as IMImageInfo7MBS) 921
- 6.6.45 Constructor(info as IMImageInfo7MBS, width as integer, height as integer, background as IMPixelInfo7MBS) 921

6.6.47 Contrast(sharpen as boolean) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Enhances the intensity differences between the lighter and darker elements of the image.
Notes: Set sharpen to a True to increase the image contrast otherwise the contrast is reduced.

sharpen: Increase or decrease image contrast.

See ContrastImage function in ImageMagick documentation.

6.6.48 ContrastStretch(BlackPoint as double, WhitePoint as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Is a simple image enhancement technique that attempts to improve the contrast in an image by 'stretching' the range of intensity values it contains to span a desired range of values.

Notes: It differs from the more sophisticated histogram equalization in that it can only apply a linear scaling function to the image pixel values. As a result the 'enhancement' is less harsh.

BlackPoint: the black point.
WhitePoint: the white point.

See ContrastStretchImage function in ImageMagick documentation.

6.6.49 Convolve(kernelInfo as IMKernelInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a custom convolution kernel to the image.
Notes: See ConvolveImage function in ImageMagick documentation.

6.6.50 CopyPicture as Picture

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Copies RGB or gray pixels into a new Xojo picture.
Notes: You must make sure you have the right color space to make this work well with colorspaces. Xojo on Mac uses Generic RGB, while Windows may use the colorspace of the display.

As conversion between various quantum depths can be slow, please use rarely.

See also ExportPixels and ImportPixels to read/write pixel data to/from memory blocks.
See GetAuthenticPixels function in ImageMagick documentation.

6.6.51 CreateHBITMAP as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a Windows HBITMAP from an image.
Notes: See ImageToHBITMAP function in ImageMagick documentation.

6.6.52 Crop(rect as IMRectangleInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extracts a region of the image starting at the offset defined by geometry.
Notes: Region must be fully defined, and no special handling of geometry flags is performed.

See CropImage function in ImageMagick documentation.

6.6.53 CropToTiles(CropGeometry as string) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crops a single image, into a possible list of tiles.
Notes: This may include a single sub-region of the image. This basically applies all the normal geometry flags for Crop.

CropGeometry: A crop geometry string.

See CropImageToTiles function in ImageMagick documentation.

6.6.54 CycleColormap(displace as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Displaces an image's colormap by a given number of positions.
Notes: If you cycle the colormap a number of times you can produce a psychodelic effect.

WARNING: this assumes an images colormap is in a well know and defined order. Currently Imagemagick has no way of setting that order.

displace: displace the colormap this amount.

See CycleColormap function in ImageMagick documentation.

6.6.55 Decipher(passkey as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts cipher pixels to plain pixels.
Notes: See xx function in ImageMagick documentation.

6.6.56 DefineProperty(PropertyKeyValue as String) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Associates an assignment string of the form "key=value" with an artifact or options.
Notes: It is equivelent to SetProperty().
See DefineImageProperty function in ImageMagick documentation.

6.6.57 DeleteImageProfile(Name as String) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Deletes a profile from the image by its name.
Notes: See xx function in ImageMagick documentation.

6.6.58 DeleteProperty(Key as String) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Deletes an image property.
Notes: key: the image property.
See DeleteImageProperty function in ImageMagick documentation.

6.6.59 Deskew(x as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes skew from the image.
Notes: Skew is an artifact that occurs in scanned images because of the camera being misaligned, imperfections in the scanning or surface, or simply because the paper was not placed completely flat when scanned.

The result will be auto-croped if the artifact "deskew:auto-crop" is defined, while the amount the image is to be deskewed, in degrees is also saved as the artifact "deskew:angle".

threshold: separate background from foreground.

See DeskewImage function in ImageMagick documentation.

6.6.60 Despeckle as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduces the speckle noise in an image while perserving the edges of the original image.
Notes: A speckle removing filter uses a complementary hulling technique (raising pixels that are darker than their surrounding neighbors, then complementarily lowering pixels that are brighter than their surrounding neighbors) to reduce the speckle index of that image (reference Crimmins speckle removal).

See DespeckleImage function in ImageMagick documentation.

6.6.61 DestroyImageList

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys an image list.
Notes: See DestroyImageList function in ImageMagick documentation.

6.6.62 DestroyImageProfiles

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Releases memory associated with an image profile map.
Notes: See xx function in ImageMagick documentation.

6.6.63 DestroyProperties

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Destroys all properties and associated memory attached to the given image.
Notes: See DestroyImageProperties function in ImageMagick documentation.

6.6.64 DisposeImages as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the coalesced frames of a GIF animation as it would appear after the GIF dispose method of that frame has been applied.
Notes: That is it returned the appearance of each frame before the next is overlaid.

See DisposeImages function in ImageMagick documentation.

6.6.65 Distort(DistortImageMethod as integer, values() as double, bestfit as boolean) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Distorts an image using various distortion methods, by mapping color lookups of the source image to a new destination image usally of the same size as the source image, unless 'bestfit' is set to true.
Notes: If 'bestfit' is enabled, and distortion allows it, the destination image is adjusted to ensure the whole source 'image' will just fit within the final destination image, which will be sized and offset accordingly. Also in many cases the virtual offset of the source image will be taken into account in the mapping.

If the '-verbose' control option has been set print to standard error the equicelent '-fx' formula with coefficients for the function, if practical.

method: the method of image distortion.
ArcDistortion always ignores source image offset, and always 'bestfit' the destination image with the top left corner offset relative to the polar mapping center.
Affine, Perspective, and Bilinear, do least squares fitting of the distrotion when more than the minimum number of control point pairs are provided.
Perspective, and Bilinear, fall back to a Affine distortion when less than 4 control point pairs are provided. While Affine distortions let you use any number of control point pairs, that is Zero pairs is a No-Op (viewport only) distortion, one pair is a translation and two pairs of control points do a scale-rotate-translate, without any shearing.

values: an array of floating point arguments for this method.

bestfit: Attempt to 'bestfit' the size of the resulting image. This also forces the resulting image to be a 'layered' virtual canvas image. Can be overridden using 'distort:viewport' setting.

Extra Controls from Image meta-data (artifacts)...

- "verbose" Output to stderr alternatives, internal coefficents, and FX equivalents for the distortion operation (if feasible). This forms an extra check of the distortion method, and allows users access to the internal constants IM calculates for the distortion.
- "distort:viewport" Directly set the output image canvas area and offest to use for the resulting image, rather than use the original images canvas, or a calculated 'bestfit' canvas.
- "distort:scale" Scale the size of the output canvas by this amount to provide a method of Zooming, and for super-sampling the results.

Other settings that can effect results include

- 'interpolate' For source image lookups (scale enlargements)
- 'filter' Set filter to use for area-resampling (scale shrinking). Set to 'point' to turn off and use 'interpolate' lookup instead

See DistortImage function in ImageMagick documentation.

6.6.66 DistortResize(x as Integer, y as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image using the equivalent but slower image distortion operator.
Notes: The filter is applied using a EWA cylindrical resampling. But like resize the final image size is limited to whole pixels with no effects by virtual-pixels on the result.

Note that images containing a transparency channel will be twice as slow to resize as images one without transparency.

See DistortResizeImage function in ImageMagick documentation.

6.6.67 Edge(radius as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Finds edges in an image.
Notes: Radius defines the radius of the convolution filter. Use a radius of 0 and Edge selects a suitable radius for you.

radius: the radius of the pixel neighborhood.

See EdgeImage function in ImageMagick documentation.

6.6.68 Emboss(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a grayscale image with a three-dimensional effect.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma).

For reasonable results, radius should be larger than sigma. Use a radius of 0 and Emboss() selects a suitable radius for you.

radius: the radius of the pixel neighborhood.
sigma: the standard deviation of the Gaussian, in pixels.

See EmbossImage function in ImageMagick documentation.

6.6.69 Encipher(passkey as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts pixels to cipher-pixels.
Notes: See xx function in ImageMagick documentation.

6.6.70 Enhance as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a digital filter that improves the quality of a noisy image.
Notes: See xx function in ImageMagick documentation.

6.6.71 Equalize as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a histogram equalization to the image.
Notes: See xx function in ImageMagick documentation.

6.6.72 Excerpt(rect as IMRectangleInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a excerpt of the image as defined by the geometry.
Notes: geometry: Define the region of the image to extend with members x, y, width, and height.

See ExcerptImage function in ImageMagick documentation.

6.6.73 ExportPixels(x as integer, y as integer, width as integer, height as integer, map as string, storageType as integer, data as Ptr) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extracts pixel data from an image and returns it to you.
Notes: The method returns MagickTrue on success otherwise MagickFalse if an error is encountered. The data is returned as char, short int, Quantum, unsigned int, unsigned long long, float, or double in the order specified by map.

Suppose you want to extract the first scanline of a 640x480 image as character data in red-green-blue order:

image.ExportImagePixels(0, 0, 640, 480, ”RGB”, kCharPixel, pixels)

- x,y,width,height: These values define the perimeter of a region of pixels you want to extract.
- map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (0 is transparent), O = opacity (0 is opaque), C = cyan, Y = yellow, M = magenta, K = black, I = intensity (for grayscale), P = pad.
- type: Define the data type of the pixels. Float and double types are normalized to [0..1] otherwise [0..QuantumRange] . Choose from these types: CharPixel (char *), DoublePixel (double *), FloatPixel (float *), LongPixel (unsigned int *), LongLongPixel (unsigned long long *), QuantumPixel (Quantum *), or ShortPixel (unsigned short *).

pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.

See ExportImagePixels function in ImageMagick documentation.

6.6.74 Extent(rect as IMRectangleInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends the image as defined by the geometry, gravity, and image background color.
Notes: Set the (x,y) offset of the geometry to move the original image relative to the extended image.

rect: Define the region of the image to extend with members x, y, width, and height.

See ExtentImage function in ImageMagick documentation.
See also:

- 6.6.235 extent as Integer 978

6.6.75 Flip as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a vertical mirror image by reflecting the pixels around the central x-axis.
Notes: See FlipImage function in ImageMagick documentation.

6.6.76 Flop as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a horizontal mirror image by reflecting the pixels around the central y-axis.
Notes: See FlopImage function in ImageMagick documentation.

6.6.77 ForwardFourierTransformImage(modulus as Boolean) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Implements the discrete Fourier transform (DFT) of the image either as a magnitude / phase or real / imaginary image pair.
Notes: modulus: if true, return as transform as a magnitude / phase pair otherwise a real / imaginary image pair.
See ForwardFourierTransformImage function in ImageMagick documentation.

6.6.78 Frame(frameInfo as IMFrameInfo7MBS, CompositeOperator as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a simulated three-dimensional border around the image.
Notes: The color of the border is defined by the matte_color member of image. Members width and height of frame_info specify the border width of the vertical and horizontal sides of the frame. Members inner and outer indicate the width of the inner and outer shadows of the frame.

frameInfo: Define the width and height of the frame and its bevels.
CompositeOperator: the composite operator.

See FrameImage function in ImageMagick documentation.

6.6.79 Fx(expression as string) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a mathematical expression to the specified image.
Notes: expression: A mathematical expression.

See FxImage function in ImageMagick documentation.

6.6.80 Gamma(level as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gamma-corrects a particular image channel.
Notes: The same image viewed on different devices will have perceptual differences in the way the image's intensities are represented on the screen. Specify individual gamma levels for the red, green, and blue channels, or adjust all three with the gamma parameter. Values typically range from 0.8 to 2.3.

You can also reduce the influence of a particular channel with a gamma value of 0.

level: the image gamma as a string (e.g. 1.6,1.2,1.0).

See GammaImage function in ImageMagick documentation.
See also:

- 6.6.240 Gamma as Double 979

6.6.81 GaussianBlur(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Blurs an image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, the radius should be larger than sigma. Use a radius of 0 and GaussianBlur selects a suitable radius for you

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.

See GaussianBlurImage function in ImageMagick documentation.

6.6.82 GetImageDynamicThreshold(clusterThreshold as Double, smoothThreshold as Double, byref pixelinfo as IMPixelInfo7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the dynamic threshold for an image.
Notes: clusterThreshold: This double represents the minimum number of pixels contained in a hexahedra before it can be considered valid (expressed as a percentage).
smoothThreshold: the smoothing threshold eliminates noise in the second derivative of the histogram. As the value is increased, you can expect a smoother second derivative.
pixelinfo: return the dynamic threshold here.

Returns true on success.
See GetImageDynamicThreshold function in ImageMagick documentation.

6.6.83 GetImageProfile(name as string) as string

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets a profile associated with an image by name.
Notes: See xx function in ImageMagick documentation.

6.6.84 GetImageQuantizeError as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Measures the difference between the original and quantized images.
Notes: This difference is the total quantization error.
The error is computed by summing over all pixels in an image the distance squared in RGB space between each reference pixel value and its quantized value. These values are computed:

- mean_error_per_pixel: This value is the mean error for any single pixel in the image.
- normalized_mean_square_error: This value is the normalized mean quantization error for any single pixel in the image. This distance measure is normalized to a range between 0 and 1. It is independent of the range of red, green, and blue values in the image.
- normalized_maximum_square_error: This value is the normalized maximum quantization error for any single pixel in the image. This distance measure is normalized to a range between 0 and 1. It is independent of the range of red, green, and blue values in your image.

See GetImageQuantizeError function in ImageMagick documentation.

6.6.85 GetMagickProperty(ImageInfo as IMImageInfo7MBS = nil, embedText as String) as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets attributes or calculated values that is associated with a fixed known property name, or single letter property.
Notes: It may be called if no image is defined (IMv7), in which case only global ImageInfo values are available:

\n newline
\r carriage return
<less-than character.
>greater-than character.
& ampersand character.
%% a percent sign
%b file size of image read in
%c comment meta-data property
%d directory component of path
%e filename extension or suffix
%f filename (including suffix)
%g layer canvas page geometry (equivalent to "%Wx%H%X%Y")
%h current image height in pixels
%i image filename (note: becomes output filename for "info:")
%k CALCULATED: number of unique colors
%l label meta-data property
%m image file format (file magic)
%n number of images in current image sequence
%o output filename (used for delegates)
%p index of image in current image list
%q quantum depth (compile-time constant)
%r image class and colorspace
%s scene number (from input unless re-assigned)
%t filename without directory or extension (suffix)
%u unique temporary filename (used for delegates)
%w current width in pixels
%x x resolution (density)
%y y resolution (density)
%z image depth (as read in unless modified, image save depth)
%A image transparency channel enabled (true/false)
%B file size of image in bytes
%C image compression type
%D image GIF dispose method
%G original image size (%wx%h; before any resizes)
%H page (canvas) height
%M Magick filename (original file exactly as given, including read mods)
%O page (canvas) offset (= %X%Y)

%P page (canvas) size (= %Wx%H)
%Q image compression quality (0 = default)
%S ?? scenes ??
%T image time delay (in centi-seconds)
%U image resolution units
%W page (canvas) width
%X page (canvas) x offset (including sign)
%Y page (canvas) y offset (including sign)
%Z unique filename (used for delegates)
%@ CALCULATED: trim bounding box (without actually trimming)
%# CALCULATED: 'signature' hash of image values

This routine only handles specifically known properties. It does not handle special prefixed properties, profiles, or expressions. Nor does it return any free-form property strings.

The returned string is stored in a structure somewhere, and should not be directly freed. If the string was generated (common) the string will be stored as as either as artifact or option 'get-property'. These may be deleted (cleaned up) when no longer required, but neither artifact or option is guranteed to exist.
See GetMagickProperty function in ImageMagick documentation.

6.6.86 GetNextImageProfile as string

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets the next profile name for an image.
Notes: See xx function in ImageMagick documentation.

6.6.87 GetNextImageProperty as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets the next free-form string property name.
Notes: See GetNextImageProperty function in ImageMagick documentation.

6.6.88 GetProperty(PropertyKey as String) as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gets a value associated with an image property.
Notes: This includes, profile prefixes, such as "exif:", "iptc:" and "8bim:" It does not handle non-prifile prefixes, such as "fx:", "option:", or "artifact:".

The returned string is stored as a properity of the same name for faster lookup later.
See GetImageProperty function in ImageMagick documentation.

6.6.89 Grayscale(PixelIntensityMethod as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Converts the image to grayscale.
Notes: PixelIntensityMethod: the pixel intensity method.

See GrayscaleImage function in ImageMagick documentation.

6.6.90 HaldClut(clutImage as IMImage7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a Hald color lookup table to the image.
Notes: A Hald color lookup table is a 3-dimensional color cube mapped to 2 dimensions. Create it with the HALD coder. You can apply any color transformation to the Hald image and then use this method to apply the transform to the image.

clutImage: the color lookup table image for replacement color values.

See HaldClutImage function in ImageMagick documentation.

6.6.91 HandleMemory as memoryblock

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a memoryblock with a copy of the internal image structure.
Notes: See xx function in ImageMagick documentation.

6.6.92 Histogram as IMPixelInfo7MBS()

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the unique colors in an image.
Notes: Returns array of colors.
See GetImageHistogram function in ImageMagick documentation.

6.6.93 IdentifyImageGray as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns grayscale if all the pixels in the image have the same red, green, and blue intensities, and bi-level is the intensity is either 0 or QuantumRange.
Notes: Returns one of the image type constants.
Otherwise undefined is returned.
See IdentifyImageGray function in ImageMagick documentation.

6.6.94 IdentifyImageMonochrome as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if all the pixels in the image have the same red, green, and blue intensities and the intensity is either 0 or QuantumRange.
Notes: See IdentifyImageMonochrome function in ImageMagick documentation.

6.6.95 IdentifyImageType as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the potential type of image.
Notes: Bilevel Grayscale GrayscaleMatte Palette PaletteMatte TrueColor TrueColorMatte ColorSeparation ColorSeparationMatte

To ensure the image type matches its potential, use SetImageType.

See IdentifyImageType function in ImageMagick documentation.

6.6.96 IdentifyPaletteImage as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the image has 256 unique colors or less.
Notes: See IdentifyPaletteImage function in ImageMagick documentation.

6.6.97 ImageDepth as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the depth of a particular image channel.
Notes: See GetImageDepth function in ImageMagick documentation.

6.6.98 ImageQuantumDepth(constrain as Boolean = false) as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the depth of the image rounded to a legal quantum depth: 8, 16, or 32.
Notes: constrain: A value other than false, constrains the depth to a maximum of MAGICKCORE_QUANTUM_DEPTH.

See GetImageQuantumDepth function in ImageMagick documentation.

6.6.99 ImageStatistics as IMChannelStatistics7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns statistics for each channel in the image.
Notes: The statistics include the channel depth, its minima, maxima, mean, standard deviation, kurtosis and skewness.

See GetImageStatistics function in ImageMagick documentation.

6.6.100 ImagesToBlob(info as IMImageInfo7MBS) as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImageToBlobs saves multiple images in a blob.
Notes: It returns the image as a formatted blob. The magick member of the Image determines the format of the returned blob (GIF, JPEG, PNG, etc.). This method is the equivalent of WriteImage, but writes the formatted "file" to a memory buffer rather than to an actual file.
See ImagesToBlob function in ImageMagick documentation.

6.6.101 ImageToBlob(info as IMImageInfo7MBS) as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: ImageToBlob implements direct to memory image formats.
Notes: It returns the image as a formatted blob. The magick member of the Image determines the format of the returned blob (GIF, JPEG, PNG, etc.). This method is the equivalent of WriteImage, but writes the

formatted "file" to a memory buffer rather than to an actual file.
See ImageToBlob function in ImageMagick documentation.

6.6.102 Implode(amount as double, pixelInterpolateMethod as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image that is a copy of an existing one with the image pixels "implode" by the specified percentage.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Method ImplodeImage returns a pointer to the image after it is implode. A null image is returned if there is a memory shortage.

amount: Define the extent of the implosion.
method: the pixel interpolation method.

See ImplodeImage function in ImageMagick documentation.

6.6.103 ImportPixels(x as integer, y as integer, width as integer, height as integer, map as string, storageType as integer, data as Ptr) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Accepts pixel data and stores in the image at the location you specify.
Notes: The method returns true on success otherwise false if an error is encountered. The pixel data can be either char, Quantum, short int, unsigned int, unsigned long long, float, or double in the order specified by map.

Suppose your want to upload the first scanline of a 640x480 image from character data in red-green-blue order:

image.ImportImagePixels(0, 0, 640, 480, ”RGB”, kCharPixel, pixels)

- x,y,width,height: These values define the perimeter of a region of pixels you want to define.
- map: This string reflects the expected ordering of the pixel array. It can be any combination or order of R = red, G = green, B = blue, A = alpha (0 is transparent), O = opacity (0 is opaque), C = cyan, Y = yellow, M = magenta, K = black, I = intensity (for grayscale), P = pad.
- type: Define the data type of the pixels. Float and double types are normalized to [0..1] otherwise [0..QuantumRange] . Choose from these types: CharPixel (char *), DoublePixel (double *), FloatPixel (float *), LongPixel (unsigned int *), LongLongPixel (unsigned long long *), QuantumPixel (Quantum *), or ShortPixel (unsigned short *).
- pixels: This array of values contain the pixel components as defined by map and type. You must preallocate this array where the expected length varies depending on the values of width, height, map, and type.

See ImportImagePixels function in ImageMagick documentation.

6.6.104 IntegralRotate(degrees as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rotates the image an integral of 90 degrees.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the rotated image.

rotations: Specifies the number of 90 degree rotations.

See IntegralRotateImage function in ImageMagick documentation.

6.6.105 InterpolativeResize(columns as Integer, Rows as Integer, PixelInterpolateMethod as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resizes an image using the specified interpolation method.
Notes: columns: the number of columns in the resized image.
rows: the number of rows in the resized image.
method: the pixel interpolation method.

See xx function in ImageMagick documentation.

6.6.106 InterpretProperties(ImageInfo as IMImageInfo7MBS = nil, embedText as String) as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces any embedded formatting characters with the appropriate image property and returns the interpreted text.
Notes: This searches for and replaces \n \r \replaced by newline, return, and percent resp. &lt; &gt; &amp; replaced by '<', '>', '&' resp. replaced by percent

x [x] where 'x' is a single letter properity, case sensitive). [type:name] where 'type' a is special and known prefix. [name] where 'name' is a specifically known attribute, calculated value, or a per-image property string name, or a per-image 'artifact' (as generated from a global option). It may contain ':' as long as the prefix is not special.

Single letter substitutions will only happen if the character before the percent is NOT a number. But braced substitutions will always be performed. This prevents the typical usage of percent in a interpreted geometry argument from being substituted when the percent is a geometry flag.

If 'glob-expresions' ('*' or '?' characters) is used for 'name' it may be used as a search pattern to print multiple lines of "name=value\n" pairs of the associacted set of properties.

ImageInfo: the image info. (required)
image: the image.
embedText: the address of a character string containing the embedded formatting characters.
See InterpretImageProperties function in ImageMagick documentation.

6.6.107 InverseFourierTransformImage(phaseImage as IMImage7MBS, modulus as Boolean) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Implements the inverse discrete Fourier transform (DFT) of the image either as a magnitude / phase or real / imaginary image pair.
Notes: self: the magnitude or real image.
phaseImage: the phase or imaginary image.
modulus: if true, return transform as a magnitude / phase pair otherwise a real / imaginary image pair.
See InverseFourierTransformImage function in ImageMagick documentation.

6.6.108 IsEqual(other as IMImage7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compare the pixels of two images and returns immediately if any pixel is not identical.
Notes: See IsImagesEqual function in ImageMagick documentation.

6.6.109 IsHighDynamicRangeImage as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns MagickTrue if any pixel component is non-integer or exceeds the bounds of the quantum depth.
Notes: e.g. for Q16 0..65535.

See IsHighDynamicRangeImage function in ImageMagick documentation.

6.6.110 IsHistogram as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns MagickTrue if the image has 1024 unique colors or less.
Notes: See IsHistogramImage function in ImageMagick documentation.

6.6.111 IsImageOpaque as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if none of the pixels in the image have an alpha value other than OpaqueAlpha (QuantumRange).
Notes: Will return true immediatally if alpha channel is not available.

See IsImageOpaque function in ImageMagick documentation.

6.6.112 Kuwahara(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Kuwahara is an edge preserving noise reduction filter.
Notes: radius: the square window radius.
sigma: the standard deviation of the Gaussian, in pixels.

See KuwaharaImage function in ImageMagick documentation.

6.6.113 Level(BlackPoint as double, WhitePoint as double, gamma as Double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adjusts the levels of a particular image channel by scaling the colors falling between specified white and black points to the full available quantum range.
Notes: The parameters provided represent the black, and white points. The black point specifies the darkest color in the image. Colors darker than the black point are set to zero. White point specifies the lightest color in the image. Colors brighter than the white point are set to the maximum quantum value.

If a '!' flag is given, map black and white colors to the given levels rather than mapping those levels to black and white. See LevelizeImage() below.

Gamma specifies a gamma correction to apply to the image.

BlackPoint: The level to map zero (black) to.
WhitePoint: The level to map QuantumRange (white) to.

See LevelImage function in ImageMagick documentation.

6.6.114 LevelImageColors(BlackColor as IMPixelInfo7MBS, WhiteColor as IMPixelInfo7MBS, invert as Boolean) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Levels image colors.
Notes: Maps the given color to "black" and "white" values, linearly spreading out the colors, and level values on a channel by channel bases, as per LevelImage(). The given colors allows you to specify different level ranges for each of the color channels separately.

If the boolean 'invert' is set true the image values will modifyed in the reverse direction. That is any existing "black" and "white" colors in the image will become the color values given, with all other values compressed appropriatally. This effectivally maps a greyscale gradient into the given color gradient.

BlackColor: The color to map black to/from
WhiteColor: The color to map white to/from
invert: if true map the colors (levelize), rather than from (level)

See LevelImageColors function in ImageMagick documentation.

6.6.115 Levelize(BlackPoint as double, WhitePoint as double, gamma as Double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies the reversed LevelImage() operation to just the specific channels specified.
Notes: It compresses the full range of color values, so that they lie between the given black and white points. Gamma is applied before the values are mapped.

LevelizeImage() can be called with by using a +level command line API option, or using a '!' on a -level or LevelImage() geometry string.

It can be used to de-contrast a greyscale image to the exact levels specified. Or by using specific levels for each channel of an image you can convert a gray-scale image to any linear color gradient, according to those levels.

BlackPoint: The level to map zero (black) to.
WhitePoint: The level to map QuantumRange (white) to.
gamma: adjust gamma by this factor before mapping values.

See LevelizeImage function in ImageMagick documentation.

6.6.116 LinearStretch(BlackPoint as double, WhitePoint as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Discards any pixels below the black point and above the white point and levels the remaining pixels.
Notes: blackPoint: the black point.
whitePoint: the white point.

See LinearStretchImage function in ImageMagick documentation.

6.6.117 LiquidRescale(columns as Integer, Rows as Integer, deltaX as Double, rigidity as Double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Rescales image with seam carving.
Notes: columns: the number of columns in the rescaled image.
rows: the number of rows in the rescaled image.
deltaX: maximum seam transversal step (0 means straight seams).
rigidity: introduce a bias for non-straight seams (typically 0).

See LiquidRescaleImage function in ImageMagick documentation.

6.6.118 LocalContrast(radius as double, strength as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Attempts to increase the appearance of large-scale light-dark transitions.
Notes: Local contrast enhancement works similarly to sharpening with an unsharp mask, however the mask is instead created using an image with a greater blur distance.

radius: the radius of the Gaussian blur, in percentage with 100 resulting in a blur radius of 20 of largest dimension.
strength: the strength of the blur mask in percentage.

See LocalContrastImage function in ImageMagick documentation.

6.6.119 Magnify as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Doubles the size of the image with a pixel art scaling algorithm.
Notes: See xx function in ImageMagick documentation.

6.6.120 MergeImageLayers(ImageLayerMethod as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Composes all the image layers from the current given image onward to produce a single image of the merged layers.
Notes: The inital canvas's size depends on the given LayerMethod, and is initialized using the first images background color. The images are then compositied onto that image in sequence using the given composition that has been assigned to each individual image.

method: the method of selecting the size of the initial canvas.
MergeLayer: Merge all layers onto a canvas just large enough to hold all the actual images. The virtual canvas of the first image is preserved but otherwise ignored.
FlattenLayer: Use the virtual canvas size of first image. Images which fall outside this canvas is clipped. This can be used to 'fill out' a given virtual canvas.
MosaicLayer: Start with the virtual canvas of the first image, enlarging left and right edges to contain all images. Images with negative offsets will be clipped.
TrimBoundsLayer: Determine the overall bounds of all the image layers just as in "MergeLayer", then adjust the the canvas and offsets to be relative to those bounds, without overlaying the images.
WARNING: a new image is not returned, the original image sequence page data is modified instead.

See MergeImageLayers function in ImageMagick documentation.

6.6.121 Minify as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Is a convenience method that scales an image proportionally to half its size.
Notes: See xx function in ImageMagick documentation.

6.6.122 MinMaxStretch(black as double, white as double, gamma as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Stretches image colors.
Notes: Uses the exact minimum and maximum values found in each of the channels given, as the BlackPoint and WhitePoint to linearly stretch the colors (and histogram) of the image. The stretch points are also moved further inward by the adjustment values given.

If the adjustment values are both zero this function is equivalent to a perfect normalization (or autolevel) of the image.

Each channel is stretched independantally of each other (producing color distortion) unless the special 'SyncChannels' flag is also provided in the channels setting. If this flag is present the minimum and maximum point will be extracted from all the given channels, and those channels will be stretched by exactly the same amount (preventing color distortion).

In the special case that only ONE value is found in a channel of the image that value is not stretched, that value is left as is.

The 'SyncChannels' is turned on in the 'DefaultChannels' setting by default.

See MinMaxStretchImage function in ImageMagick documentation.

black, white: move the black / white point inward from the minimum and maximum points by this color value.
gamma: the gamma.

6.6.123 Modify as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Ensures that there is only a single reference to the image to be modified, updating the provided image pointer to point to a clone of the original image if necessary.
Notes: See ModifyImage function in ImageMagick documentation.

6.6.124 Modulate(modulate as String) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Lets you control the brightness, saturation, and hue of an image.
Notes: Modulate represents the brightness, saturation, and hue as one parameter (e.g. 90,150,100). If the image colorspace is HSL, the modulation is lightness, saturation, and hue. For HWB, use blackness, whiteness, and hue. And for HCL, use chrome, luma, and hue.

modulate: Define the percent change in brightness, saturation, and hue.

See ModulateImage function in ImageMagick documentation.

6.6.125 Morphology(MorphologyMethod as Integer, iterations as Integer, kernel as IMKernelInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a user supplied kernel to the image according to the given mophology method.
Notes: This function applies any and all user defined settings before calling the above internal function MorphologyApply().

User defined settings include... * Output Bias for Convolution and correlation ("-define convolve:bias=??") * Kernel Scale/normalize settings ("-define convolve:scale=??") This can also includes the addition of a scaled unity kernel. * Show Kernel being applied ("-define morphology:showKernel=1")

Other operators that do not want user supplied options interfering, especially "convolve:bias" and "morphology:showKernel" should use MorphologyApply() directly.

See MorphologyImage function in ImageMagick documentation.

6.6.126 MotionBlur(radius as double, sigma as double, angle as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Simulates motion blur.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and MotionBlurImage() selects a suitable radius for you. Angle gives the angle of the blurring motion.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
angle: Apply the effect along this angle.

See MotionBlurImage function in ImageMagick documentation.

6.6.127 Negate(gray as boolean = false) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Negates the colors in the reference image.
Notes: The grayscale option means that only grayscale values within the image are negated.

grayscale: If True, only negate grayscale pixels within the image.

See xx function in ImageMagick documentation.

6.6.128 Normalize as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Normalize method enhances the contrast of a color image by mapping the darkest 2 percent of all pixel to black and the brightest 1 percent to white.
Notes: See NormalizeImage function in ImageMagick documentation.

6.6.129 NumberColors as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the number of unique colors in an image.
Notes: See GetNumberColors function in ImageMagick documentation.

6.6.130 OilPaint(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a special effect filter that simulates an oil painting.
Notes: Each pixel is replaced by the most frequent color occurring in a circular region defined by radius.

radius: the radius of the circular neighborhood.
sigma: the standard deviation of the Gaussian, in pixels.

See xx function in ImageMagick documentation.

6.6.131 OneAuthenticPixel(X as Integer, Y as Integer) as MemoryBlock

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a single pixel at the specified (x,y) location.
Notes: The image background color is returned if an error occurs.
See GetOneAuthenticPixel function in ImageMagick documentation.

Memoryblock should have size of ChannelCount * QuantumSize.
Format of pixel values depend on what quantum size/type your library copy uses!

Use OneVirtualPixelInfo for a type independent color information.

6.6.132 OneVirtualPixel(X as Integer, Y as Integer) as MemoryBlock

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a single virtual pixel at the specified (x,y) location.
Notes: The image background color is returned if an error occurs.
If you plan to modify the pixel, use GetOneAuthenticPixel() instead.
See GetOneVirtualPixel function in ImageMagick documentation.

Memoryblock should have size of ChannelCount * QuantumSize.
Format of pixel values depend on what quantum size/type your library copy uses!

Use OneVirtualPixelInfo for a type independent color information.

6.6.133 OneVirtualPixelInfo(virtualPixelMethod as Integer, X as Integer, Y as Integer) as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a single pixel at the specified (x,y) location.
Notes: The image background color is returned if an error occurs.
If you plan to modify the pixel, use GetOneAuthenticPixel() instead.
See GetOneVirtualPixelInfo function in ImageMagick documentation.

6.6.134 OptimizeImageLayers as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compares each image the GIF disposed forms of the previous image in the sequence.
Notes: From this it attempts to select the smallest cropped image to replace each frame, while preserving the results of the GIF animation.

See OptimizeImageLayers function in ImageMagick documentation.

6.6.135 OptimizeImageTransparency

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Takes a frame optimized GIF animation, and compares the overlayed pixels against the disposal image resulting from all the previous frames in the animation.
Notes: Any pixel that does not change the disposal image (and thus does not effect the outcome of an overlay) is made transparent.

WARNING: This modifies the current images directly, rather than generate a new image sequence.

See OptimizeImageTransparency function in ImageMagick documentation.

6.6.136 OptimizePlusImageLayers as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: OptimizePlusImageLayers is exactly as OptimizeImageLayers(), but may also add or even remove extra frames in the animation, if it improves the total number of pixels in the resulting GIF animation.
Notes: See OptimizeImagePlusLayers function in ImageMagick documentation.

6.6.137 OrderedDither(threshold as string) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Will perform a ordered dither based on a number of pre-defined dithering threshold maps, but over multiple intensity levels, which can be different for different channels, according to the input argument.
Notes: threshold: A string containing the name of the threshold dither map to use, followed by zero or more numbers representing the number of color levels tho dither between.
Any level number less than 2 will be equivalent to 2, and means only binary dithering will be applied to each color channel.
No numbers also means a 2 level (bitmap) dither will be applied to all channels, while a single number is the number of levels applied to each channel in sequence. More numbers will be applied in turn to each of the color channels.
For example: ”o3x3,6” will generate a 6 level posterization of the image with a ordered 3x3 diffused pixel

dither being applied between each level. While checker,8,8,4 will produce a 332 colormaped image with only a single checkerboard hash pattern (50 grey) between each color level, to basically double the number of color levels with a bare minimim of dithering.

See OrderedDitherImage function in ImageMagick documentation.

6.6.138 Perceptible(epsilon as Double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Set each pixel whose value is less than | epsilon | to epsilon or -epsilon (whichever is closer) otherwise the pixel value remains unchanged.
Notes: epsilon: the epsilon threshold (e.g. 1.0e-9).

See PerceptibleImage function in ImageMagick documentation.

6.6.139 Ping(path as string) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns all the properties of an image or image sequence except for the pixels.
Notes: It is much faster and consumes far less memory than Read(). On failure, a nil image is returned and exception describes the reason for the failure.

See PingImage function in ImageMagick documentation.
See also:

- 6.6.273 Ping as Boolean 986

6.6.140 Posterize(levels as Integer, DitherMethod as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reduces the image to a limited number of colors for a ”poster” effect.
Notes: levels: Number of color levels allowed in each channel. Very low values (2, 3, or 4) have the most visible effect.
DitherMethod: choose from UndefinedDitherMethod, NoDitherMethod, RiemersmaDitherMethod, FloydSteinbergDitherMethod.

Returns true on success.
See PosterizeImage function in ImageMagick documentation.

6.6.141 Preview(PreviewType as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tiles 9 thumbnails of the specified image with an image processing operation applied with varying parameters.
Notes: This may be helpful pin-pointing an appropriate parameter for a particular image processing operation.

preview: the image processing operation. See preview constants.

See xx function in ImageMagick documentation.

6.6.142 Profile(name as string, ProfileData as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Associates, applies, or removes an ICM, IPTC, or generic profile with / to / from an image.
Notes: If the profile is nil, it is removed from the image otherwise added or applied. Use a name of '*' and a profile of nil to remove all profiles from the image.

ICC and ICM profiles are handled as follows: If the image does not have an associated color profile, the one you provide is associated with the image and the image pixels are not transformed. Otherwise, the colorspace transform defined by the existing and new profile are applied to the image pixels and the new profile is associated with the image.

name: Name of profile to add or remove: ICC, IPTC, or generic profile.
ProfileData: the profile data.

See ProfileImage function in ImageMagick documentation.

6.6.143 Properties as Dictionary

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries all properties with values.
Example:

```
Dim f As FolderItem = GetFolderItem("/Users/cs/Pictures/IMG_3625.jpg", FolderItem.PathTypeNative)
Dim i As New IMImageInfo7MBS

i.Filename = f.NativePath
```

```
Dim p As IMImage7MBS = i.ReadImage
Dim d As Dictionary = p.Properties
Break // see in debugger
```

Notes: See ResetImagePropertyIterator, GetNextImageProperty and GetImageProperty functions in ImageMagick documentation.

6.6.144 Quantize(quantizeInfo as IMQuantizeInfo7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Analyzes the colors within a reference image and chooses a fixed number of colors to represent the image.
Notes: The goal of the algorithm is to minimize the color difference between the input and output image while minimizing the processing time.

quantizeInfo: Specifies a QuantizeInfo object.

Returns true on success.
See QuantizeImage function in ImageMagick documentation.

6.6.145 QuantizeImages(quantizeInfo as IMQuantizeInfo7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: analyzes the colors within a set of reference images and chooses a fixed number of colors to represent the set.
Notes: The goal of the algorithm is to minimize the color difference between the input and output images while minimizing the processing time.

quantizeInfo: Specifies an QuantizeInfo structure.
self: Specifies a list of Images.
See QuantizeImages function in ImageMagick documentation.

6.6.146 QueueAuthenticPixels(X as Integer, Y as Integer, Width as Integer, Height as Integer) as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queues a mutable pixel region.
Notes: If the region is successfully initialized a pointer to a Quantum array representing the region is returned, otherwise nil is returned. The returned pointer may point to a temporary working buffer for the pixels or it may point to the final location of the pixels in memory.

Write-only access means that any existing pixel values corresponding to the region are ignored. This is useful if the initial image is being created from scratch, or if the existing pixel values are to be completely replaced without need to refer to their pre-existing values. The application is free to read and write the pixel buffer returned by QueueAuthenticPixels() any way it pleases. QueueAuthenticPixels() does not initialize the pixel array values. Initializing pixel array values is the application's responsibility.

Performance is maximized if the selected region is part of one row, or one or more full rows, since then there is opportunity to access the pixels in-place (without a copy) if the image is in memory, or in a memory-mapped file. The returned pointer must *never* be deallocated by the user.

Pixels accessed via the returned pointer represent a simple array of type Quantum. If the image type is CMYK or the storage class is PseudoClass, call GetAuthenticMetacontent() after invoking GetAuthenticPixels() to obtain the meta-content (of type void) corresponding to the region. Once the Quantum (and/or Quantum) array has been updated, the changes must be saved back to the underlying image using SyncAuthenticPixels() or they may be lost.

x, y, width and height define the perimeter of a region of pixels.

See QueueAuthenticPixels function in ImageMagick documentation.
Format of pixel values depend on what quantum size/type your library copy uses!

6.6.147 RaiseImage(rectangle as IMRectangleInfo7MBS, raise as boolean) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a simulated three-dimensional button-like effect by lightening and darkening the edges of the image.
Notes: Members width and height of raise_info define the width of the vertical and horizontal edge of the effect.

rectangle: Define the width and height of the raise area.
raise: A value other than zero creates a 3-D raise effect, otherwise it has a lowered effect.

See RaiseImage function in ImageMagick documentation.

6.6.148 RandomThreshold(minThreshold as Double, maxThreshold as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the value of individual pixels based on the intensity of each pixel compared to a random threshold.
Notes: The result is a low-contrast, two color image.

low,high: Specify the high and low thresholds. These values range from 0 to QuantumRange.

See xx function in ImageMagick documentation.

6.6.149 RangeThreshold(lowBlack as Double, lowWhite as double, highWhite as double, highBlack as double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies soft and hard thresholding.
Notes: lowBlack: Define the minimum black threshold value.
lowWhite: Define the minimum white threshold value.
highWhite: Define the maximum white threshold value.
highBlack: Define the maximum black threshold value.

See RangeThresholdImage function in ImageMagick documentation.

6.6.150 Read(path as string) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image or image sequence from a file or file handle.
Notes: The method returns a nil if there is a memory shortage or if the image cannot be read. On failure, a nil image is returned and exception describes the reason for the failure.

See ReadImage function in ImageMagick documentation.

6.6.151 Remap(quantizeInfo as IMQuantizeInfo7MBS, remapImage as IMImage7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces the colors of an image with the closest of the colors from the reference image.
Notes: quantizeInfo: Specifies an QuantizeInfo object.
remapImage: the reference image.

Returns true on success.
See RemapImage function in ImageMagick documentation.

6.6.152 RemapImages(quantizeInfo as IMQuantizeInfo7MBS, remapImage as IMImage7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Replaces the colors of a sequence of images with the closest color from a reference image.
Notes: quantizeInfo: Specifies an QuantizeInfo object.
remapImage: the reference image.

Returns true on success.
See RemapImages function in ImageMagick documentation.

6.6.153 RemoveDuplicateLayers

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that is exactly the same as the next image in the given image list.
Notes: Image size and virtual canvas offset must also match, though not the virtual canvas size itself.

No check is made with regards to image disposal setting, though it is the dispose setting of later image that is kept. Also any time delays are also added together. As such coalesced image animations should still produce the same result, though with duplicte frames merged into a single frame.

See xx function in ImageMagick documentation.

6.6.154 RemoveFirstImageFromList as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes and returns the first image in the list.
Notes: If the given image list pointer pointed to the removed first image, it is set to the new first image of list, or nil if list was emptied, otherwise it is left as is.

See RemoveFirstImageFromList function in ImageMagick documentation.

6.6.155 RemoveImageProfile(name as string) as string

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes a named profile from the image and returns its value.
Notes: See xx function in ImageMagick documentation.

6.6.156 RemoveProperty(Key as String)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes a property from the image and returns its value.
Notes: See RemoveImageProperty function in ImageMagick documentation.

6.6.157 RemoveZeroDelayLayers

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Removes any image that as a zero delay time.
Notes: Such images generally represent intermediate or partial updates in GIF animations used for file optimization. They are not ment to be displayed to users of the animation. Viewable images in an animation should have a time delay of 3 or more centi-seconds (hundredths of a second).

However if all the frames have a zero time delay, then either the animation is as yet incomplete, or it is not a GIF animation. This a non-sensible situation, so no image will be removed and a 'Zero Time Animation' warning (exception) given.

No warning will be given if no image was removed because all images had an appropriate non-zero time delay set.

Due to the special requirements of GIF disposal handling, GIF animations should be coalesced first, before calling this function, though that is not a requirement.

See RemoveZeroDelayLayers function in ImageMagick documentation.

6.6.158 Resample(xResolution as Double, yResolution as Double, filter as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize image in terms of its pixel size, so that when displayed at the given resolution it will be the same size in terms of real world units as the original image at the original resolution.
Notes: xResolution: the new image x resolution.
yResolution: the new image y resolution.
filter: Image filter to use.

See ResampleImage function in ImageMagick documentation.

6.6.159 ResetImageProfileIterator

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image profile iterator.
Notes: Use it in conjunction with GetNextImageProfile to iterate over all the profiles associated with an image.

See ResetImageProfileIterator function in ImageMagick documentation.

6.6.160 ResetPropertyIterator

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resets the image properties iterator.
Notes: Use it in conjunction with GetNextImageProperty() to iterate over all the values associated with an image property.
See ResetImagePropertyIterator function in ImageMagick documentation.

6.6.161 Resize(columns as integer, rows as integer, FilterID as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions, using the given filter (see AcquireFilterInfo()).
Notes: If an undefined filter is given the filter defaults to Mitchell for a colormapped image, a image with a matte channel, or if the image is enlarged. Otherwise the filter defaults to a Lanczos.

columns: the number of columns in the scaled image.
rows: the number of rows in the scaled image.
filterID: Image filter to use.

See ResizeImage function in ImageMagick documentation.

6.6.162 Roll(x as integer, y as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Offsets an image as defined by x and y offsets.
Notes: x: the number of columns to roll in the horizontal direction.
y: the number of rows to roll in the vertical direction.

See RollImage function in ImageMagick documentation.

6.6.163 Rotate(degrees as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image that is a rotated copy of an existing one.
Notes: Positive angles rotate counter-clockwise (right-hand rule), while negative angles rotate clockwise. Rotated images are usually larger than the originals and have 'empty' triangular corners. X axis. Empty triangles left over from shearing the image are filled with the background color defined by member 'background_color' of the image. RotateImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

degrees: Specifies the number of degrees to rotate the image.

See RotateImage function in ImageMagick documentation.

6.6.164 RotationalBlur(Angle as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a radial blur to the image.
Notes: angle: the angle of the radial blur.
blur: the blur.

See RotationalBlurImage function in ImageMagick documentation.

6.6.165 Sample(columns as integer, rows as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales an image to the desired dimensions with pixel sampling.
Notes: Unlike other scaling methods, this method does not introduce any additional color into the scaled image.

columns: the number of columns in the sampled image.
rows: the number of rows in the sampled image.

See SampleImage function in ImageMagick documentation.

6.6.166 Scale(columns as integer, rows as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions.
Notes: columns: the number of columns in the scaled image.
rows: the number of rows in the scaled image.

See ScaleImage function in ImageMagick documentation.

6.6.167 ScaleQuantumToChar(value as Double) as UInt8

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales a quantum value to 8 bit unsigned integer.
Notes: Depending on which library is loaded a quantum is an UInt8, UInt16, UInt32, Single or Double value.
This function normalizes and rounds values to nearest 8 bit integer value.

6.6.168 Segment(colorspaceType as Integer, verbose as Boolean, clusterThreshold as Double, smoothThreshold as Double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Segment an image by analyzing the histograms of the color components and identifying units

that are homogeneous with the fuzzy C-means technique.
Notes: colorspace: Indicate the colorspace.
verbose: Set to true to print detailed information about the identified classes.
clusterThreshold: This represents the minimum number of pixels contained in a hexahedra before it can be considered valid (expressed as a percentage).
smoothThreshold: the smoothing threshold eliminates noise in the second derivative of the histogram. As the value is increased, you can expect a smoother second derivative.

Returns true on success.
See SegmentImage function in ImageMagick documentation.

6.6.169 SelectiveBlur(radius as double, sigma as double, threshold as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Selectively blur pixels within a contrast threshold.
Notes: It is similar to the unsharpen mask that sharpens everything with contrast above a certain threshold.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
threshold: only pixels within this contrast threshold are included in the blur operation.

See SelectiveBlurImage function in ImageMagick documentation.

6.6.170 Separate(ChannelType as Integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Separates a channel from the image and returns it as a grayscale image.
Notes: See ChannelType constants for parameter.
See SeparateImage function in ImageMagick documentation.

6.6.171 SeparateImages as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns a separate grayscale image for each channel specified.
Notes: See SeparateImages function in ImageMagick documentation.

6.6.172 SetGray as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if all the pixels in the image have the same red, green, and blue intensities and changes the type of the image to bi-level or grayscale.
Notes: See xx function in ImageMagick documentation.

6.6.173 SetImageAlphaChannel(AlphaChannelOption as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Activates, deactivates, resets, or sets the alpha channel.
Notes: The alpha channel types: ActivateAlphaChannel, AssociateAlphaChannel, CopyAlphaChannel, DeactivateAlphaChannel, DisassociateAlphaChannel, ExtractAlphaChannel, OffAlphaChannel, OnAlphaChannel, OpaqueAlphaChannel, SetAlphaChannel, ShapeAlphaChannel, and TransparentAlphaChannel.

see AlphaChannel constants.

See SetImageAlphaChannel function in ImageMagick documentation.

6.6.174 SetImageColorMetric(other as IMImage7MBS) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Measures the difference between colors at each pixel location of two images.
Notes: A value other than 0 means the colors match exactly. Otherwise an error measure is computed by summing over all pixels in an image the distance squared in RGB space between each image pixel and its corresponding pixel in the reconstruct image. The error measure is assigned to these image members:

- mean_error_per_pixel: The mean error for any single pixel in the image.
- normalized_mean_error: The normalized mean quantization error for any single pixel in the image. This distance measure is normalized to a range between 0 and 1. It is independent of the range of red, green, and blue values in the image.
- normalized_maximum_error: The normalized maximum quantization error for any single pixel in the image. This distance measure is normalized to a range between 0 and 1. It is independent of the range of red, green, and blue values in your image.

A small normalized mean square error, accessed as image->normalized_mean_error, suggests the images are very similar in spatial layout and color.

See SetImageColorMetric function in ImageMagick documentation.

6.6.175 SetImageColorspace(Colorspace as integer) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the colorspace member of the Image object.
Notes: See SetImageColorspace function in ImageMagick documentation.

6.6.176 SetImageDepth(depth as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the depth of the image.
Notes: See SetImageDepth function in ImageMagick documentation.

6.6.177 SetImageProfile(name as string, ProfileData as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a named profile to the image.
Notes: If a profile with the same name already exists, it is replaced. This method differs from the ProfileImage() method in that it does not apply CMS color profiles.

name: the profile name, for example icc, exif, and 8bim (8bim is the Photoshop wrapper for iptc profiles).
profile: A String that contains the named profile.

See SetImageProfile function in ImageMagick documentation.

6.6.178 SetImageType(type as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets the type of image.
Notes: Choose from these types:
Bilevel Grayscale GrayscaleMatte Palette PaletteMatte TrueColor TrueColorMatte ColorSeparation ColorSeparationMatte OptimizeType

See SetImageType function in ImageMagick documentation.

6.6.179 SetMonochrome as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Checks if image is monochrome and changes type to black & white.
Notes: Returns true if all the pixels in the image have the same red, green, and blue intensities and the intensity is either 0 or QuantumRange and changes the type of the image to bi-level.

See SetImageMonochrome function in ImageMagick documentation.

6.6.180 SetPicture(pic as Picture) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sets pixel to the values in the picture.
Notes: You must make sure you have the right color space to make this work well with colorspaces.
Xojo on Mac uses Generic RGB, while Windows may use the colorspace of the display.

Returns true in case of success and false in case of failure.
As conversion between various quantum depths can be slow, please use rarely.

See also ExportPixels and ImportPixels to read/write pixel data to/from memory blocks.
See GetAuthenticPixels and SyncAuthenticPixels functions in ImageMagick documentation.

6.6.181 SetProperty(PropertyKey as String, Value as String) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Saves the given string value either to specific known attribute or to a freeform property string.
Notes: Attempting to set a property that is normally calculated will produce an exception.
See SetImageProperty function in ImageMagick documentation.

6.6.182 Shade(gray as boolean, azimuth as double, elevation as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shines a distant light on an image to create a three-dimensional effect.
Notes: You control the positioning of the light with azimuth and elevation; azimuth is measured in degrees off the x axis and elevation is measured in pixels above the Z axis.

gray: A value other than zero shades the intensity of each pixel.
azimuth, elevation: Define the light source direction.

See ShadeImage function in ImageMagick documentation.

6.6.183 Sharpen(radius as double, sigma as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens the image.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and Sharpen selects a suitable radius for you.

Using a separable kernel would be faster, but the negative weights cancel out on the corners of the kernel producing often undesirable ringing in the filtered result; this can be avoided by using a 2D gaussian shaped image sharpening kernel instead.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Laplacian, in pixels.

See SharpenImage function in ImageMagick documentation.

6.6.184 Shave(rect as IMRectangleInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Shaves pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

Shave returns a pointer to the shaved image. A null image is returned if there is a memory shortage or if the image width or height is zero.

rect: Specifies a RectangleInfo which defines the region of the image to crop.

See ShaveImage function in ImageMagick documentation.

6.6.185 Shear(Xshear as double, Yshear as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image that is an image copy of an existing one.
Notes: Shearing slides one edge of an image along the X or Y axis, creating a parallelogram. An X direction shear slides an edge along the X axis, while a Y direction shear slides an edge along the Y axis. The amount of the shear is controlled by a shear angle. For X direction shears, Xshear is measured relative to the Y axis, and similarly, for Y direction shears Yshear is measured relative to the X axis. Empty triangles left over from shearing the image are filled with the background color defined by member 'backgroundColor' of the image.. Shear() allocates the memory necessary for the new Image structure and returns a pointer to the new image.

See ShearImage function in ImageMagick documentation.

6.6.186 ShearRotate(degrees as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new image that is a rotated copy of an existing one.
Notes: Positive angles rotate counter-clockwise (right-hand rule), while negative angles rotate clockwise. Rotated images are usually larger than the originals and have 'empty' triangular corners. X axis. Empty triangles left over from shearing the image are filled with the background color defined by member 'background_color' of the image. ShearRotateImage allocates the memory necessary for the new Image structure and returns a pointer to the new image.

degrees: Specifies the number of degrees to rotate the image.

See ShearRotateImage function in ImageMagick documentation.

6.6.187 SigmoidalContrast(sharpen as boolean, contrast as Double, midpoint as Double) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: adjusts the contrast of an image with a non-linear sigmoidal contrast algorithm.
Notes: Increase the contrast of the image using a sigmoidal transfer function without saturating highlights or shadows. Contrast indicates how much to increase the contrast (0 is none; 3 is typical; 20 is pushing it); mid-point indicates where midtones fall in the resultant image (0 is white; 50 is middle-gray; 100 is black). Set sharpen to MagickTrue to increase the image contrast otherwise the contrast is reduced.

sharpen: Increase or decrease image contrast.

contrast: strength of the contrast, the larger the number the more 'threshold-like' it becomes.
midpoint: midpoint of the function as a color value 0 to QuantumRange.

See SigmoidalContrastImage function in ImageMagick documentation.

6.6.188 Solarize(threshold as double) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Applies a special effect to the image, similar to the effect achieved in a photo darkroom by selectively exposing areas of photo sensitive paper to light.
Notes: Threshold ranges from 0 to QuantumRange and is a measure of the extent of the solarization.

threshold: Define the extent of the solarization.

See SolarizeImage function in ImageMagick documentation.

6.6.189 SortColormapByIntensity as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: sorts the colormap of a PseudoClass image by decreasing color intensity.
Notes: See SortColormapByIntensity function in ImageMagick documentation.

6.6.190 SparseColor(SparseColorMethod as Integer, arguments() as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Given a set of coordinates, interpolates the colors found at those coordinates, across the whole image, using various methods.
Notes: method: the method to fill in the gradient between the control points.
The methods used for SparseColor() are often simular to methods used for DistortImage(), and even share the same code for determination of the function coefficents, though with more dimensions (or resulting values).
values: array of floating point arguments for this method– x,y,color_values– with color_values given as normalized values.

See SparseColorImage function in ImageMagick documentation.

6.6.191 Splice(rect as IMRectangleInfo7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Splices a solid color into the image as defined by the geometry.
Notes: rect: Define the region of the image to splice with members x, y, width, and height.

See SpliceImage function in ImageMagick documentation.

6.6.192 Spread(pixelInterpolateMethod as integer, radius as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Is a special effects method that randomly displaces each pixel in a square area defined by the radius parameter.
Notes: method: intepolation method.
radius: choose a random pixel in a neighborhood of this extent.

See SpreadImage function in ImageMagick documentation.

6.6.193 Stegano(watermarkImage as IMImage7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Hides a digital watermark within the image.
Notes: Recover the hidden watermark later to prove that the authenticity of an image. Offset defines the start position within the image to hide the watermark.

See SteganoImage function in ImageMagick documentation.

6.6.194 Stereo(otherImage as IMImage7MBS) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Combines two images and produces a single image that is the composite of a left and right image of a stereo pair.
Notes: Special red-green stereo glasses are required to view this effect.

See StereoImage function in ImageMagick documentation.

6.6.195 Strip as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Strips an image of all profiles and comments.
Notes: See StripImage function in ImageMagick documentation.

6.6.196 Swirl(degrees as double, pixelInterpolateMethod as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Swirls the pixels about the center of the image, where degrees indicates the sweep of the arc through which each pixel is moved.
Notes: You get a more dramatic effect as the degrees move from 1 to 360.

degrees: Define the tightness of the swirling effect.
method: the pixel interpolation method.

See xx function in ImageMagick documentation.

6.6.197 Sync as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Synchronizes image properties.
Notes: See SyncImage function in ImageMagick documentation.

6.6.198 SyncAuthenticPixels as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Saves the image pixels to the in-memory or disk cache.
Notes: The method returns MagickTrue if the pixel region is flushed, otherwise MagickFalse.
See SyncAuthenticPixels function in ImageMagick documentation.

6.6.199 Texture(Image as IMImage7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Layer a texture on pixels matching image background color.
Notes: See xx function in ImageMagick documentation.

6.6.200 Thumbnail(columns as integer, rows as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Changes the size of an image to the given dimensions and removes any associated profiles.
Notes: The goal is to produce small low cost thumbnail images suited for display on the Web.

columns: the number of columns in the scaled image.
rows: the number of rows in the scaled image.

See ThumbnailImage function in ImageMagick documentation.

6.6.201 TransformColorspace(ColorSpace as Integer) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Transforms an image colorspace, changing the image data to reflect the new colorspace.
Notes: See TransformImageColorspace function in ImageMagick documentation.

6.6.202 Transpose as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a horizontal mirror image by reflecting the pixels around the central y-axis while rotating them by 90 degrees.
Notes: See TransposeImage function in ImageMagick documentation.

6.6.203 Transverse as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a vertical mirror image by reflecting the pixels around the central x-axis while rotating them by 270 degrees.
Notes: See TransverseImage function in ImageMagick documentation.

6.6.204 Trim as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Trims pixels from the image edges.
Notes: It allocates the memory necessary for the new Image structure and returns a pointer to the new image.

See TrimImage function in ImageMagick documentation.

6.6.205 UniqueColors as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the unique colors of an image.
Notes: See UniqueImageColors function in ImageMagick documentation.

6.6.206 UnsharpMask(radius as double, sigma as double, amount as double, threshold as double) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Sharpens one or more image channels.
Notes: We convolve the image with a Gaussian operator of the given radius and standard deviation (sigma). For reasonable results, radius should be larger than sigma. Use a radius of 0 and UnsharpMask selects a suitable radius for you.

radius: the radius of the Gaussian, in pixels, not counting the center pixel.
sigma: the standard deviation of the Gaussian, in pixels.
gain: the percentage of the difference between the original and the blur image that is added back into the original.
threshold: the threshold in pixels needed to apply the diffence gain.

See UnsharpMaskImage function in ImageMagick documentation.

6.6.207 VirtualPixels(X as Integer, Y as Integer, Width as Integer, Height as Integer) as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns an immutable pixel region.
Notes: If the region is successfully accessed, a pointer to it is returned, otherwise nil is returned. The returned pointer may point to a temporary working copy of the pixels or it may point to the original pixels in memory. Performance is maximized if the selected region is part of one row, or one or more full rows, since there is opportunity to access the pixels in-place (without a copy) if the image is in memory, or in a memory-mapped file. The returned pointer must *never* be deallocated by the user.

Pixels accessed via the returned pointer represent a simple array of type Quantum. If the image type is CMYK or the storage class is PseudoClass, call AuthenticMetacontent after invoking AuthenticPixels to access the meta-content (of type void) corresponding to the the region.

If you plan to modify the pixels, use GetAuthenticPixels() instead.

Note, the VirtualPixels() and AuthenticPixels() methods are not thread-safe. In a threaded environment, use CacheViewVirtualPixels() or CacheViewAuthenticPixels() instead.

See GetVirtualPixels function in ImageMagick documentation.
Format of pixel values depend on what quantum size/type your library copy uses!

6.6.208 Wave(amplitude as double, wavelength as double, pixelInterpolateMethod as integer) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a "ripple" effect in the image by shifting the pixels vertically along a sine wave whose amplitude and wavelength is specified by the given parameters.
Notes: amplitude, wavelength: Define the amplitude and wave length of the sine wave.
pixelInterpolateMethod: the pixel interpolation method.

See WaveImage function in ImageMagick documentation.

6.6.209 WhiteThreshold(thresholds as string) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Is like ThresholdImage() but forces all pixels above the threshold into white while leaving all pixels at or below the threshold unchanged.
Notes: threshold: Define the threshold value.

See WhiteThresholdImage function in ImageMagick documentation.

6.6.210 Properties

6.6.211 AlphaColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha color.
Notes: When you query this value, you get a copy in a new object.
To modify, you need to change copy and assign back.
(Read and Write property)

6.6.212 AlphaTrait as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The pixel trait for alpha channel.
Notes: Is transparency channel defined and active?
(Read and Write property)

6.6.213 AuthenticMetacontent as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the authentic metacontent corresponding with the last call to QueueAuthenticPixels() or GetVirtualPixels().
Notes: Nil is returned if the associated pixels are not available.
See GetAuthenticMetacontent function in ImageMagick documentation.
(Read only property)

6.6.214 AuthenticPixelQueue as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the authentic pixels associated corresponding with the last call to QueueAuthenticPixels or GetAuthenticPixels.
Notes: See GetAuthenticPixelQueue function in ImageMagick documentation.
(Read only property)

6.6.215 BackgroundColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Current background color attribute.
Notes: When you query this value, you get a copy in a new object.
To modify, you need to change copy and assign back.
(Read and Write property)

6.6.216 BlackPointCompensation as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to enable black point compensation for color space conversions.
Notes: (Read and Write property)

6.6.217 BlobSize as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The size of the blob.
Notes: (Read only property)

6.6.218 BorderColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Current bordercolor attribute.
Notes: When you query this value, you get a copy in a new object.
To modify, you need to change copy and assign back.
(Read and Write property)

6.6.219 ChannelMask as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The channel mask field.
Notes: See Channel constants.
(Read and Write property)

6.6.220 Channels as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The channel mask.
Notes: (Read and Write property)

6.6.221 Colors as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Size of color table, or actual color count.
Notes: Only valid if image is not DirectClass.
(Read only property)

6.6.222 Colorspace as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorspace of image data.
Notes: (Read only property)

6.6.223 Columns as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Physical width of image in pixel.
Notes: (Read only property)

6.6.224 Compose as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Alpha composition method for layered images.
Notes: (Read and Write property)

6.6.225 CompositeMask as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to composite with mask.
Notes: (Read and Write property)

6.6.226 Compression as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compression of image when read/write.
Notes: (Read and Write property)

6.6.227 Debug as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Debug output attribute.
Notes: (Read and Write property)

6.6.228 Delay as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Animation delay time.
Notes: (Read and Write property)

6.6.229 Depth as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Depth of image on read/write.
Notes: (Read only property)

6.6.230 Directory as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The directory.
Notes: (Read and Write property)

6.6.231 Dispose as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: GIF animation disposal method.
Notes: (Read and Write property)

6.6.232 Dither as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Dithering on/off.
Notes: (Read and Write property)

6.6.233 Duration as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Total animation duration.
Notes: sum(delay*iterations)
(Read and Write property)

6.6.234 Endian as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Raw data integer ordering on read/write.
Notes: (Read and Write property)

6.6.235 extent as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Size of image read from disk.
Notes: (Read only property)
See also:

- 6.6.74 Extent(rect as IMRectangleInfo7MBS) as IMImage7MBS 930

6.6.236 ExtractInfo as IMRectangleInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Extract Info.
Notes: (Read and Write property)

6.6.237 Filename as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path of the image.
Notes: (Read and Write property)

6.6.238 Filter as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resize/distort filter to apply.
Notes: (Read and Write property)

6.6.239 Fuzz as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Current color fuzz attribute.
Notes: (Read and Write property)

6.6.240 Gamma as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The gamme of the image.
Notes: (Read only property)
See also:

- 6.6.80 Gamma(level as double) as Boolean 932

6.6.241 Geometry as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The geomtry.
Notes: (Read and Write property)

6.6.242 Gravity as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Gravity attribute for positioning in image.
Notes: (Read and Write property)

6.6.243 Handle as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.6.244 HasAlphaChannel as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns false if the image alpha channel is not activated.
Notes: That is, the image is RGB rather than RGBA or CMYK rather than CMYKA.
See GetImageAlphaChannel function in ImageMagick documentation.
(Read only property)

6.6.245 Height as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Physical height of image in pixel.
Notes: (Read only property)

6.6.246 ImageExtent as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries image extent.
Notes: See GetImageExtent function in ImageMagick documentation.
(Read only property)

6.6.247 ImageType as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image type.
Notes: See Type constants like TrueColorType for RGB images.
(Read only property)

6.6.248 Intensity as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Method to generate an intensity value from a pixel.
Notes: (Read and Write property)

6.6.249 Interlace as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The interlace setting.
Notes: (Read and Write property)

6.6.250 Interpolate as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Interpolation of color for between pixel lookups.
Notes: (Read and Write property)

6.6.251 IsBlobExempt as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is exempt.
Notes: See IsBlobExempt function in ImageMagick documentation.
(Read only property)

6.6.252 IsBlobSeekable as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is seekable.
Notes: See IsBlobSeekable function in ImageMagick documentation.
(Read only property)

6.6.253 IsBlobTemporary as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the blob is temporary.
Notes: See IsBlobTemporary function in ImageMagick documentation.
(Read only property)

6.6.254 IsImageGray as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the type of the image is grayscale or bi-level.
Notes: See IsImageGray function in ImageMagick documentation.
(Read only property)

6.6.255 IsImageMonochrome as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if type of the image is bi-level.
Notes: See IsImageMonochrome function in ImageMagick documentation.
(Read only property)

6.6.256 IsImageObject as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true if the image sequence contains a valid set of image objects.
Notes: See IsImageObject function in ImageMagick documentation.
(Read only property)

6.6.257 IsPaletteImage as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns MagickTrue if the image is PseudoClass and has 256 unique colors or less.
Notes: See IsPaletteImage function in ImageMagick documentation.
(Read only property)

6.6.258 IsTaintImage as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns true any pixel in the image has been altered since it was first constituted.
Notes: has image been modified since reading
(Read only property)

6.6.259 Iterations as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Number of interations for GIF animations.
Notes: (Read and Write property)

6.6.260 LastError as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code.
Notes: (Read and Write property)

6.6.261 LastException as IMException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception raised.
Notes: (Read and Write property)

6.6.262 ListImage as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image list links.
Notes: (Read only property)

6.6.263 Magick as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The magick module to use for encode/decode.
Notes: (Read and Write property)

6.6.264 MaskTrait as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The pixel trait for the mask.
Notes: Apply the clip or composite mask .
(Read and Write property)

6.6.265 MatteColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The matte color.
Notes: When you query this value, you get a copy in a new object.
To modify, you need to change copy and assign back.
(Read and Write property)

6.6.266 MetacontentExtent as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The size of meta content.
Notes: (Read only property)

6.6.267 Montage as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The montage setting.
Notes: (Read and Write property)

6.6.268 NextImage as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image list links.
Notes: (Read only property)

6.6.269 NumberChannels as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of channels.
Notes: (Read only property)

6.6.270 NumberMetaChannels as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of meta channels..
Notes: (Read only property)

6.6.271 Orientation as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Photo orientation of image.
Notes: (Read and Write property)

6.6.272 Page as IMRectangleInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Virtual canvas size and offset of image.
Notes: (Read and Write property)

6.6.273 Ping as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: No image data read, just attributes.
Notes: (Read only property)
See also:

- 6.6.139 Ping(path as string) as IMImage7MBS 952

6.6.274 PixelCacheFilename as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries file path for cache file.
Notes: See GetPixelCacheFilename function in ImageMagick documentation.
(Read only property)

6.6.275 PixelCacheType as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries pixel cache type.
Notes: See GetImagePixelCacheType function in ImageMagick documentation.
(Read only property)

6.6.276 PreviousImage as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image list links.
Notes: (Read only property)

6.6.277 Quality as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compression quality setting, meaning varies.
Notes: (Read and Write property)

6.6.278 ReadMask as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to read mask.
Notes: (Read and Write property)

6.6.279 ReferenceCount as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns reference count for this image.
Notes: Several image objects can share pixel memory.
Calling Modify function makes sure you have your own pixels to modify.
(Read only property)

6.6.280 Release as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to release the object in destructor.
Notes: (Read and Write property)

6.6.281 RenderingIntent as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rendering intent.
Notes: (Read and Write property)

6.6.282 Resolution as IMPointInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image resolution/density.
Notes: See also Units property for the resolution unit.

(Read and Write property)

6.6.283 Rows as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Physical height of image in pixel.
Notes: (Read only property)

6.6.284 Scene as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Index of image in multi-image file.
Notes: (Read and Write property)

6.6.285 StartLoop as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The start loop value.
Notes: For GIF.
(Read and Write property)

6.6.286 StorageClass as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The storage class of the image.
Notes: (Read only property)

6.6.287 Taint as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Has image been modified since reading.
Notes: (Read and Write property)

6.6.288 TicksPerSecond as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Units for delay time, default 100 for GIF.
Notes: (Read and Write property)

6.6.289 TileOffset as IMRectangleInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The tile offset.
Notes: (Read and Write property)

6.6.290 TotalColors as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Total number of colors.
Notes: (Read and Write property)

6.6.291 TransparentColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Color for 'transparent' color index in GIF.
Notes: When you query this value, you get a copy in a new object.
To modify, you need to change copy and assign back.
(Read and Write property)

6.6.292 Type as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image type.
Notes: (Read only property)

6.6.293 Units as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Resolution/density ppi or ppc.
Notes: (Read and Write property)

6.6.294 VirtualMetacontent as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the virtual metacontent corresponding with the last call to QueueAuthenticPixels or GetVirtualPixels.
Notes: Nil is returned if the meta-content are not available.
See GetVirtualMetacontent function in ImageMagick documentation.
(Read only property)

6.6.295 VirtualPixelQueue as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns the virtual pixels associated corresponding with the last call to QueueAuthenticPixels or GetVirtualPixels.
Notes: See GetVirtualPixelQueue function in ImageMagick documentation.
(Read only property)

6.6.296 Width as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Physical width of image in pixel.
Notes: (Read only property)

6.6.297 WriteMask as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to write mask.
Notes: (Read and Write property)

6.6.298 Constants

Constants

Constant	Value	Description
kTransparentAlpha	0	The value for transparent alpha.

Intent

Constant	Value	Description
kAbsoluteIntent	3	
kPerceptualIntent	2	
kRelativeIntent	4	
kSaturationIntent	1	
kUndefinedIntent	0	

Alpha Channel

Constant	Value	Description
kActivateAlphaChannel	1	
kAssociateAlphaChannel	2	
kBackgroundAlphaChannel	3	
kCopyAlphaChannel	4	
kDeactivateAlphaChannel	5	
kDisassociateAlphaChannel	7	
kDiscreteAlphaChannel	6	
kExtractAlphaChannel	8	
kOffAlphaChannel	9	
kOnAlphaChannel	10	
kOpaqueAlphaChannel	11	
kRemoveAlphaChannel	12	
kSetAlphaChannel	13	
kShapeAlphaChannel	14	
kTransparentAlphaChannel	15	
kUndefinedAlphaChannel	0	

Complex Operators

Constant	Value	Description
kAddComplexOperator	1	Add
kConjugateComplexOperator	2	ConjugateC
kDivideComplexOperator	3	Divide
kMagnitudePhaseComplexOperator	4	Magnitude Phase
kMultiplyComplexOperator	5	Multiply
kRealImaginaryComplexOperator	6	Real Imaginary
kSubtractComplexOperator	7	Subtract
kUndefinedComplexOperator	0	Undefined

Preview

Constant	Value	Description
kAddNoisePreview	14	
kBlurPreview	16	
kBrightnessPreview	6	
kCharcoalDrawingPreview	28	
kDespecklePreview	12	
kDullPreview	9	
kEdgeDetectPreview	18	
kGammaPreview	7	
kGrayscalePreview	10	
kHuePreview	4	
kImplodePreview	25	
kJPEGPreview	29	
kOilPaintPreview	27	
kQuantizePreview	11	
kRaisePreview	22	
kReduceNoisePreview	13	
kRollPreview	3	
kRotatePreview	1	
kSaturationPreview	5	
kSegmentPreview	23	
kShadePreview	21	
kSharpenPreview	15	
kShearPreview	2	
kSolarizePreview	20	
kSpiffPreview	8	
kSpreadPreview	19	
kSwirlPreview	24	
kThresholdPreview	17	
kUndefinedPreview	0	
kWavePreview	26	

Distortion

Constant	Value	Description
kAffineDistortion	1	
kAffineProjectionDistortion	2	
kArcDistortion	9	
kBarrelDistortion	14	
kBarrelInverseDistortion	15	
kBilinearDistortion	6	
kBilinearForwardDistortion	6	
kBilinearReverseDistortion	7	
kCylinder2PlaneDistortion	12	
kDePolarDistortion	11	
kPerspectiveDistortion	4	
kPerspectiveProjectionDistortion	5	
kPlane2CylinderDistortion	13	
kPolarDistortion	10	
kPolynomialDistortion	8	
kResizeDistortion	17	
kScaleRotateTranslateDistortion	3	
kSentinelDistortion	18	
kShepardsDistortion	16	
kUndefinedDistortion	0	

Composite Operators

Constant	Value	Description
kAlphaCompositeOp	1	
kAtopCompositeOp	2	
kBlendCompositeOp	3	
kBlurCompositeOp	4	
kBumpmapCompositeOp	5	
kChangeMaskCompositeOp	6	
kClearCompositeOp	7	
kColorBurnCompositeOp	8	
kColorDodgeCompositeOp	9	
kColorizeCompositeOp	10	
kCopyAlphaCompositeOp	17	
kCopyBlackCompositeOp	11	
kCopyBlueCompositeOp	12	
kCopyCompositeOp	13	
kCopyCyanCompositeOp	14	
kCopyGreenCompositeOp	15	
kCopyMagentaCompositeOp	16	
kCopyRedCompositeOp	18	
kCopyYellowCompositeOp	19	
kDarkenCompositeOp	20	
kDarkenIntensityCompositeOp	21	
kDifferenceCompositeOp	22	
kDisplaceCompositeOp	23	
kDissolveCompositeOp	24	
kDistortCompositeOp	25	
kDivideDstCompositeOp	26	
kDivideSrcCompositeOp	27	
kDstAtopCompositeOp	28	
kDstCompositeOp	29	
kDstInCompositeOp	30	
kDstOutCompositeOp	31	
kDstOverCompositeOp	32	
kExclusionCompositeOp	33	
kHardLightCompositeOp	34	
kHardMixCompositeOp	35	
kHueCompositeOp	36	
kInCompositeOp	37	
kIntensityCompositeOp	38	
kLightenCompositeOp	39	
kLightenIntensityCompositeOp	40	
kLinearBurnCompositeOp	41	
kLinearDodgeCompositeOp	42	
kLinearLightCompositeOp	43	
kLuminizeCompositeOp	44	
kMathematicsCompositeOp	45	
kMinusDstCompositeOp	46	
kMinusSrcCompositeOp	47	
kModulateCompositeOp	48	
kModulusAddCompositeOp	49	
kModulusSubtractCompositeOp	50	
kMultiplyCompositeOp	51	
kNoCompositeOp	52	
kOutCompositeOp	53	
kOverCompositeOp	54	
kOverlayCompositeOp	55	
kPegtopLightCompositeOp	56	
kPinLightCompositeOp	57	
kPlusCompositeOp	58	

Dispose

Constant	Value	Description
kBackgroundDispose	2	
kNoneDispose	1	
kPreviousDispose	3	
kUndefinedDispose	0	
kUnrecognizedDispose	0	

Color Interpolate

Constant	Value	Description
kBarycentricColorInterpolate	1	
kBilinearColorInterpolate	7	
kInverseColorInterpolate	19	
kManhattanColorInterpolate	20	
kPolynomialColorInterpolate	8	
kShepardsColorInterpolate	16	
kUndefinedColorInterpolate	0	
kVoronoiColorInterpolate	18	

Color Types

Constant	Value	Description
kBilevelType	1	
kColorSeparationAlphaType	9	
kColorSeparationType	8	
kGrayscaleAlphaType	3	
kGrayscaleType	2	
kOptimizeType	10	
kPaletteAlphaType	5	
kPaletteBilevelAlphaType	11	
kPaletteType	4	
kTrueColorAlphaType	7	
kTrueColorType	6	
kUndefinedType	0	

Orientations

Constant	Value	Description
kBottomLeftOrientation	4	
kBottomRightOrientation	3	
kLeftBottomOrientation	8	
kLeftTopOrientation	5	
kRightBottomOrientation	7	
kRightTopOrientation	6	
kTopLeftOrientation	1	
kTopRightOrientation	2	
kUndefinedOrientation	0	

Color Spaces

Constant	Value	Description
kCMYColorspace	1	
kCMYKColorspace	2	
kGRAYColorspace	3	
kHCLColorspace	4	
kHCLpColorspace	5	
kHSBColorspace	6	
kHSIColorspace	7	
kHSLColorspace	8	
kHSVColorspace	9	
kHWBColorspace	10	
kLabColorspace	11	
kLCHabColorspace	13	
kLCHColorspace	12	
kLCHuvColorspace	14	
kLinearGRAYColorspace	33	
kLMSColorspace	16	
kLogColorspace	15	
kLuvColorspace	17	
kOHTAColorspace	18	
kRec601YCbCrColorspace	19	
kRec709YCbCrColorspace	20	
kRGBColorspace	21	
kscRGBColorspace	22	
ksRGBColorspace	23	
kTransparentColorspace	24	
kUndefinedColorspace	0	
kxyYColorspace	25	
kXYZColorspace	26	
kYCbCrColorspace	27	
kYCCColorspace	28	
kYDbDrColorspace	29	
kYIQColorspace	30	
kYPbPrColorspace	31	
kYUVColorspace	32	

Layers

Constant	Value	Description
kCoalesceLayer	1	
kCompareAnyLayer	2	
kCompareClearLayer	3	
kCompareOverlayLayer	4	
kCompositeLayer	12	
kDisposeLayer	5	
kFlattenLayer	14	
kMergeLayer	13	
kMosaicLayer	15	
kOptimizeImageLayer	7	
kOptimizeLayer	6	
kOptimizePlusLayer	8	
kOptimizeTransLayer	9	
kRemoveDupsLayer	10	
kRemoveZeroLayer	11	
kTrimBoundsLayer	16	
kUndefinedLayer	0	

Cache

Constant	Value	Description
kDiskCache	1	
kDistributedCache	2	
kMapCache	3	
kMemoryCache	4	
kPingCache	5	
kUndefinedCache	0	

Interlace

Constant	Value	Description
kGIFInterlace	5	
kJPEGInterlace	6	
kLineInterlace	2	
kNoInterlace	1	
kPartitionInterlace	4	
kPlaneInterlace	3	
kPNGInterlace	7	
kUndefinedInterlace	0	

Threshold Methods

Constant	Value	Description
kKapurThresholdMethod	1	
kOTSUThresholdMethod	2	
kTriangleThresholdMethod	3	
kUndefinedThresholdMethod	0	

Resolutions

Constant	Value	Description
kPixelsPerCentimeterResolution	2	
kPixelsPerInchResolution	1	
kUndefinedResolution	0	

6.7 class IMImageAffineMatrix7MBS

6.7.1 class IMImageAffineMatrix7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an affine transformation matrix.

6.7.2 Methods

6.7.3 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.
Notes: Initializes with an identity matrix.

6.7.4 Properties

6.7.5 RX as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotation-x part of the affine transformation.
Notes: (Read and Write property)

6.7.6 RY as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The rotation-y part of the affine transformation.
Notes: (Read and Write property)

6.7.7 SX as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale-x part of the affine transformation.
Notes: (Read and Write property)

6.7.8 SY as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The scale-y part of the affine transformation.
Notes: (Read and Write property)

6.7.9 TX as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate-x part of the affine transformation.
Notes: (Read and Write property)

6.7.10 TY as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The translate-y part of the affine transformation.
Notes: (Read and Write property)

6.8 class IMImageInfo7MBS

6.8.1 class IMImageInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for image info.
Notes: This represents usually an image you want to load or save.
Blog Entries

- MBS Xojo Plugins, version 20.0pr5

6.8.2 Methods

6.8.3 BlobToImage(Data as MemoryBlock) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: BlobToImage implements direct to memory image formats. It returns the blob as an image.
Notes: See BlobToImage function in ImageMagick documentation.
See also:

- 6.8.4 BlobToImage(Data as String) as IMImage7MBS 1002

6.8.4 BlobToImage(Data as String) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: BlobToImage implements direct to memory image formats. It returns the blob as an image.
Notes: See BlobToImage function in ImageMagick documentation.
See also:

- 6.8.3 BlobToImage(Data as MemoryBlock) as IMImage7MBS 1002

6.8.5 Clone as IMImageInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Makes a copy of the given image info object.
Notes: See CloneImageInfo function in ImageMagick documentation.

6.8.6 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty image info.
Notes: See CloneImageInfo function in ImageMagick documentation.
See also:

- 6.8.7 Constructor(ImageInfo as IMImageInfo7MBS) 1003

6.8.7 Constructor(ImageInfo as IMImageInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of an existing image info.
Notes: See CloneImageInfo function in ImageMagick documentation.
See also:

- 6.8.6 Constructor 1002

6.8.8 PingBlob(Data as MemoryBlock) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pings a blob.
Notes: Reads image attributes, but not image data.
See PingBlob function in ImageMagick documentation.
See also:

- 6.8.9 PingBlob(Data as String) as IMImage7MBS 1003

6.8.9 PingBlob(Data as String) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pings a blob.
Notes: Reads image attributes, but not image data.
See PingBlob function in ImageMagick documentation.
See also:

- 6.8.8 PingBlob(Data as MemoryBlock) as IMImage7MBS 1003

6.8.10 PingImage as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns all the properties of an image or image sequence except for the pixels.
Notes: It is much faster and consumes far less memory than ReadImage. On failure, a nil image is returned

and exception describes the reason for the failure.

See PingImage function in ImageMagick documentation.

6.8.11 PingImages(filename as String) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Pings one or more images and returns them as an image list.
Notes: See xx function in ImageMagick documentation.

6.8.12 ReadImage as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads an image or image sequence from a file or file handle.
Notes: The method returns a nil if there is a memory shortage or if the image cannot be read. On failure, a nil image is returned and exception describes the reason for the failure.

See ReadImage function in ImageMagick documentation.

6.8.13 ReadImages(filename as String) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads one or more images and returns them as an image list.
Notes: See xx function in ImageMagick documentation.

6.8.14 ReadInlineImage(filename as String) as IMImage7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Reads a Base64-encoded inline image or image sequence.
Notes: The method returns a nil if there is a memory shortage or if the image cannot be read. On failure, a nil image is returned and exception describes the reason for the failure.

See ReadInlineImage function in ImageMagick documentation.

6.8.15 WriteImage(image as IMImage7MBS) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Writes an image or an image sequence to a file or file handle.
Notes: If writing to a file is on disk, the name is defined by the filename member of the image object. WriteImage returns False is there is a memory shortage or if the image cannot be written. Check the exception member of image to determine the cause for any failure.

See WriteImage function in ImageMagick documentation.

6.8.16 WriteImages(image as IMImage7MBS, filename as String) as boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Writes an image sequence into one or more files.
Notes: While WriteImage can write an image sequence, it is limited to writing the sequence into a single file using a format which supports multiple frames. WriteImages, however, does not have this limitation, instead it generates multiple output files if necessary (or when requested). When ImageInfo's adjoin flag is set to False, the file name is expected to include a printf-style formatting string for the frame number (e.g. "image02d.png").

See WriteImages function in ImageMagick documentation.

6.8.17 Properties

6.8.18 Adjoin as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Save images to separate scene files.
Notes: (Read and Write property)

6.8.19 Affirm as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Affirm flag.
Notes: (Read and Write property)

6.8.20 AlphaColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha color value.
Notes: (Read and Write property)

6.8.21 Antialias as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The anti alias flag.
Notes: (Read and Write property)

6.8.22 BackgroundColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The background color.
Notes: (Read and Write property)

6.8.23 BorderColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The border color.
Notes: (Read and Write property)

6.8.24 Channel as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The channel types.
Notes: See ChannelType constants.
(Read and Write property)

6.8.25 ColorSpace as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Colorspace of image data.
Notes: See Colorspace constants.
(Read and Write property)

6.8.26 Compose as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Alpha composition method for layered images.
Notes: (Read and Write property)

6.8.27 Compression as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compression of image when read/write.
Notes: (Read and Write property)

6.8.28 Debug as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Debug enabled.
Notes: (Read and Write property)

6.8.29 Density as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: DUP for image and draw_info.
Notes: (Read and Write property)

6.8.30 Depth as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Depth of image on read/write.
Notes: (Read and Write property)

6.8.31 Dither as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: dithering on/off.
Notes: (Read and Write property)

6.8.32 Endian as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: raw data integer ordering on read/write.
Notes: (Read and Write property)

6.8.33 Extract as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Crop/resize string on image read.
Notes: (Read and Write property)

6.8.34 Filename as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The file path of the image.
Notes: (Read and Write property)

6.8.35 Font as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Font name for drawing functions.
Notes: (Read and Write property)

6.8.36 Fuzz as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Current color fuzz attribute.
Notes: (Read and Write property)

6.8.37 Handle as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.8.38 Interlace as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Interlace for image write.
Notes: See Interlace constants.
(Read and Write property)

6.8.39 LastError as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code.
Notes: (Read and Write property)

6.8.40 LastException as IMException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception raised.
Notes: (Read and Write property)

6.8.41 Length as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Length of data.
Notes: (Read and Write property)

6.8.42 Magick as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The magick module to use for encode/decode.
Notes: (Read and Write property)

6.8.43 MatteColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The matte color.
Notes: (Read and Write property)

6.8.44 Monochrome as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to read/write pcl,pdf,ps,xps as monocrome image.
Notes: (Read and Write property)

6.8.45 Orientation as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The orientation of the image.
Notes: See orientation constants.
(Read and Write property)

6.8.46 Page as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The page number.
Notes: (Read and Write property)

6.8.47 Ping as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: If enabled only fast read image attributes, not image data.
Notes: (Read and Write property)

6.8.48 PointSize as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The point size.
Notes: (Read and Write property)

6.8.49 Quality as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Compression quality setting, meaning varies.
Notes: (Read and Write property)

6.8.50 Release as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Wether to release the object in destructor.
Notes: (Read and Write property)

6.8.51 SamplingFactor as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: JPEG write sampling factor.
Notes: (Read and Write property)

6.8.52 Scene as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Starting value for image save numbering.
Notes: (Read and Write property)

6.8.53 SceneCount as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Total number of images in list - for escapes.
Notes: (Read and Write property)

6.8.54 Scenes as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scene numbers that is to be read in.
Notes: (Read and Write property)

6.8.55 ServerName as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: X windows server name - display/animate.
Notes: (Read and Write property)

6.8.56 Size as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Image generation size.
Notes: (Read and Write property)

6.8.57 Synchronize as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The Synchronize flag.
Notes: (Read and Write property)

6.8.58 Temporary as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether image file is temporary.
Notes: image file to be deleted after read ”empemeral:”.
(Read and Write property)

6.8.59 Texture as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: montage/display background tile.
Notes: (Read and Write property)

6.8.60 TransparentColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: color for transparent index in color tables.
Notes: (Read and Write property)

6.8.61 Type as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The image type.
Notes: e.g. TrueColorAlphaType.
(Read and Write property)

6.8.62 Unique as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Unique tempory filename - delegates.
Notes: (Read and Write property)

6.8.63 Units as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The resolution unit.
Notes: Density pixels/inch or pixel/cm.
Either kPixelsPerInchResolution or kPixelsPerCentimeterResolution.

(Read and Write property)

6.8.64 Verbose as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Verbose output enable/disable.
Notes: (Read and Write property)

6.9 class IMKernelInfo7MBS

6.9.1 class IMKernelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The kernel information details.

6.9.2 Methods

6.9.3 Clone as IMKernelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Clones the kernel info object.
Notes: Creates a new clone of the given Kernel List so that its can be modified without effecting the original.

6.9.4 Constructor(KernelInfo as IMKernelInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the kernel.
See also:

- 6.9.5 Constructor(kernelString as String) 1015
- 6.9.6 Constructor(Type as Integer, GeometryInfo as IMGeometryInfo7MBS) 1017

6.9.5 Constructor(kernelString as String)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Takes the given string (generally supplied by the user) and converts it into a Morphology/Convolution Kernel.
Notes: This allows users to specify a kernel from a number of pre-defined kernels, or to fully specify their own kernel for a specific Convolution or Morphology Operation.

The kernel so generated can be any rectangular array of floating point values (doubles) with the 'control point' or 'pixel being affected' anywhere within that array of values.

Previously IM was restricted to a square of odd size using the exact center as origin, this is no longer the case, and any rectangular kernel with any value being declared the origin. This in turn allows the use of highly asymmetrical kernels.

The floating point values in the kernel can also include a special value known as 'nan' or 'not a number' to indicate that this value is not part of the kernel array. This allows you to shaped the kernel within its rectangular area. That is 'nan' values provide a 'mask' for the kernel shape. However at least one non-nan value must be provided for correct working of a kernel.

The returned kernel should be freed using the DestroyKernelInfo() when you are finished with it. Do not free this memory yourself.

Input kernel defintion strings can consist of any of three types.

"name:args [[@><] " Select from one of the built in kernels, using the name and geometry arguments supplied. See AcquireKernelBuiltIn()

"WxH [+X+Y] [@><] :num, num, num ..." a kernel of size W by H, with W*H floating point numbers following. the 'center' can be optionally be defined at +X+Y (such that +0+0 is top left corner). If not defined the pixel in the center, for odd sizes, or to the immediate top or left of center for even sizes is automatically selected.

"num, num, num, num, ..." list of floating point numbers defining an 'old style' odd sized square kernel. At least 9 values should be provided for a 3x3 square kernel, 25 for a 5x5 square kernel, 49 for 7x7, etc. Values can be space or comma separated. This is not recommended.

You can define a 'list of kernels' which can be used by some morphology operators A list is defined as a semi-colon separated list kernels.

" kernel ; kernel ; kernel ; "

Any extra ';' characters, at start, end or between kernel defintions are simply ignored.

The special flags will expand a single kernel, into a list of rotated kernels. A '@' flag will expand a 3x3 kernel into a list of 45-degree cyclic rotations, while a '>' will generate a list of 90-degree rotations. The '<' also exands using 90-degree rotates, but giving a 180-degree reflected kernel before the +/- 90-degree rotations, which can be important for Thinning operations.

Note that 'name' kernels will start with an alphabetic character while the new kernel specification has a ':' character in its specification string. If neither is the case, it is assumed an old style of a simple list of numbers generating a odd-sized square kernel has been given.

kernelString: the Morphology/Convolution kernel wanted.

See also AcquireKernelInfo in ImageMagick documentation.
See also:

- 6.9.4 Constructor(KernelInfo as IMKernelInfo7MBS) 1015
- 6.9.6 Constructor(Type as Integer, GeometryInfo as IMGeometryInfo7MBS) 1017

6.9.6 Constructor(Type as Integer, GeometryInfo as IMGeometryInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Returns one of the 'named' built-in types of kernels used for special purposes such as gaussian blurring, skeleton pruning, and edge distance determination.
Notes: They take a KernelType, and a set of geometry style arguments, which were typically decoded from a user supplied string, or from a more complex Morphology Method that was requested.

A description of each parameter follows:

type: the pre-defined type of kernel wanted
args: arguments defining or modifying the kernel

Convolution Kernels
Unity The a No-Op or Scaling single element kernel.
Gaussian: { radius } , { sigma } Generate a two-dimensional gaussian kernel, as used by -gaussian. The sigma for the curve is required. The resulting kernel is normalized,
If 'sigma' is zero, you get a single pixel on a field of zeros.
NOTE: that the 'radius' is optional, but if provided can limit (clip) the final size of the resulting kernel to a square 2*radius+1 in size. The radius should be at least 2 times that of the sigma value, or sever clipping and aliasing may result. If not given or set to 0 the radius will be determined so as to produce the best minimal error result, which is usally much larger than is normally needed.
LoG: { radius } , { sigma } "Laplacian of a Gaussian" or "Mexician Hat" Kernel. The supposed ideal edge detection, zero-summing kernel.
An alturnative to this kernel is to use a "DoG" with a sigma ratio of approx 1.6 (according to wikipedia).
DoG: { radius } , { sigma1 } , { sigma2 } "Difference of Gaussians" Kernel. As "Gaussian" but with a gaussian produced by 'sigma2' subtracted from the gaussian produced by 'sigma1'. Typically sigma2 >sigma1. The result is a zero-summing kernel.
Blur: { radius } , { sigma } [, { angle }] Generates a 1 dimensional or linear gaussian blur, at the angle given (current restricted to orthogonal angles). If a 'radius' is given the kernel is clipped to a width of 2*radius+1. Kernel can be rotated by a 90 degree angle.
If 'sigma' is zero, you get a single pixel on a field of zeros.
Note that two convolutions with two "Blur" kernels perpendicular to each other, is equivalent to a far larger "Gaussian" kernel with the same sigma value, However it is much faster to apply. This is how the "-blur" operator actually works.
Comet: { width } , { sigma } , { angle } Blur in one direction only, much like how a bright object leaves a

comet like trail. The Kernel is actually half a gaussian curve, Adding two such blurs in opposite directions produces a Blur Kernel. Angle can be rotated in multiples of 90 degrees.
Note that the first argument is the width of the kernel and not the radius of the kernel.
Binomial: [{ radius }] Generate a discrete kernel using a 2 dimentional Pascel's Triangle of values. Used for special forma of image filters.
See also:

- 6.9.4 Constructor(KernelInfo as IMKernelInfo7MBS) 1015
- 6.9.5 Constructor(kernelString as String) 1015

6.9.7 Scale(scaleFactor as double, GeometryFlags as integer)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales the given kernel.
Notes: Scales the given kernel list by the given amount, with or without normalization of the sum of the kernel values (as per given flags).

By default (no flags given) the values within the kernel is scaled directly using given scaling factor without change.

If either of the two 'normalize_flags' are given the kernel will first be normalized and then further scaled by the scaling factor value given.

Kernel normalization ('normalize_flags' given) is designed to ensure that any use of the kernel scaling factor with 'Convolve' or 'Correlate' morphology methods will fall into -1.0 to +1.0 range. Note that for non-HDRI versions of IM this may cause images to have any negative results clipped, unless some 'bias' is used.

More specifically. Kernels which only contain positive values (such as a 'Gaussian' kernel) will be scaled so that those values sum to +1.0, ensuring a 0.0 to +1.0 output range for non-HDRI images.

For Kernels that contain some negative values, (such as 'Sharpen' kernels) the kernel will be scaled by the absolute of the sum of kernel values, so that it will generally fall within the +/- 1.0 range.

For kernels whose values sum to zero, (such as 'Laplician' kernels) kernel will be scaled by just the sum of the postive values, so that its output range will again fall into the +/- 1.0 range.

For special kernels designed for locating shapes using 'Correlate', (often only containing +1 and -1 values, representing foreground/brackground matching) a special normalization method is provided to scale the positive values separately to those of the negative values, so the kernel will be forced to become a zero-sum kernel better suited to such searches.

WARNING: Correct normalization of the kernel assumes that the '*_range' attributes within the kernel structure have been correctly set during the kernels creation.

NOTE: The values used for 'normalize_flags' have been selected specifically to match the use of geometry options, so that '!' means NormalizeValue, '^' means CorrelateNormalizeValue. All other GeometryFlags values are ignored.

scaleFactor: zero. If the kernel is normalized regardless of any flags.
GeometryFlags: specifically: NormalizeValue, CorrelateNormalizeValue, and/or PercentValue.

See also ScaleKernelInfo in ImageMagick documentation.

6.9.8 ScaleGeometry(Geometry as string)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Scales using a geometry.
Notes: Takes a geometry argument string, typically provided as a ”-set option:convolve:scale { geometry } ” user setting, and modifies the kernel according to the parsed arguments of that setting.

The first argument (and any normalization flags) are passed to ScaleKernelInfo() to scale/normalize the kernel. The second argument is then passed to UnityAddKernelInfo() to add a scled unity kernel into the scaled/normalized kernel.

geometry: ”-set option:convolve:scale { geometry } ” setting.

See also ScaleGeometryKernelInfo in ImageMagick documentation.

6.9.9 UnityAddKernelInfo(scale as double)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Adds a given amount of the 'Unity' Convolution Kernel to the given pre-scaled and normalized Kernel.
Notes: This in effect adds that amount of the original image into the resulting convolution kernel. This value is usually provided by the user as a percentage value in the 'convolve:scale' setting.

The resulting effect is to convert the defined kernels into blended soft-blurs, unsharp kernels or into sharpening kernels.

See also UnityAdditionKernelInfo in ImageMagick documentation.

6.9.10 Properties

6.9.11 Angle as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The angle parameter.
Notes: (Read and Write property)

6.9.12 Handle as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.9.13 Height as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height value.
Notes: (Read and Write property)

6.9.14 LastError as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Last error code.
Notes: (Read and Write property)

6.9.15 LastException as IMException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Last exception raised.
Notes: (Read and Write property)

6.9.16 Maximum as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Maximum parameter.
Notes: (Read and Write property)

6.9.17 Minimum as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Minimum parameter.
Notes: (Read and Write property)

6.9.18 NegativeRange as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The negative range parameter.
Notes: (Read and Write property)

6.9.19 NextKernel as IMKernelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Next kernel in a list of kernels.
Notes: (Read only property)

6.9.20 PositiveRange as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The postive range parameter.
Notes: (Read and Write property)

6.9.21 Release as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to release to kernel info structure in destructor.
Notes: (Read and Write property)

6.9.22 Type as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of kernel.
Notes: See Kernel constants.
(Read and Write property)

6.9.23 Values as Ptr

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The pointer to the values.
Notes: (Read only property)

6.9.24 Width as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Width value.
Notes: (Read and Write property)

6.9.25 X as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: X coordinate.
Notes: (Read and Write property)

6.9.26 Y as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Y coordinate.
Notes: (Read and Write property)

6.9.27 Constants

Kernel

Constant	Value	Description
KernelBinomial	7	Binomial
KernelBlur	5	Convolution Kernels, Gaussian Based.
KernelChebyshev	33	One Distance Measuring Kernel.
KernelComet	6	Convolution Kernels, Gaussian Based.
KernelCompass	13	Convolution Kernel.
KernelConvexHull	30	A Hit And Miss Kernel.
KernelCorners	25	A Hit And Miss Kernel.
KernelCross	21	A shape kernel.
KernelDiagonals	26	A Hit And Miss Kernel.
KernelDiamond	15	A shape kernel.
KernelDisk	19	A shape kernel.
KernelDoG	3	Convolution Kernels, Gaussian Based.
KernelEdges	24	A Hit And Miss Kernel.
KernelEuclidean	36	One Distance Measuring Kernel.
KernelFreiChen	10	Convolution Kernel.
KernelGaussian	2	Convolution Kernels, Gaussian Based.
KernelKirsch	14	Convolution Kernel.
KernelLaplacian	8	Convolution Kernel.
KernelLineEnds	27	A Hit And Miss Kernel.
KernelLineJunctions	28	A Hit And Miss Kernel.
KernelLoG	4	Convolution Kernels, Gaussian Based.
KernelManhattan	34	One Distance Measuring Kernel.
KernelOctagon	18	A shape kernel.
KernelOctagonal	35	One Distance Measuring Kernel.
KernelPeaks	23	A Hit And Miss Kernel.
KernelPlus	20	A shape kernel.
KernelPrewitt	12	Convolution Kernel.
KernelRectangle	17	A shape kernel.
KernelRidges	29	A Hit And Miss Kernel.
KernelRing	22	A shape kernel.
KernelRoberts	11	Convolution Kernel.
KernelSkeleton	32	A Hit And Miss Kernel.
KernelSobel	9	Convolution Kernel.
KernelSquare	16	A shape kernel.
KernelThinSE	31	A Hit And Miss Kernel.
KernelUndefined	0	equivalent to Unity Kernel
KernelUnity	1	The no-op or 'original image' kernel
KernelUserDefinedKernel	37	User Specified Kernel Array

Morphology

Constant	Value	Description
MorphologyBottomHat	17	
MorphologyClose	9	
MorphologyCloseIntensity	11	
MorphologyConvolve	1	
MorphologyCorrelate	2	
MorphologyDilate	4	
MorphologyDilateIntensity	6	
MorphologyDistance	21	
MorphologyEdge	15	
MorphologyEdgeIn	13	
MorphologyEdgeOut	14	
MorphologyErode	3	
MorphologyErodeIntensity	5	
MorphologyHitAndMiss	18	
MorphologyIterativeDistance	7	
MorphologyOpen	8	
MorphologyOpenIntensity	10	
MorphologySmooth	12	
MorphologyThicken	20	
MorphologyThinning	19	
MorphologyTopHat	16	
MorphologyUndefined	0	
MorphologyVoronoi	22	

6.10 class IMMagickInfo7MBS

6.10.1 class IMMagickInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an encoder/decoder module.

6.10.2 Properties

6.10.3 Description as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The description field.
Notes: e.g. ”Joint Photographic Experts Group JFIF format” for JPEG format.
(Read only property)

6.10.4 flags as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Various flags.
Notes: (Read only property)

6.10.5 FormatType as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of format defined.
Notes: (Read only property)

6.10.6 MimeType as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The mime type.
Notes: e.g. ”image/jpeg” for JPEG format.
(Read only property)

6.10.7 ModuleName as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The module name.
Notes: (Read only property)

6.10.8 Name as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The name of the encoder/decoder.
Notes: (Read only property)

6.10.9 Note as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The note for this magick.
Notes: (Read only property)

6.10.10 Version as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The version number.
Notes: e.g. ”libjpeg 90” for JPEG library.
(Read only property)

6.10.11 Constants

Flags

Constant	Value	Description
FlagAdjoin	&h0001	Supports adjoin.
FlagBlobSupport	&h0002	Supports blob handling, e.g. reading image from memory.
FlagDecoderSeekableStream	&h0200	Decoder supports seakable streams.
FlagDecoderThreadSupport	&h0004	Supports decoding with threads.
FlagEncoderSeekableStream	&h0400	Encoder supports seekable streams.
FlagEncoderThreadSupport	&h0008	Supports encoding with threads.
FlagEndianSupport	&h0010	Supports endian.
FlagNone	&h0000	No flags set.
FlagRawSupport	&h0020	Whether RAW processing is supported.
FlagSeekableStream	&h0040	Supports seakable streams.
FlagStealth	&h0080	Stealth flag (hidden).
FlagUseExtension	&h0100	Uses extension.

Format Types

Constant	Value	Description
FormatTypeExplicit	2	Explicit defined format type.
FormatTypeImplicit	1	Implicit defined format type.
FormatTypeUndefined	0	Not defined.

6.11 class IMMagickInfoList7MBS

6.11.1 class IMMagickInfoList7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for an array of magick info objects.

6.11.2 Methods

6.11.3 Item(index as integer) as IMMagickInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Queries an item by the index.
Notes: Index is zero based to Count-1.

6.11.4 Properties

6.11.5 Count as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of entries.
Notes: (Read only property)

6.11.6 Handle as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read only property)

6.12 class IMMissingFunctionException7MBS

6.12.1 class IMMissingFunctionException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The exception for a missing function.
Notes: Subclass of the RuntimeException class.

6.13 class IMMontageInfo7MBS

6.13.1 class IMMontageInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for montage settings.

6.13.2 Methods

6.13.3 Clone(ImageInfo as IMImageInfo7MBS) as IMMontageInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the montage settings.

6.13.4 Close

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Closes object.
Notes: Same as destructor.

6.13.5 Constructor(ImageInfo as IMImageInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

6.13.6 Properties

6.13.7 AlphaColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha color.
Notes: (Read and Write property)

6.13.8 BackgroundColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The background color.
Notes: (Read and Write property)

6.13.9 BorderColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The border color.
Notes: (Read and Write property)

6.13.10 BorderWidth as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The border width.
Notes: (Read and Write property)

6.13.11 Debug as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to output debug messages.
Notes: (Read and Write property)

6.13.12 Filename as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The filename.
Notes: (Read and Write property)

6.13.13 Fill as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The fill color.
Notes: (Read and Write property)

6.13.14 Font as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The font name.
Notes: (Read and Write property)

6.13.15 Frame as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The frame.
Notes: (Read and Write property)

6.13.16 Geometry as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The geometry.
Notes: (Read and Write property)

6.13.17 Gravity as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The gravity setting.
Notes: (Read and Write property)

6.13.18 Handle as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.13.19 LastError as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code.
Notes: (Read and Write property)

6.13.20 LastException as IMException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception raised.
Notes: (Read and Write property)

6.13.21 MatteColor as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The matte color.
Notes: (Read and Write property)

6.13.22 PointSize as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The point size.
Notes: (Read and Write property)

6.13.23 Release as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to release object in destructor.
Notes: (Read and Write property)

6.13.24 Shadow as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to use shadow.
Notes: (Read and Write property)

6.13.25 Stroke as IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The stroke color.
Notes: (Read and Write property)

6.13.26 Texture as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The texture string.
Notes: (Read and Write property)

6.13.27 Tile as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The tile string.
Notes: (Read and Write property)

6.13.28 Title as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The title string.
Notes: (Read and Write property)

6.13.29 Constants

Montage Modes

Constant	Value	Description
MontageModeConcatenate	3	Concatenate
MontageModeFrame	1	Frame
MontageModeUndefined	0	Undefined
MontageModeUnframe	2	Unframe

6.14 class IMOptionInfo7MBS

6.14.1 class IMOptionInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for options.

6.14.2 Methods

6.14.3 CommandOptionToMnemonic(option as integer, type as integer) as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Queries command name for a given option number.

6.14.4 GetCommandOptionInfo(name as string) as IMOptionInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Queries option info for given command code.

6.14.5 IsCommandOption(name as string) as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Whether the given option is a command option.

6.14.6 Properties

6.14.7 Flags as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The flags.
Notes: (Read and Write property)

6.14.8 Mnemonic as String

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The name of the option.
Notes: (Read and Write property)

6.14.9 Stealth as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The stealth property.
Notes: This option is hidden.
(Read and Write property)

6.14.10 Type as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The type of option.
Notes: (Read and Write property)

6.14.11 Constants

Validate Types

Constant	Value	Description
AllValidate	&h7fffffff	
ColorspaceValidate	&h00001	
CompareValidate	&h00002	
CompositeValidate	&h00004	
ConvertValidate	&h00008	
FormatsDiskValidate	&h00010	
FormatsMapValidate	&h00020	
FormatsMemoryValidate	&h00040	
IdentifyValidate	&h00080	
ImportExportValidate	&h00100	
MagickValidateOptions	74	
MontageValidate	&h00200	
NoValidate	&h00000	
StreamValidate	&h00400	
UndefinedValidate	0	

Flags

Constant	Value	Description
AlwaysInterpretArgsFlag	&h0400	Always Interpret escapes in Args. CF: ”convert” compatibility mode
DeprecateOptionFlag	&h4000	Deprecate option (no code).
DrawInfoOptionFlag	&h0002	Setting stored in DrawInfo
FireOptionFlag	&h2000	Convert operation seq firing point
GenesisOptionFlag	&h0080	MagickCommandGenesis() Only Option
GlobalOptionFlag	&h0008	Global Setting or Control
ImageInfoOptionFlag	&h0001	Setting stored in ImageInfo
ListOperatorFlag	&h0040	Multi-Image processing operator
NeverInterpretArgsFlag	&h0800	Never Interpret escapes in Args. EG: filename, or delayed escapes.
NoImageOperatorFlag	&h0010	Images not required operator
NonMagickOptionFlag	&h1000	Option not used by Magick Command.
QuantizeInfoOptionFlag	&h0004	Setting stored in QuantizeInfo
ReplacedOptionFlag	&h8800	Replaced Option (but still works).
SettingOptionFlags	&h000F	mask any setting option
SimpleOperatorFlag	&h0020	Simple Image processing operator
SpecialOptionFlag	&h0100	Operator with Special Requirements. EG: for specific CLI commands
UndefinedOptionFlag	&h0000	option flag is not in use

Command Options

Constant	Value	Description
MagickAlignOptions	0	
MagickAlphaChannelOptions	1	
MagickAutoThresholdOptions	77	
MagickBooleanOptions	2	
MagickCacheOptions	3	
MagickChannelOptions	4	
MagickClassOptions	5	
MagickCLIOptions	79	
MagickClipPathOptions	6	
MagickCoderOptions	7	
MagickColorOptions	8	
MagickColorspaceOptions	9	
MagickCommandOptions	10	
MagickComplexOptions	11	
MagickComplianceOptions	12	
MagickComposeOptions	13	
MagickCompressOptions	14	
MagickConfigureOptions	15	
MagickDataTypeOptions	16	
MagickDebugOptions	17	
MagickDecorateOptions	18	
MagickDelegateOptions	19	
MagickDirectionOptions	20	
MagickDisposeOptions	21	
MagickDistortOptions	22	
MagickDitherOptions	23	
MagickEndianOptions	24	
MagickEvaluateOptions	25	
MagickFillRuleOptions	26	
MagickFilterOptions	27	
MagickFontOptions	28	
MagickFontsOptions	29	
MagickFormatOptions	30	
MagickFunctionOptions	31	
MagickGradientOptions	32	
MagickGravityOptions	33	
MagickIntensityOptions	34	
MagickIntentOptions	35	
MagickInterlaceOptions	36	
MagickInterpolateOptions	37	
MagickKernelOptions	38	
MagickLayerOptions	39	
MagickLineCapOptions	40	
MagickLineJoinOptions	41	
MagickListOptions	42	
MagickLocaleOptions	43	
MagickLogEventOptions	44	
MagickLogOptions	45	
MagickMagicOptions	46	
MagickMethodOptions	47	
MagickMetricOptions	48	
MagickMimeOptions	49	
MagickModeOptions	50	
MagickModuleOptions	51	
MagickMorphologyOptions	52	
MagickNoiseOptions	53	
MagickOrientationOptions	54	
MagickPixelChannelOptions	55	

6.15 class IMPixelInfo7MBS

6.15.1 class IMPixelInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for a pixel info.
Blog Entries

- MBS Xojo Plugins, version 20.5pr6

6.15.2 Methods

6.15.3 Clone as IMPixelInfo7MBS

Plugin Version: 20.5, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a copy of the object.

6.15.4 Constructor(Image as IMImage7MBS = nil)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The constructor.

6.15.5 Properties

6.15.6 alpha as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha quantum.
Notes: Defaults to opaque value, e.g. 65535 for 16 bit.
(Read and Write property)

6.15.7 AlphaTrait as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The alpha channel pixel trait.
Notes: If zero, no alpha.

(Read and Write property)

6.15.8 black as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The black color quantum.
Notes: (Read and Write property)

6.15.9 blue as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The blue color quantum.
Notes: (Read and Write property)

6.15.10 ColorSpace as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color space.
Notes: Defaults to ksRGBColorspace.
(Read and Write property)

6.15.11 Count as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The count value.
Notes: (Read and Write property)

6.15.12 Depth as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color depth.
Notes: Defaults to quantum depth.
(Read and Write property)

6.15.13 fuzz as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The fuzz value.
Notes: (Read and Write property)

6.15.14 green as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The green color quantum.
Notes: (Read and Write property)

6.15.15 index as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The index quantum.
Notes: (Read and Write property)

6.15.16 red as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The red color quantum.
Notes: (Read and Write property)

6.15.17 StorageClass as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The storage class.
Notes: Defaults to kDirectClass.
(Read and Write property)

6.16 class IMPointInfo7MBS

6.16.1 class IMPointInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for a x/y point.

6.16.2 Methods

6.16.3 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the point object with 0/0 values.
See also:

- 6.16.4 Constructor(X as Double, Y as Double) 1043

6.16.4 Constructor(X as Double, Y as Double)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the point object with given values.
See also:

- 6.16.3 Constructor 1043

6.16.5 Properties

6.16.6 X as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x coordinate.
Notes: (Read and Write property)

6.16.7 Y as Double

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y coordinate.
Notes: (Read and Write property)

6.17 class IMQuantizeInfo7MBS

6.17.1 class IMQuantizeInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for quantize settings.
Blog Entries

- MBS Xojo Plugins, version 20.0pr5

6.17.2 Methods

6.17.3 Clone as IMQuantizeInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Makes a copy of the given quantize info object.
Notes: See CloneQuantizeInfo function in ImageMagick documentation.

6.17.4 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates an empty quantize info.
Notes: See AcquireQuantizeInfo function in ImageMagick documentation.
See also:

- 6.17.5 Constructor(ImageInfo as IMImageInfo7MBS) 1045
- 6.17.6 Constructor(QuantizeInfo as IMQuantizeInfo7MBS) 1046

6.17.5 Constructor(ImageInfo as IMImageInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Creates a new quantize info.
Notes: If ImageInfo is not empty, copies a few settings from there.
See AcquireQuantizeInfo function in ImageMagick documentation.
See also:

- 6.17.4 Constructor 1045
- 6.17.6 Constructor(QuantizeInfo as IMQuantizeInfo7MBS) 1046

6.17.6 Constructor(QuantizeInfo as IMQuantizeInfo7MBS)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Makes a copy of the given quantize info object.
Notes: See CloneQuantizeInfo function in ImageMagick documentation.
See also:

- 6.17.4 Constructor 1045
- 6.17.5 Constructor(ImageInfo as IMImageInfo7MBS) 1045

6.17.7 Properties

6.17.8 ColorSpace as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The color space.
Notes: (Read and Write property)

6.17.9 DitherMethod as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The dither method.
Notes: (Read and Write property)

6.17.10 Handle as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The internal object reference.
Notes: (Read and Write property)

6.17.11 LastError as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last error code.
Notes: (Read and Write property)

6.17.12 LastException as IMException7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The last exception raised.
Notes: (Read and Write property)

6.17.13 MeasureError as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to measure error.
Notes: (Read and Write property)

6.17.14 NumberColors as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The number of colors.
Notes: (Read and Write property)

6.17.15 Release as Boolean

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Whether to release memory in destructor.
Notes: (Read and Write property)

6.17.16 TreeDepth as UInt64

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Tree depth.
Notes: (Read and Write property)

6.17.17 Constants

Dither Methods

Constant	Value	Description
DitherMethodFloydSteinberg	3	FloydSteinberg
DitherMethodNo	1	No
DitherMethodRiemersma	2	Riemersma
DitherMethodUndefined	0	Undefined

6.18 class IMRectangleInfo7MBS

6.18.1 class IMRectangleInfo7MBS

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The class for a rectangle.

6.18.2 Methods

6.18.3 Constructor

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the point object with zero values.
See also:

- 6.18.4 Constructor(X as Integer, Y as Integer, Width as Integer, Heigth as Integer) 1049

6.18.4 Constructor(X as Integer, Y as Integer, Width as Integer, Heigth as Integer)

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: Initializes the point object with given values.
See also:

- 6.18.3 Constructor 1049

6.18.5 Properties

6.18.6 Height as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The height of the rectangle.
Notes: (Read and Write property)

6.18.7 Width as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The width of the rectangle.
Notes: (Read and Write property)

6.18.8 X as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The x coordinate of the rectangle.
Notes: (Read and Write property)

6.18.9 Y as Integer

Plugin Version: 20.0, Platforms: macOS, Linux, Windows, Targets: All.

Function: The y coordinate of the rectangle.
Notes: (Read and Write property)

Chapter 7

List of Questions in the FAQ

Chapter 8

The FAQ

8.0.1 Can anyone help me convert seconds to time in this format hh:mm:ss?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Sure, here's a routine I use (which has an advantage over the previously-posted Date-based solution in that you don't have to rely on the creation of an object – all that happens is some division and string concatenation):
Example:

```
Function SecsToTimeString(timeInSecs as Integer, padHours as boolean, padMinutes as boolean) as string
// Given an amount time (in seconds), generates a string representing that amount
// of time. The padHours and padMinutes parameters determine whether to display
// hours and minutes if their values are zero.

// Examples:
// timeInSecs = 90, padHours = true; returns "00:01:30"
// timeInSecs = 1, padHours = false, padMinutes = true; returns "00:01"
// timeInSecs = 3601, padMinutes = false; returns "01:00:01"

dim hours, minutes, seconds as Integer
dim hoursString, minutesString as string

hours = timeInSecs / 3600
minutes = (timeInSecs mod 3600) / 60
seconds = timeInSecs mod 60

if hours = 0 then
if padHours then
hoursString = "00:"
else
hoursString = ""
end if
```

```
else
hoursString = Format(hours, "##\:")
end if
if minutes = 0 then
if hours <>0 or padMinutes then
minutesString = "00:"
else
minutesString = ""
end if
else
minutesString = Format(minutes, "00\:")
end if

return hoursString + minutesString + Format(seconds, "00")
End Function
```

Notes: (from the rb mailinglist)

8.0.2 Do you have plugins for Android?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Since there is no plugin SDK for Android, we have no way to make a plugin for Android.
Notes: We support macOS, Windows, Linux and iOS.

8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use functions from NSColor to get proper highlight color in RGB:
Example:

```
Function ProperHighlightColor(active as Boolean) As Color
#if TargetCocoa
Dim theColor As NSColorMBS
If active Then
theColor = NSColorMBS.alternateSelectedControlColor
Else
theColor = NSColorMBS.secondarySelectedControlColor
End If

Dim rgbColor As NSColorMBS = theColor.colorUsingColorSpaceName(NSColorSpaceMBS.NSCalibrate-
dRGBColorSpace)
If rgbColor <>Nil Then
Dim red as Integer = rgbColor.redComponent * 255.0
Dim green as Integer = rgbColor.greenComponent * 255.0
Dim blue as Integer = rgbColor.blueComponent * 255.0
Return RGB(red, green, blue)
Else
Return HighlightColor
End If
#else
return HighlightColor
#endif
End Function
```

Notes: As you see we convert color to Calibrated RGB for best results.
See also:

- 8.0.4 How to catch delete key? 1063
- 8.0.5 How to convert cmyk to rgb? 1064
- 8.0.6 How to delete a folder? 1065
- 8.0.7 How to detect if CPU if 64bit processor? 1066
- 8.0.8 How to query variant type string for a variant? 1067
- 8.0.9 How to refresh a htmlviewer on Windows? 1068

8.0.4 How to catch delete key?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following is the code in keydown event catches delete or backspace keys.
Example:

```
Function KeyDown(Key As String) As Boolean
if asc(key) = 8 or asc(key) = 127 then
MsgBox "Delete"
Return true
end if
End Function
```

See also:

- 8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1062

8.0.5 How to convert cmyk to rgb?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

The following is the code to convert cmyk values to an RGB color datatype.
It's just a basic estimate of the color values. If you are looking for completely color accurate solution, this is not it. It should work for most people. :)
Example:

```
Function CMYKToRGB(c as Integer, m as Integer, y as Integer, k as Integer) As color
// converts c,m,y,k values (0-100) to color data type RGB
// place this in a method. Supply C,M,Y,K values-
// it returns color datatype


dim color_RGB as color
dim r, g, b as Integer

r=255-round(2.55*(c+k))
if r<0 then
r=0
end if
g=255-round(2.55*(m+k))
if g<0 then
g=0
end if
b=255-round(2.55*(y+k))
if b<0 then
b=0
end if

color_RGB=RGB(r,g,b)

return color_RGB

End Function
```

Notes:

(from the rb mailinglist)
See also:

- 8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1062
- 8.0.4 How to catch delete key? 1063
- 8.0.6 How to delete a folder? 1065
- 8.0.7 How to detect if CPU if 64bit processor? 1066
- 8.0.8 How to query variant type string for a variant? 1067
- 8.0.9 How to refresh a htmlviewer on Windows? 1068

8.0.6 How to delete a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following is the code deletes a folder recursively.
Example:

```
Sub deletefolder(f as folderitem)
dim files(-1) as FolderItem

if f=nil then Return

// delete single file
if f.Directory=false then
f.Delete
Return
end if

// get a list of all items in that folder
dim i,c as Integer
c=F.Count
for i=1 to c
files.Append f.TrueItem(i)
next

// delete each item
for each fo as FolderItem in files
if fo=nil then
' ignore
elseif fo.Directory then
deletefolder fo
fo.delete
else ' file
```

```
fo.Delete
end if
next

f.Delete
End Sub
```

See also:

- 8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1062
- 8.0.4 How to catch delete key? 1063
- 8.0.5 How to convert cmyk to rgb? 1064
- 8.0.7 How to detect if CPU if 64bit processor? 1066
- 8.0.8 How to query variant type string for a variant? 1067
- 8.0.9 How to refresh a htmlviewer on Windows? 1068

8.0.7 How to detect if CPU if 64bit processor?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Via CPUID you can ask CPU:
Example:

```
dim c as new CPUIDMBS

if c.Flags(CPUIDMBS.kFeatureLM) then
MsgBox ”64-bit CPU”
else
MsgBox ”32-bit CPU”
end if
```

Notes: Should work on all intel compatible CPUs.
See also:

- 8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1062
- 8.0.4 How to catch delete key? 1063
- 8.0.5 How to convert cmyk to rgb? 1064
- 8.0.6 How to delete a folder? 1065
- 8.0.8 How to query variant type string for a variant? 1067
- 8.0.9 How to refresh a htmlviewer on Windows? 1068

8.0.8 How to query variant type string for a variant?

Plugin Version: 20.5, Platforms: macOS, Linux, Windows.

Answer: The following example function returns type string for variant.
Example:

```
Public Function VariantTypeString(v as Variant) as string
// Xojo’s VarType doesn’t know Unsigned integers
’Dim type As Integer = VarType(v)

// MBS VarType can detect unsigned integer
Dim type As Integer = GetVariantTypeMBS(v)

Dim IsArray As Boolean = BitwiseAnd(type, Variant.TypeArray) = Variant.TypeArray

// type without array
type = BitwiseAnd(type, Bitwise.OnesComplement(Variant.TypeArray))

// build a dictionary to map types on first call
Static TypeMap As Dictionary
If TypeMap = Nil Then
TypeMap = New Dictionary
TypeMap.Value(Variant.TypeBoolean) = ”Boolean”
TypeMap.Value(Variant.TypeCFStringRef) = ”CFStringRef”
TypeMap.Value(Variant.TypeColor) = ”Color”
TypeMap.Value(Variant.TypeCString) = ”CString”
TypeMap.Value(Variant.TypeCurrency) = ”Currency”
TypeMap.Value(Variant.TypeDate) = ”Date”
TypeMap.Value(Variant.TypeDateTime) = ”DateTime”
TypeMap.Value(Variant.TypeDouble) = ”Double”
TypeMap.Value(Variant.TypeInt32) = ”Int32”
TypeMap.Value(Variant.TypeInt64) = ”Int64”
TypeMap.Value(Variant.TypeInteger) = ”Integer”
TypeMap.Value(Variant.TypeNil) = ”Nil”
TypeMap.Value(Variant.TypeObject) = ”Object”
TypeMap.Value(Variant.TypeOSType) = ”OSType”
TypeMap.Value(Variant.TypePString) = ”PString”
TypeMap.Value(Variant.TypePtr) = ”Ptr”
TypeMap.Value(Variant.TypeSingle) = ”Single”
TypeMap.Value(Variant.TypeString) = ”String”
TypeMap.Value(Variant.TypeStructure) = ”Structure”
TypeMap.Value(Variant.TypeText) = ”Text”
TypeMap.Value(Variant.TypeWindowPtr) = ”WindowPtr”
TypeMap.Value(Variant.TypeWString) = ”WString”

// MBS extra types
TypeMap.Value(Variant.TypeInt32+100) = ”UInt32”
TypeMap.Value(Variant.TypeInt64+100) = ”UInt64”
```

```
End If

// lookup type

#if DebugBuild then
If Not TypeMap.HasKey(type) Then
Break // missing type
End If
#endif

If IsArray Then
Return "Array of " + TypeMap.Lookup(type,"?")
Else
Return TypeMap.Lookup(type,"?")
End If
End Function
```

See also:

- 8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1062
- 8.0.4 How to catch delete key? 1063
- 8.0.5 How to convert cmyk to rgb? 1064
- 8.0.6 How to delete a folder? 1065
- 8.0.7 How to detect if CPU if 64bit processor? 1066
- 8.0.9 How to refresh a htmlviewer on Windows? 1068

8.0.9 How to refresh a htmlviewer on Windows?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can ask the browser to reload the website with this code line:
Example:

```
call htmlViewer1.IERunJavaScriptMBS("javascript:document.location.reload()")
```

See also:

- 8.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 1062
- 8.0.4 How to catch delete key? 1063
- 8.0.5 How to convert cmyk to rgb? 1064

8.0.10 Is there an example for vector graphics in Xojo?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this example inside the paint event of a window:
Example:

```
dim v as Group2D
dim r as RectShape
dim s as StringShape

const pi=3.14

s=new StringShape
s.Text=”Hello World!”
s.TextFont=”Geneva”
s.TextSize=24
s.FillColor=rgb(0,0,255)
s.Italic=true
s.y=5
s.x=0

r=new RectShape

r.X=0
r.y=0
r.Height=100
r.Width=180
r.BorderColor=rgb(255,0,0)
r.FillColor=rgb(0,255,0)
r.BorderWidth=5
r.Border=50

v=new Group2d
v.Append r
v.Append s
v.Rotation=pi*-20.0/180.0
v.x=150
v.y=150

g.DrawObject v
```

8.0.11 Picture functions do not preserve resolution values?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, the picture functions return pictures with no/default resolution values.
Example:

```
dim l as Picture = LogoMBS(500)

l.HorizontalResolution = 300
l.VerticalResolution = 300

dim r as Picture = l.Rotate90MBS

MsgBox str(r.HorizontalResolution)+" x "+str(r.VerticalResolution)

r.HorizontalResolution = l.HorizontalResolution
r.VerticalResolution = l.VerticalResolution

MsgBox str(r.HorizontalResolution)+" x "+str(r.VerticalResolution)
```

Notes: So please fix them yourself after calling a function.

Maybe in the future this changes, but currently you can't really set this easily from plugin code.

8.0.12 A toolbox call needs a rect - how do I give it one?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Fill a memoryblock like this:
Example:

```
Dim MB As Memoryblock
MB = NewMemoryBlock(8)
MB.Short(0) = window1.Top
MB.Short(2) = window1.Left
MB.Short(4) = window1.Height+window1.Top // bottom
MB.Short(6) = window1.Width+window1.Left // right
```

8.0.13 API client not supported?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: If you get this exception message on SQLConnectionMBS.Connect, we may have a problem.
Notes: First case is that the given thing is not supported (e.g. MS SQL directly on Mac).
Second case is that the plugin compilation went wrong and the support for the database was not linked into the plugin. Like MySQL missing or MS SQL on Windows missing. In that case please contact us to fix the plugin.

8.0.14 Can I access Access Database with Java classes?

Plugin Version: all, Platform: Windows.

Answer: You can use ucanaccess to access databases created with Microsoft
Example:

```
dim options(-1) as string

// load all the jar files we have in a folder called java:

dim appFolder as FolderItem = GetFolderItem(””)

Dim count as Integer = appFolder.Parent.Child(”java”).Count
dim libjs() as string
For i as Integer = 1 to count
Dim f As FolderItem = appFolder.Parent.Child(”java”).item(i)
If f <>Nil and f.Exists Then
libjs.append f.NativePath+”;”
End If
Next

// now init virtual machine
dim librery as string = Join(libjs, ””)
dim vm as new JavaVMMBS(librery)

if vm.Handle = 0 then
MsgBox ”Failed to initialize virtual machine”
else
// now make a new database connection with ucanaccess
dim d as new JavaDatabaseMBS(vm,”net.ucanaccess.jdbc.UcanaccessDriver”)
Dim DbFile as FolderItem = appFolder.Parent.Child(”Database11.accdb”)
dim j as JavaConnectionMBS = d.getConnection(”jdbc:ucanaccess://”+DbFile.NativePath)

// select and show values
dim r as JavaResultSetMBS = j.MySelectSQL(”Select * From test”)
while r.NextRecord
MsgBox r.getString(”FirstName”) +” ”+ r.getString(”LastName”)
wend

end if
```

```
Exception e as JavaExceptionMBS
MsgBox e.message+" errorcode: "+str(e.ErrorNumber)
```

Notes: see website:
http://ucanaccess.sourceforge.net/site.html

8.0.15 Can I create PDF from Xojo Report using DynaPDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, we have a graphics class integration for DynaPDF.
Notes: Since MBS Plugin in version 19.2, we can integrate reports with Xojo.

8.0.16 Can I use AppleScripts in a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, but they run on the server, not on the client.
Example:

```
dim a as new AppleScriptMBS

// query my application name
a.Compile "tell application ""System Events"" to return name of current application"

// run
a.Execute

// show result
label1.text = a.Result

// shows something like "My Application.fcgi.debug"
```

Notes: This can be useful to control the server from remote, if and only if the your sever is running Mac OS X.

8.0.17 Can I use graphics class with DynaPDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Sorry, no. We can’t provide a graphics subclass from plugin.
Notes: The is a feature request to allow graphics subclasses:
Feedback case 11391: feedback://showreport?report_id=11391

8.0.18 Can I use sockets on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, but they run on the server, not on the client.
Notes: You can use HTTPSocket, SMTPSocket, POP3Socket, SMTPSecureSocket, SecurePOP3Socket, EasyTCPSocket, EasyUDPSocket, AutoDiscovery, our Bonjour classes or our CURL* classes. But all of them work on the server, not on the client.

This means if you search for a printer with Bonjour, you can find the printers in the local network on your server hosting site. Using SMTPSocket may be a good idea for sending emails from the server like notifications.

8.0.19 Can I use your ChartDirector plugin on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, our ChartDirector plugin works just fine on the Xojo Web Edition.
Example:

```
// The data for the pie chart
dim data(-1) as Double=array(55.0, 18.0, 25.0, 22.0, 18.0, 30.0, 35.0)

// The labels for the pie chart, Words are choosen random to check font!
dim labels(-1) as string=array(”Germany”,”Italy”,”France”,”Spain”,”UK”,”Poland”,”Russia”)

// The colors to use for the sectors
dim colors(-1) as Integer

colors.Append &h66aaee
colors.Append &heebb22
colors.Append &hbbbbbb
colors.Append &h8844ff

if TargetLinux then
CDBaseChartMBS.SetFontSearchPath ”/usr/share/fonts/truetype/msttcorefonts”
end if

// Create a PieChart object of size 360 x 300 pixels
dim c as new CDPieChartMBS(700, 600)
```

```
c.setBackground(c.linearGradientColor(0, 0, 0, c.getHeight(), &h0000cc, &h000044))
c.setRoundedFrame(&hffffff, 16)
dim tt as CDTextBoxMBS = c.addTitle("ChartDirector Demonstration", "timesbi.ttf", 18)
tt.setMargin(0, 0, 16, 0)
tt.setFontColor(&hFFFFFF)

// Set the center of the pie at (180, 140) and the radius to 100 pixels
c.setPieSize 350,300,150
// Set the sector colors
c.setColors(c.kDataColor, colors)

// Draw the pie in 3D with a pie thickness of 20 pixels
c.set3D(20)

dim t as CDTextBoxMBS = c.setLabelStyle("arialbd.ttf", 10, &h000000)
t.setBackground(CDPieChartMBS.kSameAsMainColor, CDPieChartMBS.kTransparent, CDPieChartMBS.softLighting(CDPieChartMBS.kRight, 0))
t.setRoundedCorners(8)

// Use local gradient shading for the sectors, with 5 pixels wide
// semi-transparent white (bbffffff) borders
c.setSectorStyle(CDPieChartMBS.kLocalGradientShading, &hbbffffff, 0)

// Set the pie data and the pie labels
c.setData data,labels
call c.setLabelStyle "arialbd.ttf",18

dim pic as picture = c.makeChartPicture
dim wp as new WebPicture(pic, Picture.FormatJPEG) // JPEG makes it smaller and faster

ImageView1.Picture=wp
```

Notes: Be aware that our plugin produces pictures for you, which you assign to ImageViews. Tranfserring those pictures takes time, so you can optimize that with using WebPicture class. There you can decide between different compressions to improve speed (use JPEG instead of PNG).

e.g. if you use ubuntu, you can install the ttf-mscorefonts-installer package and call this method with "/usr/share/fonts/truetype/msttcorefonts" as the path. No backslash on the end of a path, please.

8.0.20 Can I use your DynaPDF plugin on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, our DynaPDF plugin works just fine on the Xojo Web Edition.
Notes: PDF files are created on the server. You may want to offer a preview to the user which uses reduced resolution images to reduce the time to download the PDF.

See our Create PDF example for the Xojo Web Edition.

8.0.21 Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

8.0.22 Can you get an unique machine ID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: There is nothing like an unique machine ID.
Notes: 1:
You can use the MAC IDs of the network interfaces.
This can be changed by the user with software tools.
And the list of network interfaces changes if user reorder the interfaces.

2:
You can use the system folder creation date/time.
This may stay equal after cloning machines or after migration to
new PC.

3:
You can use the Mac Serialnumber.
Mac only and it can happen that a Mac does not have a serial
number.

4:
You can use the x86 CPU ID.
This is x86 CPU only and does not avoid running on the same CPU in
different PCs.

8.0.23 ChartDirector: Alignment Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Alignment Specification
Notes: In many ChartDirector objects, you may specify the alignment of the object's content relative to its boundary. For example, for a TextBox object, you may specify the text's alignment relative to the box boundary by using TextBox.setAlignment.

The ChartDirector API defines several constants for the alignment options.

ConstantValueDescription

BottomLeft	1	The leftmost point on the bottom line.
BottomCenter	2	The center point on the bottom line.
BottomRight	3	The rightmost point on the bottom line.
Left	4	The leftmost point on the middle horizontal line.
Center	5	The center point on the middle horizontal line.
Right	6	The rightmost point on the middle horizontal line.
TopLeft	7	The leftmost point on the top line.
TopCenter	8	The center point on the top line.
TopRight	9	The rightmost point on the top line.
Bottom	2	The center point on the bottom line. Same as BottomCenter.
Top	8	The center point on the top line. Same as TopCenter.
TopLeft2	10	An alternative top-left position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, TopLeft2 refers to refers to the left of the top side, while TopLeft refers to the top of the left side. The reverse applies for a horizontal axis.
TopRight2	11	An alternative top-right position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, TopRight2 refers to refers to the right of the top side, while TopRight refers to the top of the right side. The reverse applies for a horizontal axis.
BottomLeft2	12	An alternative bottom-left position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, BottomLeft2 refers to refers to the left of the bottom side, while BottomLeft refers to the bottom of the left side. The reverse applies for a horizontal axis.
BottomRight2	13	An alternative bottom-right position used in Axis.setTitlePos for axis title positioning only. For a vertical axis, BottomRight2 refers to refers to the right of the bottom side, while BottomRight refers to the bottom of the right side. The reverse applies for a horizontal axis.

8.0.24 ChartDirector: Color Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Color Specification
Notes: Many functions in the ChartDirector API accept colors as parameters. ChartDirector supports col-

ors specified in web and HTML compatible ARGB format, in which ARGB refers to the Alpha transparency, Red, Green and Blue components of the color.

In addition to ARGB colors, ChartDirector supports "dynamic" colors. A dynamic color is a color that changes depending on the position of the pixels. The "dynamic" colors that ChartDirector supports include "pattern colors", "metal colors", "gradient colors", "zone colors" and "dash line colors".

ChartDirector supports specifying colors indirectly using "palette colors". When a "palette color" is used, the color is specified as an index to a palette. The actual color is looked up from the palette.ARGB Color ARGB color consists of 4 components - alpha transparency, red, green and blue. The four components are encoded as a 32-bit number, with each component occupying 8 bits. In hexadecimal notation, it is AARRGGBB, where AA, RR, GG and BB are the alpha transparency, red, green and blue components.

Each component ranges from 00 - FF (0 - 255), representing its intensity. For example, pure red color is 00FF0000, pure green color is 0000FF00, and pure blue color is 000000FF. White color is 00FFFFFF, and black color is 00000000.

Most programming language requires you to put special prefix in front of hexadecimal characters. For C++, the prefix is "0x". For example, the syntax for the hexadecimal number 00FFFFFF is 0x00FFFFFF, or simply 0xFFFFFF.
For the alpha transparency component, a zero value means the color is not transparent all at. This is equivalent to traditional RGB colors. A non-zero alpha transparency means the the color is partially transparent. The larger the alpha transparency, the more transparent the color will be. If a partially transparent color is used to draw something, the underlying background can still be seen.

For example, 80FF0000 is a partially transparent red color, while 00FF0000 is a non-transparent red color.

Note that ChartDirector's ARGB color is web and HTML compatible. For example, red is FF0000, the same as in HTML. There are many resources on the web that provide tables in which you can click a color and it will show its HTML color code. These color codes can be used in ChartDirector.

If alpha transparency is FF (255), the color is totally transparent. That means the color is invisible. It does not matter what the RGB components are. So in ChartDirector, only one totally transparent color is used - FF000000. All other colors of the form FFnnnnnn are reserved to represent palette colors and dynamic colors, and should not be interpreted as the normal ARGB colors.

The totally transparent color FF000000 is often used in ChartDirector to disable drawing something. For example, if you want to disable drawing the border of a rectangle, you can set the border color to totally transparent.

For convenience, ChartDirector defines a constant called Transparent, which is equivalent to FF000000.Pattern Color

A pattern color is a dynamic color that changes according to a 2D periodic pattern. When it is used to fill an area, the area will look like being tiled with a wallpaper pattern.

Pattern colors are created using BaseChart.patternColor, BaseChart.patternColor2, DrawArea.patternColor and DrawArea.patternColor2. The patternColor method creates pattern colors using an array of colors as a bitmap. The patternColor2 method creates pattern colors by loading the patterns from image files.

These methods return a 32-bit integer acting as a handle to the pattern color. The handle can be used in any ChartDirector API that expects a color as its input.Metal Color
A metal color is a color of which the brightness varies smoothly across the chart surface as to make the surface looks shiny and metallic. ChartDirector supports using any color as the base color of the metal color. In particular, using yellow and grey as the base colors will result in metal colors that look gold and silver.

Metal colors are most often used as background colors of charts. They are created using CDBaseChartMBS.metalColor, CDBaseChartMBS.goldColor and CDBaseChartMBS.silverColor. The first method allows you to specify an arbitrary base color. The second and third methods use yellow and grey as the base colors, resulting in gold and silver metal colors.

These methods return a 32-bit integer acting as a handle to the gradient color. The handle can be used in any ChartDirector API that expects a color as its input.Gradient Color
A gradient color is a color that changes progressively across a direction.

Gradient colors are created using BaseChart.gradientColor, BaseChart.gradientColor2, DrawArea.gradientColor and DrawArea.gradientColor2. The gradientColor method creates a 2-point gradient color that changes from color A to color B. The gradientColor2 method creates a multi-point gradient colors that changes from color A to B to C ....

These methods return a 32-bit integer acting as a handle to the gradient color. The handle can be used in any ChartDirector API that expects a color as its input.

One common use of multi-point gradient colors is to define colors that have metallic look and feel. Please refer to DrawArea.gradientColor2 for details.Dash Line Colors
A dash line color is a color that switches on and off periodically. When used to draw a line, the line will appear as a dash line.

Dash line colors are created using BaseChart.dashLineColor and DrawArea.dashLineColor. They accept a line color and a dash pattern code as arguments, and return a 32-bit integer acting as a handle to the dash line color. The handle can be used in any ChartDirector API that expects a color as its input.Zone Colors
A zone color is for XY charts only. It is a color that automatically changes upon reaching a data threshold value along the x-axis or y-axis. Zone colors are created using Layer.xZoneColor, Layer.yZoneColor, XYChart.xZoneColor or XYChart.yZoneColor.Palette Colors
Palette colors are colors of the format FFFFnnnn, where the least significant 16 bits (nnnn) are the index to the palette. A palette is simply an array of colors. For a palette color, the actual color is obtained by

looking up the palette using the index. For example, the color FFFF0001 is the second color in the palette (first color is index 0).

The colors in the palette can be ARGB colors or ”dynamic” colors (pattern, gradient and dash line colors).

The first eight palette colors have special significance. The first three palette colors are the background color, default line color, and default text color of the chart. The 4th to 7th palette colors are reserved for future use. The 8th color is a special dynamic color that is equal to the data color of the ”current data set”.

The 9th color (index = 8) onwards are used for automatic data colors. For example, in a pie chart, if the sector colors are not specified, ChartDirector will automatically use the 9th color for the first sector, the 10th color for the second sector, and so on. Similarly, for a multi-line chart, if the line colors are not specified, ChartDirector will use the 9th color for the first line, the 10th color for the second line, and so on.

The ChartDirector API defines several constants to facilitate using palette colors.

ConstantValueDescription

Palette	FFFF0000	The starting point of the palette. The first palette color is (Palette + 0). The nth palette color is (Palette + n - 1).
BackgroundColor	FFFF0000	The background color.
LineColor	FFFF0001	The default line color.
TextColor	FFFF0002	The default text color.
[Reserved]	FFFF0003 - FFFF0006	These palette positions are reserved. Future versions of ChartDirector may use these palette positions for colors that have special significance.
SameAsMainColor	FFFF0007	A dynamic color that is equal to the data color of the current data set. This color is useful for objects that are associated with data sets. For example, in a pie chart, if the sector label background color is SameAsMainColor, its color will be the same as the corresponding sector color.
DataColor	FFFF0008	The starting point for the automatic data color allocation.

When a chart is created, it has a default palette. You may modify the palette using BaseChart.setColor, BaseChart.setColors, or BaseChart.setColors2.

The advantages of using palette colors are that you can change the color schemes of the chart in one place. ChartDirector comes with several built-in palettes represented by the following predefined constants.

ConstantDescription

8.0.25 ChartDirector: Font Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

defaultPalette	An array of colors representing the default palette. This palette is designed for drawing charts on white backgrounds (or lightly colored backgrounds).
whiteOnBlackPalette	An array of colors useful for drawing charts on black backgrounds (or darkly colored backgrounds).
transparentPalette	An array of colors useful drawing charts on white backgrounds (or lightly colored backgrounds). The data colors in this palette are all semi-transparent.

Answer: ChartDirector: Font Specification
Notes: Font Name
In ChartDirector, the font name is simply the file name that contains the font. For example, under the Windows platform, the "Arial" font is "arial.ttf", while the "Arial Bold" font is "arialbd.ttf".

NOTE: Mac OS X Specific Information
In Mac OS X, in addition to ".ttf", ChartDirector also supports Mac OS X font file formats, such as Font Suitcase files and Datafork files (.dfont). These files often contain multiple fonts. For example, the "GillSans.dfont" file contains 6 fonts.

So in addition to the file name, an index is needed to determine the font. The index is specified by appending a " | " character to the font name, followed by the index number. For example, the third font in "GillSans.dfont" is denoted as "GillSans.dfont | 2". (Note: The first font starts at 0.) If no index number is provided, the first font is assumed.

ChartDirector also supports using Mac OS X Font Manager names. For example, one may use "Gill Sans Light Italic" instead of using "GillSans.dfont | 1" as the font name. However, the Mac OS X Font Manager is active only if someone has logged into the Mac GUI console, so this method is only recommended for developing applications that run on the GUI console.

The sample programs that come with ChartDirector are designed to run on all operating systems, so they use generic font file names (eg. "arial.ttf") instead of Mac OS X specific names. To allow them to run on Mac OS X, ChartDirector on Mac OS X has a built-in table to map common font file names to Mac OS X font names:

"arial.ttf", "arialbd.ttf", "ariali.ttf" and "arialbi.ttf" are mapped to "Arial | 0" (Arial), "Arial | 1" (Arial Bold), "Arial | 2" (Arial Italic) and "Arial | 3" (Arial Bold Italic)

"times.ttf", "timesbd.ttf", "timesi.ttf" and "timesbi.ttf" are mapped to "Times New Roman | 0" (Times New Roman), "Times New Roman | 1" (Times New Roman Bold), "Times New Roman | 2" (Times New Roman Italic) and "Times New Roman | 3" (Times New Roman Bold Italic)

"cour.ttf", "courbd.ttf", "couri.ttf" and "courbi.ttf" are mapped to "Courier New | 0" (Courier New), "Courier New | 1" (Courier New Bold), "Courier New | 2" (Courier New Italic) and "Courier New | 3" (Courier New Bold Italic)

Font Location
ChartDirector on Windows does not come with any font files. It relies on the operating system's font files in the " [windows] \Fonts" directory. To see what fonts are installed in your operating system and their file names, use the File Explorer to view that directory.

ChartDirector on Windows will also search for the font files in the "fonts" subdirectory (if it exists) under the directory where the ChartDirector DLL "chartdir.dll" is installed. This is useful for private fonts. Also, for some especially secure web servers, the web anonymous user may not have access to the " [windows] \Fonts" directory. In this case, you may copy the font files to the above subdirectory.

ChartDirector on Mac OS X relies on operating system font files in "/Library/Fonts" and "/System/Library/Fonts".

ChartDirector on Linux, FreeBSD and Solaris assume the fonts files are in the "fonts" subdirectory under the directory where the ChartDirector shared object "libchartdir.so" is installed. ChartDirector on Linux, FreeBSD and Solaris come with a number of font files in the "fonts" subdirectory.

To keep the download size small, ChartDirector on Linux, FreeBSD and Solaris only come with some commonly used fonts. You may download additional fonts from the Internet. In particular, the Microsoft fonts at
http://sourceforge.net/project/showfiles.php?group_id=34153&release_id=105355
is highly recommended. Please refer to
http://www.microsoft.com/typography/faq/faq8.htm
on how you could use the fonts legally in your system.

ChartDirector supports True Type fonts (.ttf), Type 1 fonts (.pfa and .pfb) and Windows bitmap fonts (.fon). On Mac OS X, ChartDirector also supports Font Suitcase and Datafork (.dfont) files. On Linux, FreeBSD and Solaris, ChartDirector also supports Portable Compiled Fonts (.pcf fonts).

If you want ChartDirector to search other directories for the font files, you may list the directories in an environment variable called "FONTPATH".

If you specify an absolute path name for the font file, ChartDirector will use the absolute path name and will not search other directories.Artificial Boldening and Italicizing
Whereas most popular font comes with different styles for "normal", "bold", "italic" and "bold italic", some fonts only come with one style (the normal style). For example, the Monotype Corsiva font that comes with MS Office only has the normal style (mtcorsva.ttf). For these cases, you may append the "Bold" and/or "Italic" words after the font file name (separated with a space) to ask ChartDirector to artificially bolden and/or italicize the font. For example, you may specify the font name as "mtcorsva.ttf Bold".Font List
Instead of specifying a single font file as the font name, you may specify a list of font files as the font name, separated by semi-colons. This is useful when using international characters that are only available in some fonts.

For example, if you would like to use the Arial font (”arial.ttf”) for western characters, and the MingLiu font ”mingliu.ttc” for Chinese characters (since the Arial font does not have Chinese characters), you may specify the font name as ”arial.ttf;mingliu.ttc”. In this case, ChartDirector will try the Arial font first. If it cannot find a certain character there, it will try the MingLiu font.Indirect Font Names
ChartDirector supports several special keywords for specifying the font name indirectly. When these keywords are used as font names, ChartDirector will look up the actual font names from a font table. The keywords are as follows:

KeywordsDescription

”normal”	This default normal font, which is the first font in the font table. This is initially mapped to ”arial.ttf” (Arial).
”bold”	The default bold font, which is the second font in the font table. This is initially mapped to ”arialbd.ttf” (Arial Bold).
”italic”	The default italic font, which is the third font in the font table. This is initially mapped to ”ariali.ttf” (Arial Italic).
”boldItalic”	The default bold-italic font, which is the fourth font in the font table. This is initially mapped to ”arialbi.ttf” (Arial Bold Italic).
”fontN”	The (N + 1)th font in the font table (the first font is ”font0”).

The font table can be modified using BaseChart.setFontTable or DrawArea.setFontTable.

The advantage of using indirect font names is that you can change the fonts fonts in your charts in one place.Font Index
Most font files contain one font. However, it is possible a font file contains multiple fonts (that is, a font collection). For example, in True Type fonts, font files with extension ”.ttc” may represent a font collection.

If a font file contains multiple font, the font index can be used to specify which font to use. By default, the font index is 0, which means the first font in the font file will be used.Font Size
The font size decides how big a font will appear in the image. The font size is expressed in a font unit called points. This is the same unit used in common word processors.

Instead of specifying font size, some ChartDirector API (eg. TextBox.setFontSize) allow you to specify font height and font width separately. You may use different point sizes for font height and font width to create special effects.Font Color
This is the color to draw the font. (See Color Specification on how colors are represented in ChartDirector.)Font Angle
This is the angle in degrees by which the font should be rotated anti-clockwise.Vertical Layout
By default, text are laid out horizontally, with characters being drawn from left to right.

ChartDirector also supports vertical layout, with characters being drawn from top to bottom. For example, you may use BaseChart.addText to add text that are laid out vertically. Vertical layout is common for oriental languages such as Chinese, Japanese and Korean.

8.0.26 ChartDirector: Mark Up Language

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Mark Up Language
Notes: ChartDirector Mark Up Language (CDML) is a language for including formatting information in text strings by marking up the text with tags.

CDML allows a single text string to be rendered using multiple fonts, with different colors, and even embed images in the text.Font Styles
You can change the style of the text by using CDML tags. For example, the line:

<*font=timesi.ttf,size=16,color=FF0000>Hello <*font=arial.ttf,size=12,color=8000*>world!
will result in the following text rendered:

In general, all tags in CDML are enclosed by <* and *>. Attributes within the tags determine the styles of the text following the tags within the same block.

If you want to include <* in text without being interpreted as CDML tags, use «* as the escape sequence.

The following table describes the supported font style attributes in CDML. See Font Specification for details on various font attributes.

AttributeDescription

Set the following text to be in superscript style. This attribute does not need to have a value. (You may use ”super” as the attribute instead of ”super=1”.)

Note that unlike HTML tags, no double or single quotes are used in the tags. It is because CDML tags are often embedded as string literals in source code. The double or single quotes, if used, will conflict with the string literal quotes in the source code. Therefore in CDML, no quotes are necessary and they must not be used.

Also, unlike HTML tags, CDML uses the comma character as the delimiter between attributes. It is because certain attributes may contain embed spaces (such as the font file name). So space is not used as the delimiter and the comma character is used instead.

Note the font attribute above starts a new style section, while other attributes just modify the current style

font	Starts a new style section, and sets the font name. You may use this attribute without a value (that is, use ”font” instead of ”font=arial.ttf”) to create a new style section without modifying the font name.
size	The font size.
width	The font width. This attribute is used to set the font width and height to different values. If the width and height are the same, use the size attribute.
height	The font height. This attribute is used to set the font width and height to different values. If the width and height are the same, use the size attribute.
color	The text color in hex format.
bgColor	The background color of the text in hex format.
underline	The line width of the line used to underline the following characters. Set to 0 to disable underline.
sub	Set the following text to be in subscript style. This attribute does not need to have a value. (You may use ”sub” as the attribute instead of ”sub=1”.)
super	Set the following text to be in superscript style.

xoffset	Draw the following the text by shifting the text horizontally from the original position by the specified offset in pixels.
yoffset	Draw the following the text by shifting the text vertically from the original position by the specified offset in pixels.
advance	Move the cursor forward (to the right) by the number of pixels as specified by the value this attribute.
advanceTo	Move the cursor forward (to the right) to the position as specified by the value this attribute. The position is specified as the number of pixels to the right of the left border of the block. If the cursor has already passed through the specified position, the cursor is not moved.

section. You may use <*/font*>to terminate a style section, which will restore the font styles to the state before the style section.Blocks and Lines
In CDML, a text string may contain multiple blocks. A block may contain multiple lines of text by separating them with new line characters (”\n”) or with <*br*>. The latter is useful for programming languages that cannot represent new line characters easily.

For example, the line:

<*size=15*><*block*><*color=FF*>BLOCK<*br*>ONE<*/*>and <*block*><*color=FF00*>BLOCK<*br*>TWO
will result in the following text rendered:

The above example contains a line of text. The line contains two blocks with the characters ” and ” in between. Each block in turn contains two lines. The blocks are defined using <*block*>as the start tag and

<*/*>as the end tag.

When a block ends, font styles will be restored to the state before entering the block.Embedding Images
CDML supports embedding images in text using the following syntax:

<*img=my_image_file.png*>
where my_image_file.png is the path name of the image file.

For example, the line:

<*size=20*>A <*img=sun.png*>day
will result in the following text rendered:

ChartDirector will automatically detect the image file format using the file extension, which must either png, jpg, jpeg, gif, wbmp or wmp (case insensitive).

Please refer to BaseChart.setSearchPath or DrawArea.setSearchPath on the directory that ChartDirector will search for the file.

The <*img*>tag may optionally contain width and height attributes to specify its pixel width and height. In this case, ChartDirector will stretch or compress the image if necessary to the required width and height.Blocks Attributes
CDML supports nesting blocks, that is, a block can contain other sub-blocks. Attributes are supported in the <*block*>tag to control the alignment and orientation of the sub-blocks. The <*img=my_image_file.png*>is treated as a block for layout purposes.

For example, the line:

<*block,valign=absmiddle*><*img=molecule.png*><*block*>Hydrazino\nMolecule<*/*><*/*>
will result in the following text rendered:

The the above starts <*block,valign=absmiddle*>which specifies its content should align with each others in the vertical direction using the absolute middle alignment. The block contains an image, followed by a space characters, and then another block which has two lines of text.

The following table describes the supported attributes inside <*block*>tag:

AttributeDescription

The value baseline means the baseline of sub-blocks should align with the baseline of the block. The baseline width The width of the block in pixels. By default, the width is automatically determined to be the width necessary for the contents of the block. If the width attribute is specified, it will be used as the width of the block. If the width is insufficient for the contents, the contents will be wrapped into multiple lines.

height The height of the block in pixels. By default, the height is automatically determined to be the height necessary for the contents of the block. If the height attribute is specified, it will be used as the height of the block.

maxwidth The maximum width of the block in pixels. If the content is wider than maximum width, it will be wrapped into multiple lines.

truncate The maximum number of lines of the block. If the content requires more than the maximum number of lines, it will be truncated. In particular, if truncate is 1, the content will be truncated if it exceeds the maximum width (as specified by maxwidth or width) without wrapping. The last few characters at the truncation point will be replaced with ”...”.

linespacing The spacing between lines as a ratio to the default line spacing. For example, a line spacing of 2 means the line spacing is two times the default line spacing. The default line spacing is the line spacing as specified in the font used.

bgColor The background color of the block in hex format.

valign The vertical alignment of sub-blocks. This is for blocks that contain sub-blocks. Supported values are baseline, top, bottom, middle and absmiddle.

is the underline position of text. This is normal method of aligning text, and is the default in CDML. For images or blocks that are rotated, the baseline is the same as the bottom.

The value top means the top line of sub-blocks should align with the top line of the block.

The value bottom means the bottom line of sub-blocks should align with the bottom line of the block.

The value middle means the middle line of sub-blocks should align with the the middle line of the block. The middle line is the middle position between the top line and the baseline.

The value absmiddle means the absolute middle line of sub-blocks should align with the absolute middle line of the block. The absolute middle line is the middle position between the top line and the bottom line.

halign The horizontal alignment of lines. This is for blocks that contain multiple lines. Supported values are left, center and right.

The value left means the left border of each line should align with the left border of the block. This is the default.

The value center means the horizontal center of each line should align with the horizontal center of the block.

The value right means the right border of each line should align with the right border of the block.

angle	Rotate the content of the block by an angle. The angle is specified in degrees in counter-clockwise direction.

8.0.27 ChartDirector: Parameter Substitution and Formatting

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Parameter Substitution and Formatting
Notes: ChartDirector charts often contain a lot of text strings. For example, sector labels in pie charts, axis labels for x and y axes, data labels for the data points, HTML image maps, etc, are all text strings.

ChartDirector uses parameter substitution to allow you to configure precisely the information contained in the text and their format.
Format Strings
In parameter substitution, format strings are used to specify the entities to be include into labels and how to format numbers and dates.

For example, when drawing a pie chart with side label layout, the default sector label format string is:

" { label } ({ percent } %)"

When the sector label is actually drawn, ChartDirector will replace " { label } " with the sector name, and " { percent } " with the sector percentage. So the above label format will result is a sector label similar to "ABC (34.56%)".

You may change the sector label format by changing the format string. For example, you may change it to:

" { label } : US$ { value | 2 } K ({ percent } %)"

The sector label will then become something like "ABC: US$ 123.00 (34.56%)".

In general, in ChartDirector parameter substitution, parameters enclosed by curly brackets will be substituted with their actual values when creating the texts.

For parameters that are numbers or dates/times, ChartDirector supports a special syntax in parameter substitution to allow formatting for these values. Please refer to the Number Formatting and Date/Time Formatting sections below for details.

Parameter Expressions

ChartDirector supports numeric expressions in format strings. They are denoted by enclosing the expression with curly brackets and using ”=” as the first character. For example:

”USD { value } (Euro { = { value } *0.9 })”

In the above, ” { value } ” will be substituted with the actual value of the sector. The expression ” { = { value } *0.9 } ” will be substituted with the actual value of the sector multiplied by 0.9.

ChartDirector parameter expressions support operators ”+”, ”-”, ”*”, ”/”, ”%” (modulo) and ”^” (exponentiation). Operators ”*”, ”/”, ”%”, ”^” is computed first, followed by ”+” and ”-”. Operators of the same precedence are computed from left to right). Parenthesis ”(” and ”)” can be used to change the computation order.

Parameters for Pie Charts

The following table describes the parameters available for pie charts.

Parameter	Description
sector	The sector number. The first sector is 0, while the nth sector is (n-1).
dataSet	Same as { sector } . See above.
label	The text label of the sector.
dataSetName	Same as { label } . See above.
value	The data value of the sector.
percent	The percentage value of the sector.
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using BaseChart.addExtraField or BaseChart.addExtraField2.

Parameters for All XY Chart Layers

The followings are parameters that are apply to all XY Chart layers in general. Some layer types may have additional parameters (see below).

Note that certain parameters are inapplicable in some context. For example, when specifying the aggregate label of a stacked bar chart, the { dataSetName } parameter is inapplicable. It is because a stacked bar is composed of multiple data sets. It does not belong to any particular data set and hence does not have a data set name.

{ fieldN } means the extra field is indexed by the data point number. The Pth data point corresponds to the Pth element of the extra field.

Additional Parameters for Line Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Trend Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Box-Whisker Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for HLOC and CandleStick Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Additional Parameters for Vector Layers

The followings are parameters that are in additional to the parameters for all XY Chart layers.

Parameters for All Polar Layers

The followings are parameters that are apply to all Polar Chart layers in general. Some layer types may have additional parameters (see below).

{ fieldN } means the extra field is indexed by the data point number. The Pth data point corresponds to the Pth element of the extra field.

Additional Parameters for PolarVector Layers

The followings are parameters that are in additional to the parameters for all Polar Chart layers.

Parameters for Axis

The following table describes the parameters available for pie charts.

Number Formatting

For parameters that are numbers, ChartDirector supports a number of formatting options in parameter substitution.

For example, if you want a numeric field { value } to have a precision of two digits to the right of the decimal point, use ’,’ (comma) as the thousand separator, and use ’.’ (dot) as the decimal point, and you may use { value | 2,. } . The number 123456.789 will then be displayed as 123,456.79.

For numbers, the formatting options are specified using the following syntax:

{ [param] | [a] [b] [c] [d] }

where:

If this field starts with ”E” or ”e”, followed by a number, it means formatting the value using scientific notation with the specified number of decimal places. If the ”E” or ”e” is not followed by a number, 3 is assumed.

For example, { value | E4 } will format the value 10.3 to 1.0300E+1, and { value | e4 } will format the same value to 1.0300e+1.

If this field starts with ”G” or ”g”, followed by a number, it means formatting the value using the scientific notation only if the value is large and requires more than the specified number of digits, or the value is less than 0.001. If scientific notation is used, the number following ”G” or ”g” also specifies the number of significant digits to use. If the ”G” or ”g” is not followed by a number, 4 is assumed.

For example, consider the format string { value | G4 } . The value 10 will be formatted to 10. The value 100000 will be formatted to 1.000E+5. Similarly, for { value | g4 } , the value 10 will be formatted to 10, while the value 100000 will be formatted to 1.000e+5.

If you skip this argument, ChartDirector will display the exact value using at most 6 decimal places.

You may skip [b] [c] [d] . In this case, the default will be used.

Date/Time Formatting
For parameters that are dates/times, the formatting options can be specified using the following syntax:

{ [param] | [datetime_format_string] }

where [datetime_format_string] must start with an english character (A-Z or a-z) that is not ”G”, ”g”, ”E” or ”e”, and may contain any characters except ’ } ’. (If it starts with ”G”, ”g”, ”E” or ”e”, it will be considered as a number format string.)

Certain characters are substituted according to the following table. Characters that are not substituted will be copied to the output.

For example, a parameter substitution format of { value | mm-dd-yyyy } will display a date as something similar to 09-15-2002. A format of { value | dd/mm/yy hh:nn:ss a } will display a date as something similar to 15/09/02 03:04:05 pm.

If you want to include characters in the format string without substitution, you may enclose the characters in single or double quotes.

For example, the format { value | mmm '<*color=dd0000*>'yyyy } will display a date as something like Jan <*color=dd0000*>2005 (the <*color=dd0000*>is a CDML tag to specify red text color). Note that the <*color=dd0000*>tag is copied directly without substitution, even it contains "dd" which normally will be substituted with the day of month.
Escaping URL/HTML/CDML characters
Parameter substitution is often used to create HTML image maps. In HTML, some characters has special meanings and cannot be used reliably. For example, the '>' is used to represent the end of an HTML tag.

Furthermore, if the field happens to be used as an URL, characters such as '?', '&' and '+' also have special meanings.

By default, ChartDirector will escape template fields used in URL and query parameters when generating image maps. It will modify URL special characters to the URL escape format "%XX" (eg. "?" will become "%3F"). After that, it will modify HTML special characters to the HTML escape format "&amps;#nn;" (eg. ">" will become "&amps;#62;".). Similarly, it will escape other attributes in the image map using HTML escape format (but not URL escape format).

In addition to escaping HTML and URL special characters, ChartDirector will also remove CDML fields in creating image maps. It is because CDML is only interpreted in ChartDirector, should not be useful outside of ChartDirector (such as in browser tool tips).

In some cases, you may not want ChartDirector to escape the special characters. For example, if the parameters have already been escaped before passing to ChartDirector, you may want to disable ChartDirector from escaping them again.

ChartDirector supports the following special fields to control the escape methods - " { escape_url } ", " { noescape_url } ", " { escape_html } ", " { noescape_html } ", " { escape_cdml } " and { noescape_cdml } ". These fields enable/disable the escape methods used in the template fields that follow them.

8.0.28 ChartDirector: Shape Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

Answer: ChartDirector: Shape Specification
Notes: Several ChartDirector API accept shape specification as arguments. For example, BarLayer.setBarShape and BarLayer.setBarShape2 can be used to specify shapes of bars in bar charts, while DataSet.setDataSymbol, DataSet.setDataSymbol4, PolarLayer.setDataSymbol and PolarLayer.setDataSymbol4 can be used to specify shapes for data symbols.

Note that in addition to shapes, in many cases ChartDirector also accepts images or custom draw objects for data representation. For example, see DataSet.setDataSymbol2, DataSet.setDataSymbol3, PolarLayer.setDataSymbol2 and PolarLayer.setDataSymbol3.

Built-In Shapes

Built-in shapes are specified as integers. The integers can be explicit constants, or can be generated by a ChartDirector method for parameterized shapes. For example, a circle is represented by an explicit constant CircleShape (=7). On the other hand, the number representing a polygon depends on the number of sides the polygon has, so it is generated by using the PolygonShape method, passing in the number of sides as argument.

The following table illustrates the various ChartDirector shapes:

Custom Shapes

In ChartDirector, custom shapes are specified as an array of integers x0, y0, x1, y1, x2, y2 ... representing the coordinates of the vertices of the custom polygonal shape.

The polygon should be defined with a bounding square of 1000 x 1000 units, in which the x-axis is from -500 to 500 going from left to right, and the y-axis is from 0 to 1000 going from bottom to top.

ChartDirector will automatically scale the polygon so that 1000 units will become to the pixel size as requested by the various ChartDirector API.

As an example, the shape of the standard diamond shape in ChartDirector is represented as an array with 8 numbers:

0, 0, 500, 500, 0, 1000, -500, 500

8.0.29 Copy styled text?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: How to quickly copy styled text from one textarea to another?
Example:

```
#if TargetWin32 then
TextArea1.WinRTFDataMBS = TextArea2.WinRTFDataMBS
#elseif TargetMacOS then
TextArea1.NSTextViewMBS.textStorage.setAttributedString TextArea2.NSTextViewMBS.textStorage
#else
TextArea1.StyledText = TextArea2.StyledText
#endif
```

Notes: The code above uses special plugin functions on Mac and Windows and falls back to framework for Linux.

8.0.30 Do you have code to validate a credit card number?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can check the checksum to tell if a credit card number is not valid.
Example:

```
Dim strNumber As String
Dim nLength as Integer
Dim nValue as Integer
Dim nChecksum as Integer
Dim nIndex as Integer

strNumber = EditField1.Text
nLength = Len(strNumber)
nChecksum = 0

For nIndex = 0 To nLength - 2
nValue = Val(Mid(strNumber, nLength - (nIndex + 1), 1)) * (2 - (nIndex Mod 2))
If nValue <10 Then
nChecksum = nChecksum + nValue
Else
nChecksum = nChecksum + (nValue - 9)
End If
Next

If Val(Mid(strNumber, Len(strNumber), 1)) = (10 - (nChecksum Mod 10)) Mod 10 Then
MsgBox("The credit card number looks valid")
Else
MsgBox("The credit card number is invalid")
End IF
```

Notes: Here's some code that will validate the checksum for a credit card. It works for Visa, MasterCard, American Express and Discover. Not sure about others, but I imagine they use the same basic algorithm. Of course, this doesn't actually mean that the credit card is valid, it's only useful for helping the user catch typos.

The above code doesn't have any error checking and it expects that the credit card number will be entered without spaces, dashes or any other non-numeric characters. Addressing those issues will be an exercise left to the reader. :)

(From Mike Stefanik)

8.0.31 Do you have plugins for X-Rite EyeOne, eXact or i1Pro?

Plugin Version: all.

Answer: Our EyeOne plugin is available on request for licensees of the X-Rite SDKs.
Notes: Please first go to X-Rite and get a SDK license.
Than we can talk about the plugin.

8.0.32 Does SQL Plugin handle stored procedures with multiple result sets?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Yes, the plugin can work with multiple recordsets.
Notes: You need to use SQLCommandMBS class. When you get back results, you use FetchNext to walk over all records in the first result set. Than you simply start again with FetchNext to get the second record set.
Even the RecordSet functions should work, just use them twice to get all records from both record sets.

8.0.33 Does the plugin home home?

Plugin Version: all, Platform: macOS.

Answer: Yes, we like to know who is using the plugin, so the plugin may contact our server.
Example:

none.

Notes: Please note that this does not affect your users as the plugin will only do this in the IDE and the relevant plugin part is never included in your applications.

The plugin if used for some hours, does contact our server to provide statistical data about Xojo version and OS versions. This way we know what versions are used. We can return the version number of the current plugin which may be visible in future versions somehow. And we transmit partial licenses data so we can track use of illegal license keys.

If you do not like to have this, you can block Xojo IDE from contacting our website via your Firewall.
Blocking the transfer will not disable the plugin or change the features.
Or contact us for a plugin version which explicitly does not contain this feature.

8.0.34 folderitem.absolutepath is limited to 255 chars. How can I get longer ones?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Paths on a Mac are not unique, so use them only to display them to the user.
Example:

```
Function AbsolutePath(f as FolderItem) As String
Dim s as string
Dim nf as FolderItem
nf = f
s = ""
while nf<>nil
s = nf.name + ":" + s
nf = nf.parent
wend
Return s
End Function
```

8.0.35 Has anyone played round with using CoreImage to do things like add dissolve transitions say when changing from one tab to another within a window?

Platform: macOS.

Answer: This code implements animations for a tabpanel change:
Example:

```
// in a tabpanel.change event:

dim r as CGSTransitionRequestMBS
dim co as new CGSConnectionMBS
dim cw as CGSWindowMBS
dim ct as CGSTransitionMBS
static OldTab as Integer

cw=co.CGSWindow(window1)
If cw = Nil Then
return // 10.3...
End If
r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSFlip
r.HasBackGround=false
r.HasBackColor=false
r.Win=cw
```

```
// watch the value of the clicked tab versus the last tab
if tabpanel1.Value=0 or tabpanel1.Value <OldTab then
r.TransitionOption=r.CGSLeft
ct=co.NewTransition(r)
if ct<>Nil then
Refresh
ct.Invoke(1)
ct.Wait(1)
ct.Release
else
MsgBox "Error creating the transition."
end if
else
r.TransitionOption=r.CGSRight
ct=co.NewTransition(r)
if ct<>Nil then
Refresh
ct.Invoke(1)
ct.Wait(1)
ct.Release
else
MsgBox "Error creating the transition."
end if
end if
// Keep track of the last tab clicked
OldTab = tabpanel1.Value
```

Notes: See CGS* classes for more details.

8.0.36 How about Plugin support for older OS X?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We support in general Mac OS X 10.5 and newer.
Notes: All the 64-bit plugins on Mac require OS X 10.7.
Intel 32-bit plugins on Mac require OS X 10.5 or newer.

Currently the ChartDirector 6, GraphicsMagick and GameKit plugins requires Mac OS X 10.6.
Also for SQL Plugin the built in SQLite library requires 10.6.

8.0.37 How can I detect whether an Intel CPU is a 64bit CPU?

Plugin Version: all.

Answer: Look on the CPU family returned by sysctl:
Example:

```
Function is64bit() As Boolean

#if TargetLittleEndian

dim m as MemoryBlock = NewMemoryBlock(8)
dim family as Integer
dim s as string

m=SystemControlNameToMIBMBS("hw.cpufamily")
m=SystemControlMBS(m)

if m<>nil then
m.LittleEndian=True
family=m.Long(0)

const CPUFAMILY_INTEL_6_14 = &h73d67300 //* "Intel Core Solo" and "Intel Core Duo" (32-bit Pentium-M with SSE3) */
const CPUFAMILY_INTEL_6_15 = &h426f69ef //* "Intel Core 2 Duo" */
const CPUFAMILY_INTEL_6_23 = &h78ea4fbc //* Penryn */
const CPUFAMILY_INTEL_6_26 = &h6b5a4cd2 //* Nehalem */

Select case family
case CPUFAMILY_INTEL_6_14
Return false
case CPUFAMILY_INTEL_6_15
Return true
case CPUFAMILY_INTEL_6_23
Return true
case CPUFAMILY_INTEL_6_26
Return true
// newer CPUs may be missing here
end Select

end if

#endif

Return false

Exception
Return false
End Function
```

Notes: This code is written for Mac OS X where you only have a limited number of possible CPUs.

8.0.38 How can I disable the close box of a window on Windows?

Plugin Version: all, Platform: Windows.

Answer: The following code will remove the close item from the system menu of the window.
Example:

```
#if TargetWin32 then
Declare Function GetSystemMenu Lib "user32" (hwnd as Integer, bRevert as Integer) as Integer
Declare Function RemoveMenu Lib "user32" (hMenu as Integer, nPosition as Integer, wFlags as Integer) as
Integer
Dim hSysMenu as Integer
hSysMenu = GetSystemMenu(me.WinHWND, 0)
RemoveMenu hSysMenu, &HF060, &H0
#endif
```

Notes: The window may not be updated directly.

8.0.39 How can I get all the environment variables from Windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
#if targetWin32
declare function GetEnvironmentStrings Lib "kernel32" () as ptr
dim m as memoryBlock
dim n as Integer

m=GetEnvironmentStrings()

n=0
do
msgBox m.cstring(n)
while m.byte(n)<>0
n=n+1
wend
n=n+1
```

```
loop until m.byte(n)=0
#endif
```

Notes: The MBS Plugin has an EnvironmentMBS class for this.

8.0.40 How can i get similar behavior to Roxio Toast or iTunes where clicking a 'burn' button allows the next inserted blank CD-R to bypass the Finder and be accepted by my application?

Plugin Version: all, Platform: macOS.

Answer: You need to get a media reservation.
Example:

```
dim d as DRDeviceMBS // get a device
d.AcquireMediaReservation
```

Notes: Use the plugin function AcquireMediaReservation and later release it using ReleaseMediaReservation.
See plugin examples on how to use it and check Apples DiscRecording framework documentation for more details.

8.0.41 How can I get text from a PDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Crossplatform you can use DynaPDF Pro.
Notes: On Mac OS X you can also use PDFKit for the same job.
While DynaPDF Pro gives you each bit of text with rotation, font information and encoding details, PDFKit gives you only the text string for a PDF page.

8.0.42 How can I get text from a Word Document?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: to get the text string from a doc file, use the NSAttributedStringMBS class.
Notes: The NSAttributedStringMBS class is Mac OS X only and we have currently no solution for Windows or Linux.

Use the NSAttributedStringMBS.initWithDocFormat(data as string) as boolean method.

8.0.43 How can I get the item string for a given file creator?

Plugin Version: all.

Answer: Try this function:
Example:

```
Sub pullNativeDocs(aCREA As string)
Dim result as Integer
Dim m, k as memoryBlock
Dim f as folderItem
Dim newType as string
Dim anIcon As picture
Dim ofs as Integer

Declare Function GetFileTypesThatAppCanNativelyOpen Lib "Carbon" (appVRefNumHint as Short, appSignature as OSType, nativeTypes as Ptr) as Short Inline68K("701CABFC")
Declare Function GetDocumentKindString Lib "Carbon" (docVRefNum as Short, docType as OSType, docCreator as OSType, kindString as ptr) as Short Inline68K("7016ABFC")

listBox1.deleteAllRows

m = newMemoryBlock(1024)
result = GetFileTypesThatAppCanNativelyOpen(Volume(0).MacVRefNum, aCREA, m)
if result <>0 then
listBox1.addRow "<Not found.>"
return
end if

do
if m.byte(ofs*4) = 0 then
exit
else
newType = m.OSTypeMBS(ofs*4)
listBox1.addRow newType
k = newMemoryBlock(64)
result = GetDocumentKindString(Volume(0).MacVRefNum, newType, aCREA, k)
if result = 0 then
listBox1.cell(ofs,1) = k.pString(0)
ofs = ofs + 1
else
listBox1.cell(ofs,1) = "(unknown)"
end if

end if
```

```
loop

End Sub
```

Notes: Change ”Translation” to ”CarbonLib” for Mac OS X.

8.0.44 How can I launch an app using it’s creator code?

Plugin Version: all, Platform: macOS.

Answer: Send an AppleEvent ”odoc” with the creator code to the Finder (”MACS”):
Example:

```
Function LaunchByCreator(C As String) As Boolean
Dim A As AppleEvent
A = NewAppleEvent(”aevt”,”odoc”,”MACS”)
A.ObjectSpecifierParam(”—-”) = GetUniqueIDObjectDescriptor(”appf”,nil,C)
return A.Send
End Function
```

8.0.45 How can I learn what shared libraries are required by a plugin on Linux?

Plugin Version: all, Platform: macOS.

Answer: Please use the ldd command in the terminal.
Notes: You build an app on any platform, but for Linux.
For the resulting .so files in the libs folder, you can run the ldd command with the library path as parameter.
It shows you references lib files and you can make sure you have those installed.

This is a sample run of our graphicsmagick plugin:

```
cs@Ubuntu32:
textasciitilde /MeinProgramm/MeinProgramm Libs$ ldd libMBSGraphicsMagickPlugin17744.so
linux-gate.so.1 =>(0xb76ee000)
libdl.so.2 =>/lib/i386-linux-gnu/libdl.so.2 (0xb6f0e000)
libgtk-x11-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgtk-x11-2.0.so.0 (0xb6aa6000)
libpthread.so.0 =>/lib/i386-linux-gnu/libpthread.so.0 (0xb6a8a000)
libstdc++.so.6 =>/usr/lib/i386-linux-gnu/libstdc++.so.6 (0xb69a5000)
libm.so.6 =>/lib/i386-linux-gnu/libm.so.6 (0xb6979000)
libgcc_s.so.1 =>/lib/i386-linux-gnu/libgcc_s.so.1 (0xb695b000)
libc.so.6 =>/lib/i386-linux-gnu/libc.so.6 (0xb67b1000)
```

```
/lib/ld-linux.so.2 (0xb76ef000)
libgdk-x11-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgdk-x11-2.0.so.0 (0xb6701000)
libpangocairo-1.0.so.0 =>/usr/lib/i386-linux-gnu/libpangocairo-1.0.so.0 (0xb66f4000)
libX11.so.6 =>/usr/lib/i386-linux-gnu/libX11.so.6 (0xb65c0000)
libXfixes.so.3 =>/usr/lib/i386-linux-gnu/libXfixes.so.3 (0xb65ba000)
libatk-1.0.so.0 =>/usr/lib/i386-linux-gnu/libatk-1.0.so.0 (0xb659a000)
libcairo.so.2 =>/usr/lib/i386-linux-gnu/libcairo.so.2 (0xb64ce000)
libgdk_pixbuf-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgdk_pixbuf-2.0.so.0 (0xb64ad000)
libgio-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgio-2.0.so.0 (0xb6356000)
libpangoft2-1.0.so.0 =>/usr/lib/i386-linux-gnu/libpangoft2-1.0.so.0 (0xb632a000)
libpango-1.0.so.0 =>/usr/lib/i386-linux-gnu/libpango-1.0.so.0 (0xb62e0000)
libfontconfig.so.1 =>/usr/lib/i386-linux-gnu/libfontconfig.so.1 (0xb62ab000)
libgobject-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgobject-2.0.so.0 (0xb625c000)
libglib-2.0.so.0 =>/lib/i386-linux-gnu/libglib-2.0.so.0 (0xb6163000)
libXext.so.6 =>/usr/lib/i386-linux-gnu/libXext.so.6 (0xb6151000)
libXrender.so.1 =>/usr/lib/i386-linux-gnu/libXrender.so.1 (0xb6147000)
libXinerama.so.1 =>/usr/lib/i386-linux-gnu/libXinerama.so.1 (0xb6142000)
libXi.so.6 =>/usr/lib/i386-linux-gnu/libXi.so.6 (0xb6132000)
libXrandr.so.2 =>/usr/lib/i386-linux-gnu/libXrandr.so.2 (0xb6129000)
libXcursor.so.1 =>/usr/lib/i386-linux-gnu/libXcursor.so.1 (0xb611e000)
libXcomposite.so.1 =>/usr/lib/i386-linux-gnu/libXcomposite.so.1 (0xb611a000)
libXdamage.so.1 =>/usr/lib/i386-linux-gnu/libXdamage.so.1 (0xb6115000)
libfreetype.so.6 =>/usr/lib/i386-linux-gnu/libfreetype.so.6 (0xb607b000)
libxcb.so.1 =>/usr/lib/i386-linux-gnu/libxcb.so.1 (0xb605a000)
libpixman-1.so.0 =>/usr/lib/i386-linux-gnu/libpixman-1.so.0 (0xb5fc2000)
libpng12.so.0 =>/lib/i386-linux-gnu/libpng12.so.0 (0xb5f98000)
libxcb-shm.so.0 =>/usr/lib/i386-linux-gnu/libxcb-shm.so.0 (0xb5f93000)
libxcb-render.so.0 =>/usr/lib/i386-linux-gnu/libxcb-render.so.0 (0xb5f89000)
libz.so.1 =>/lib/i386-linux-gnu/libz.so.1 (0xb5f73000)
libgmodule-2.0.so.0 =>/usr/lib/i386-linux-gnu/libgmodule-2.0.so.0 (0xb5f6e000)
libselinux.so.1 =>/lib/i386-linux-gnu/libselinux.so.1 (0xb5f4f000)
libresolv.so.2 =>/lib/i386-linux-gnu/libresolv.so.2 (0xb5f36000)
libexpat.so.1 =>/lib/i386-linux-gnu/libexpat.so.1 (0xb5f0c000)
libffi.so.6 =>/usr/lib/i386-linux-gnu/libffi.so.6 (0xb5f05000)
libpcre.so.3 =>/lib/i386-linux-gnu/libpcre.so.3 (0xb5ec9000)
librt.so.1 =>/lib/i386-linux-gnu/librt.so.1 (0xb5ec0000)
libXau.so.6 =>/usr/lib/i386-linux-gnu/libXau.so.6 (0xb5ebb000)
libXdmcp.so.6 =>/usr/lib/i386-linux-gnu/libXdmcp.so.6 (0xb5eb4000)
cs@Ubuntu32:
textasciitilde /MeinProgramm/MeinProgramm Libs$
```

As you see all library have been found and their load address is printed behind the na,e. If a library is missing, you usually see the address missing there or being zero.

8.0.46 How can I validate an email address?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:
Example:

```
Dim re As RegEx
re = New RegEx
Dim rm As RegExMatch

re.SearchPattern = ” [ a-z0-9!#$ %&’*+/=?^_‘ { | }
textasciitilde - ] +(?:\. [ a-z0-9!#$ %&’*+/=?^_‘ { | }
textasciitilde - ] +)*@(?: [ a-z0-9 ] (?: [ a-z0-9- ] * [ a-z0-9 ] )?\.)+ [ a-z0-9 ] (?: [ a-z0-9- ] * [ a-z0-9 ] )?”
rm = re.Search(editField1.Text)

if rm = Nil Then
StaticText2.text = editField1.Text + ” not valid email”
Else
StaticText2.Text = editField1.Text + ” is valid”
End if
```

Notes: Adapted from:
http://www.regular-expressions.info/email.html

8.0.47 How do I decode correctly an email subject?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following code can be used to decode an email subject including several encodings including Base 64.
Example:

```
dim src as string // input

dim theRegex as Regex
dim theRegexMatch as RegexMatch
dim result, infoCharset, encodedPart as string
dim theStart as Integer

if instr(src, ”=?”) >0 then
theRegex = new Regex
theRegex.Options.Greedy = false
theRegex.searchPattern = ”(.*)=\?(.+)\?(Q | B)\?(.+)\?=”
theRegexMatch = theRegex.search(src)
```

```
while theRegexMatch <>nil
theStart = theRegexMatch.subExpressionStartB(0) + len(theRegexMatch.subExpressionString(0))

result = result + theRegexMatch.subExpressionString(1)
infoCharset = theRegexMatch.subExpressionString(2)
encodedPart = theRegexMatch.subExpressionString(4)
if theRegexMatch.subExpressionString(3) = ”B” then
encodedPart = DecodeBase64(encodedPart)
elseif theRegexMatch.subExpressionString(3) = ”Q” then
encodedPart = DecodeQuotedPrintable(encodedPart)
end if
if right(result, 1) = ” ” then
result = mid(result, 1, len(result)-1)
end if
encodedPart = encodedPart.DefineEncoding(GetInternetTextEncoding(infoCharset))
result = result + encodedPart

theRegex.SearchStartPosition = theStart
theRegexMatch = theRegex.search()
wend

result = result + mid(src, theStart+1)

else
result = src
end if
// theRegexMatch = theRegex.search

msgbox result
```

Notes: May not look nice depending on the controls used.
This is no longer needed when using MimeEmailMBS class which decodes for you.

8.0.48 How do I enable/disable a single tab in a tabpanel?

Plugin Version: all, Platform: macOS.

Answer: Use the TabpanelEnabledMBS method.
Example:

```
TabpanelEnabledMBS(tabpanel1, 1, false)
```

Notes: Use Carbon for MachO and CarbonLib for Mac Carbon and AppearanceLib for Mac OS Classic as

library.
For Cocoa, please use enabled property of NSTabViewItemMBS class.

8.0.49 How do I find the root volume for a file?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this function:
Example:

```
Function GetRootVolume(f as FolderItem) as FolderItem
dim root, dum as folderItem
if f <>nil then
root = f // f might be the volume
do
dum = root.parent
if dum <>nil then
root = dum
end if
loop until dum = nil
return root
end if
End Function
```

8.0.50 How do I get the current languages list?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim p as new CFPreferencesMBS
dim a as CFArrayMBS
dim s as CFStringMBS
dim o as CFObjectMBS
dim sa(-1) as string

o=p.CopyAppValue("AppleLanguages",".GlobalPreferences")

if o<>Nil then
a=CFArrayMBS(o)

dim i,c as Integer
```

```
c=a.Count-1
for i=0 to c
o=a.Item(i)

if o isa CFStringMBS then
s=CFStringMBS(o)
sa.Append s.str
end if
next
end if

MsgBox Join(sa,EndOfLine)
```

Notes: On Mac OS X you can get the list of current languages like this list:

```
de
en
ja
fr
es
it
pt
pt-PT
nl
sv
nb
da
fi
ru
pl
zh-Hans
zh-Hant
ko
```

Which has German (de) on the top for a German user.
This code has been tested on Mac OS X 10.5 only.

8.0.51 How do I get the Mac OS Version?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim i as Integer
if system.gestalt("sysv", i) then
//do this in an 'If' in case you don't get any value back at all and system.gestalt returns boolean
if i = &h750 then //If OS is 7.5
//do stuff
elseif i = &h761 then //If OS is 7.6.1
//do stuff
end if
end if
```

Notes: The MBS Plugin has a function SystemInformationMBS.OSVersionString for this.

8.0.52 How do I get the printer name?

Plugin Version: all.

Answer: For Mac OS Classic see the code below and for Mac OS X use the Carbon Print Manager Classes from the MBS Plugin.
Example:

```
dim s as String
dim i as Integer

s=app.ResourceFork.GetResource("STR ",-8192)
if s<>"" then
i=ascb(leftb(s,1))
s=mid(s,2,i)

MsgBox s
end if
```

Notes: A note from Craig Hoyt:

After looking at your example I had a little deja-vu experience. Several
years ago I played around with this same code if FutureBasic. I discovered
that it did not and still doesn't provide the 'Printer Name', it does
return the print driver name. If it returns 'LaserWriter 8' as the print
driver you can look into this file and get the 'PAPA' resource #-8192 to
get the actual Printer Name. Unfortunately this does not hold true for
other printers. My Epson and HP Printers (the Epson has an Ethernet Card
and the HP is USB) do not provide this info in their drivers. As far as I
can tell it only returns the name by polling the printer itself.

8.0.53 How do I make a metal window if RB does not allow me this?

Plugin Version: all, Platform: macOS.

Answer: The following declare turns any window on Mac OS X 10.2 or newer into a metal one.
Example:

```
declare sub ChangeWindowAttributes lib "Carbon" (win as windowptr, a as Integer, b as Integer)

ChangeWindowAttributes window1,256,0
```

Notes: May not look nice depending on the controls used.

8.0.54 How do I make a smooth color transition?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

I'd like to show in a report some bars, which start with color A
and end with color B.

The color change should be very smooth.

My problem: If I would start from 255,0,0 and end by 0,0,0, I would have
255 different colors. If the bars are longer than 255 pixels, would
this look nice?
Example:

```
// Window.Paint:
Sub Paint(g As Graphics)
dim w,w1,x,p as Integer
dim c1,c2,c as color
dim p1,p2 as Double

c1=rgb(255,0,0) // start color
c2=rgb(0,255,0) // end color

w=g.Width
w1=w-1

for x=0 to w1
p1=x/w1
p2=1.0-p1
```

```
c=rgb(c1.red*p1+c2.red*p2, c1.green*p1+c2.green*p2, c1.blue*p1+c2.blue*p2)

g.ForeColor=c
g.DrawLine x,0,x,g.Height

next
End Sub
```

Notes:

Try the code above in a window paint event handler.

8.0.55 How do I read the applications in the dock app?

Plugin Version: all, Platform: macOS.

Answer: Use CFPreferencesMBS class like in this example:
Example:

```
// Reads file names from persistent dock applications and puts them into the list

dim pref as new CFPreferencesMBS

dim persistentapps as CFStringMBS = NewCFStringMBS("persistent-apps")
dim ApplicationID as CFStringMBS = NewCFStringMBS("com.apple.dock")
dim tiledata as CFStringMBS = NewCFStringMBS("tile-data")
dim filelabel as CFStringMBS = NewCFStringMBS("file-label")

// get the array of persistent applications from dock preferences
dim o as CFObjectMBS = pref.CopyValue(persistentapps, ApplicationID, pref.kCFPreferencesCurrentUser,
pref.kCFPreferencesAnyHost)

if o isa CFArrayMBS then
dim a as CFArrayMBS = CFArrayMBS(o)

// walk over all items in array
dim c as Integer = a.Count-1
for i as Integer = 0 to c

// get dictionary describing item
o = a.Item(i)

if o isa CFDictionaryMBS then
dim d as CFDictionaryMBS = CFDictionaryMBS(o)
```

```
// and pick tile data dictionary
o = d.Value(tiledata)
if o isa CFDictionaryMBS then
d = CFDictionaryMBS(o)

// and pick there the file label
o = d.Value(filelabel)
if o isa CFStringMBS then
// and display it
dim name as string = CFStringMBS(o).str
List.AddRow name
end if
end if
end if

next

else
MsgBox ”Failed to read dock preferences.”
end if
```

Notes: You can use the CFPreferencesMBS.SetValue to change a value and CFPreferencesMBS.Synchronize to write the values to disc. You may need to restart the Dock.app if you modified things.

8.0.56 How do I truncate a file?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: In a binarystream you can set the length property to truncate.

8.0.57 How do update a Finder's windows after changing some files?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim f as folderitem // some file
dim ae as appleevent
ae=newappleevent(”fndr”,”fupd”,”MACS”)
ae.folderitemparam(”—-”)=f
if not ae.send then
//something went wrong
```

```
end if
```

Notes: The folderitem.finderupdate from the MBS Plugin does something like this.

8.0.58 How to access a USB device directly?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: First, it depends on the device.
Notes: Some devices can be talked directly from user mode code, but some require a kernel driver.

For some devices you can use plugins to access them like:

- Audio and Video sources using the QTGrabberClassMBS
- Mass storage devices using the folderitem class.
- Serial devices using the System.SerialPort function.
- HID USB devices can be used with MacHIDMBS, WinHIDMBS or LinuxHIDInterface class.
- Any USB device may be used with MacUSBMBS or WinUSBMBS classes.

In general it is always the best to take the most high level access to have others do the work for the details.

8.0.59 How to add icon to file on Mac?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use FolderItem.AddCustomIcon or NSWorkspaceMBS.setIcon functions.
Notes: Please close any open stream for the file you want to add an icon.

8.0.60 How to ask the Mac for the Name of the Machine?

Plugin Version: all, Platform: macOS.

Answer: Using Apple Events you can use this code:
Example:

```
Function Computername() As string
```

```
dim theEvent as AppleEvent
dim err as boolean

theEvent = newAppleEvent("mchn","getd","MACS")

err = theEvent.send

return theevent.ReplyString

End Function
```

Notes: Code above is for Mac OS 9!
Also the MBS Plugin has a function for this which may be faster and work also on Macs without Filesharing (which handles this event).

8.0.61 How to automatically enable retina in my apps?

Plugin Version: all, Platform: macOS.

Answer: You can run a build script on each build with this code:
Example:

```
Dim App As String = CurrentBuildLocation + "/" + CurrentBuildAppName + ".app"
Call DoShellCommand("/usr/bin/defaults write " + App + "/Contents/Info ""NSHighResolutionCapable""
YES")
```

Notes: This will set the NSHighResolutionCapable flag to YES.

8.0.62 How to avoid leaks with Cocoa functions?

Plugin Version: all, Platform: macOS.

Answer: You can try this code on Mac OS X:
Example:

```
// in a Timer Action event:
Sub Action()
static LastPool as NSAutoreleasePoolMBS = nil
static CurrentPool as NSAutoreleasePoolMBS = nil

LastPool = CurrentPool
CurrentPool = new NSAutoreleasePoolMBS
```

```
End Sub
```

Notes: With Xojo 2009r4 the code above should not be needed as Xojo runtime does automatically handle the NSAutoreleasePools for you. For older Xojo versions you need to use code with a timer with the action event above to avoid memory leaks.

Please do not use Xojo 2009r4 and newer with plugins before version 9.5. You can get crashes there which typically show a line with a objc_msgSend call.

8.0.63 How to avoid trouble connecting to oracle database with SQL Plugin?

Plugin Version: all, Platform: macOS.

Answer: For oracle the most important thing is to point the plugin to the libraries from oracle.
Notes: In environment variables, the paths like ORACLE_HOME must be defined.
On Mac OS X you also need to define DYLD_LIBRARY_PATH to point to the dylib files from oracle.

For that you need to modify /etc/launchd.conf for Mac OS X 10.8 and newer.
In older versions those variables in .MacOSX/environment.plist file in user's home.

Another way for the case you bundle things inside your app is to use the LSEnvironment key in info.plist.
In info.plist it looks like this:

```
<key>LSEnvironment</key>
<dict>
<key>test</key>
<string>Hello World</string>
</dict>
```

8.0.64 How to avoid ___NSAutoreleaseNoPool console messages in threads?

Plugin Version: all, Platform: macOS.

Answer: You need to use your own NSAutoreleasePool on a thread like this:
Example:

```
sub MyThread.run
dim pool as new NSAutoreleasePoolMBS
// do work here

pool=nil
```

```
end sub
```

Notes: For more details read here:
http://developer.apple.com/mac/library/documentation/Cocoa/Reference/Foundation/Classes/NSAutoreleasePool_Class/Reference/Reference.html

8.0.65 How to bring app to front?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac you can use this code:
Example:

```
// First way:
app.FrontMostMBS = true

// second way:
dim p as new ProcessMBS
p.GetCurrentProcess
p.FrontProcess = true

// third way:
NSApplicationMBS.sharedApplication.activateIgnoringOtherApps(true)

// for Windows:
RemoteControlMBS.WinBringWindowToTop
```

Notes: This will bring a Mac app to the front layer.

8.0.66 How to bring my application to front?

Plugin Version: all, Platform: macOS.

Answer: This makes SimpleText (Code ttxt) to the frontmost application:
Example:

```
Dim A As AppleEvent
A = NewAppleEvent(”misc”,”actv”,””)
If Not A.Send then
Beep
end if
```

Notes: (Code is Mac only)

8.0.67 How to catch Control-C on Mac or Linux in a console app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use SignalHandlerMBS class for this.
Example:

```
// watch for Control-C on Mac
call SignalHandlerMBS.SetFlagHandler(2)

dim ende as boolean = false
do
if SignalHandlerMBS.IsFlagSet(2) then
Print "Flag 2 set. Existing..."
ende = true
end if

DoEvents 1
loop until ende
```

Notes: The signal is catched, a flag is set and you can ask later in your normal application flow for the result.

8.0.68 How to change name of application menu?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Use this code to change the application menu name on Mac OS X:
Example:

```
dim mb as new MenubarMBS
dim m as MenuMBS = mb.item(1) // 1 is in my tests the app menu
if m<>Nil then
m.MenuTitle = "Hello World"
end if
```

Notes: This code is for Carbon only.

8.0.69 How to change the name in the menubar of my app on Mac OS X?

Plugin Version: all, Platform: macOS.

Answer:

You mean it screws up if the file name of the bundle itself is different
than the name of the executable file in the MacOS folder within the
bundle? If so, you should find something like this within your
Info.plist file (or the 'plst' resource that the RB IDE builds for you):

```
<key>CFBundleExecutable</key>
<string>Executable file name here</string>
```

Just make sure that file name matches.

However, if your question involves how you can change the name of the app
that appears in the menu and the dock, that's different. You can make
this name different from the file name by changing the CFBundleName key:

```
<key>CFBundleName</key>
<string>Name for menu here</string>
```

Note that if you use my free AppBundler program, this second part is
taken care of for you – just fill in a custom name in the right field.
You can find AppBundler (from Thomas Reed) at
http://www.bitjuggler.com/products/appbundler/ .

8.0.70 How to check if a folder/directory has subfolders?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this to check all items in a folder:
Example:

```
Function HasSubFolder(folder as FolderItem) As Boolean
dim c as Integer = folder.Count

for i as Integer = 1 to c
dim item as FolderItem = folder.TrueItem(i)

if item<>Nil and item.Directory then
Return true
end if
```

```
next

End Function
```

Notes: We use trueitem() here to avoid resolving alias/link files.
Also we check for nil as we may not have permission to see all items.
And if one is a directory, we return without checking the rest.

8.0.71 How to check if Macbook runs on battery or AC power?

Plugin Version: all, Platform: macOS.

Answer: Please use our IOPowerSourcesMBS class like this:
Example:

```
Function PowerSourceState() as Integer
dim p as new IOPowerSourcesMBS

// check all power sources
dim u as Integer = p.Count-1
for i as Integer = 0 to u
dim d as CFDictionaryMBS = p.Item(i)
if d<>nil then
// check if they have a power source state key:
dim o as CFObjectMBS = d.Value(NewCFStringMBS("Power Source State"))
if o isa CFStringMBS then
dim s as string = CFStringMBS(o).str

'MsgBox s

if s = "AC Power" then
Return 1
elseif s = "Battery Power" then
Return 2
end if
end if
end if
next
Return 0 // unknown
End Function
```

Notes: If you want to check the CFDictionaryMBS content, simply use a line like "dim x as dictionary = d.dictionary" and check the contents in the debugger.

8.0.72 How to check if Microsoft Outlook is installed?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: If you need Outlook for Scripting, you should simply check registry for the required Outlook.Application class:
Example:

```
Function OutlookInstalled() As Boolean
#if TargetWin32 then

try
dim r as new RegistryItem("HKEY_CLASSES_ROOT\Outlook.Application\CLSID", false)

Return true

catch r as RegistryAccessErrorException
// not installed
Return false

end try

#else

// Windows only, so false on other platforms
Return false

#endif

End Function
```

8.0.73 How to check on Mac OS which country or language is currently selected?

Plugin Version: all, Platform: macOS.

Answer: The code below returns a country value.
Example:

```
dim result as Integer

IF TargetMacOS THEN

CONST smScriptLang = 28
CONST smSystemScript = -1

DECLARE FUNCTION GetScriptManagerVariable LIB "Carbon" ( selector as Integer) as Integer
DECLARE FUNCTION GetScriptVariable LIB "Carbon" ( script as Integer, selector as Integer) as Integer

result=GetScriptVariable(smSystemScript, smScriptLang)

END IF
```

Notes: Returns values like:

For more values, check "Script.h" in the frameworks.

8.0.74 How to code sign my app with plugins?

Plugin Version: all, Platform: macOS.

Answer: When you try to code sign the application with plugin dylibs on Mac OS X, you may see error message that there is actually a signature included.
Notes: Please use the -f command line parameter with codesign utility to overwrite our MBS signature.
We sign our plugins for MacOS, iOS and Windows to make sure they have not been modified.

In terminal, you do like this:

```
cd <Path to folder of app>

xattr -cr <Appname>.app
codesign -f -s "Developer ID Application: <Your Name>" <Appname>.app/Contents/Frameworks/*.dylib
codesign -f -s "Developer ID Application: <Your Name>" <Appname>.app/Contents/Frameworks/*.framework
codesign -f -s "Developer ID Application: <Your Name>" <Appname>.app
```

Please use the name of your certificate (See keychain), the name of your app and the path to the app folder.
If you have helper apps you need to sign them first.
You can use a build step to automatically sign your app on build.

8.0.75 How to collapse a window?

Plugin Version: all, Platform: macOS.

Answer: Use this function (Mac only):
Example:

```
Sub CollapseRBwindow(w as window, CollapseStatus as boolean)
dim state, err as Integer
dim wh as MemoryBlock

Declare Function CollapseWindow Lib "Carbon" (window as Integer,collapse as Integer) as Integer

IF CollapseStatus THEN
state = 1
ELSE
state = 0
END IF

err = CollapseWindow(w.MacWindowPtr, state)

End Sub
```

Notes: Also the MBS Plugin has a window.collapsedmbs property you can set.
For Windows the MBS Plugin has a window.isiconicmbs property.

8.0.76 How to compare two pictures?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:
Example:

```
Function ComparePictures(p as picture,q as picture) as Integer
dim r,u as RGBSurface
dim x,y,n,m,h,w as Integer
dim w1,w2,h1,h2,d1,d2 as Integer
dim c1,c2 as color

h1=p.Height
h2=q.Height
w1=p.Width
w2=q.Width
d1=p.Depth
d2=q.Depth

if d1<>d2 then
Return 1
elseif w1<>w2 then
```

```
return 2
elseif h1<>h2 then
Return 3
else
r=p.RGBSurface
u=q.RGBSurface

if r=nil or u=nil then
Return -1
else
h=h1-1
w=w1-1
m=min(w,h)

for n=0 to m
c1=r.Pixel(n,n)
c2=u.Pixel(n,n)
if c1<>c2 then
Return 4
end if
next

for y=0 to h
for x=0 to w
c1=r.Pixel(x,y)
c2=u.Pixel(x,y)
if c1<>c2 then
Return 5
end if
next
next

// 0 for equal
// -1 for error (no RGBsurface)
// 1 for different depth
// 2 for different width
// 3 for different height
// 4 for different pixels (fast test)
// 5 for different pixels (slow test)
end if
end if

Exception
Return -1
End Function
```

Notes: Remember that this only works on bitmap pictures, so the picture.BitmapMBS function may be useful.

8.0.77 How to compile PHP library?

Plugin Version: all, Platform: macOS.

Answer: You have to download the source code and compile a static version of the library.
Notes: This instructions were written based on PHP 5.2.6 on Mac OS X:

- Best take a new Mac with current Xcode version installed.
- Download the source code archive. e.g. ”php-5.2.6.tar.bz2”
- Expand that archive on your harddisc.
- Open terminal window
- change directory to the php directory. e.g. ”cd /php-5.2.6”
- execute this two lines to define the supported CPU types and the minimum Mac OS X version:
- export CFLAGS=”-arch ppc -arch i386 -mmacosx-version-min=10.3”
- export CXXFLAGS=”-arch ppc -arch i386 -mmacosx-version-min=10.3”
- the command ”./configure help” does show the configure options.
- use configure with a line like this:
- ./configure –enable-embed –with-curl -enable-ftp –enable-zip –enable-sockets –enable-static –enable-soap –with-zlib –with-bz2 –enable-exif –enable-bcmath –enable-calendar
- start the compilation with ”make all”
- other option is to use ”make install” which first does the same as ”make all” and than does some installation scripts.
- you may get an error about a duplicate symbole __yytext. Search the file ”zend_ini_scanner.c”, search a line with ”char *yytext;” and change it to ”extern char *yytext;”.
- On the end you get a lot of error messages, but you have a working library (named libphp5.so) file in the invisible ”.libs” folder inside your php source folder.

Possible problems and solutions:

- If the path to your files has spaces, you can get into trouble. e.g. ”/RB Plugins/PHP” is bad as files will be searched sometimes in ”/RB”.

- If you have in /usr/local/lib libraries which conflict with the default libraries, you can get into trouble.
- If you installed some open source tools which compiled their own libraries, you can get into conflicts.
- if you have to reconfigure or after a problem, you may need to use "make clean" before you start "make all" again.

Feel free to install additional libraries and add more packages to the configure line.

8.0.78 How to convert a BrowserType to a String with WebSession.Browser?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this:
Example:

```
Function GetBrowserName(s as WebSession.BrowserType) As string
Select case s
case WebSession.BrowserType.Android
Return "Andriod"
case WebSession.BrowserType.Blackberry
Return "Blackberry"
case WebSession.BrowserType.Chrome
Return "Chrome"
case WebSession.BrowserType.ChromeOS
Return "ChromeOS"
case WebSession.BrowserType.Firefox
Return "Firefox"
case WebSession.BrowserType.InternetExplorer
Return "InternetExplorer"
case WebSession.BrowserType.Opera
Return "Opera"
case WebSession.BrowserType.Safari
Return "Safari"
case WebSession.BrowserType.SafariMobile
Return "SafariMobile"
case WebSession.BrowserType.Unknown
Return "Unknown"
else
Return "Unkown: "+str(integer(s))
end Select

End Function
```

8.0.79 How to convert a EngineType to a String with WebSession.Engine?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this:
Example:

```
Function GetRenderingEngineName(s as WebSession.EngineType) As string
Select case s
case WebSession.EngineType.Gecko
Return "Gecko"
case WebSession.EngineType.Presto
Return "Presto"
case WebSession.EngineType.Trident
Return "Trident"
case WebSession.EngineType.Unknown
Return "Unknown"
case WebSession.EngineType.WebKit
Return "WebKit"
else
Return "Unkown: "+str(integer(s))
end Select

End Function
```

8.0.80 How to convert a PlatformType to a String with WebSession.Platform?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this:
Example:

```
Function GetPlatformName(s as WebSession.PlatformType) As string
Select case s
case WebSession.PlatformType.Blackberry
Return "Blackberry"
case WebSession.PlatformType.iPad
Return "iPad"
case WebSession.PlatformType.iPhone
Return "iPhone"
case WebSession.PlatformType.iPodTouch
Return "iPodTouch"
case WebSession.PlatformType.Linux
Return "Linux"
case WebSession.PlatformType.Macintosh
Return "Macintosh"
```

```
case WebSession.PlatformType.PS3
Return "PS3"
case WebSession.PlatformType.Unknown
Return "Unknown"
case WebSession.PlatformType.WebOS
Return "WebOS"
case WebSession.PlatformType.Wii
Return "Wii"
case WebSession.PlatformType.Windows
Return "Windows"
else
Return "Unkown: "+str(integer(s))
end Select

End Function
```

8.0.81 How to convert a text to iso-8859-1 using the TextEncoder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

This code can help you althrough it's not perfect.
You need to set lc to the current color you use.
Example:

```
dim outstring as string
dim theMac, thePC as textencoding
dim Mac2PC as textconverter

theMac = getTextEncoding(0) // MacRoman
thePC = getTextEncoding(&h0201) // ISOLatin1

Mac2PC = getTextConverter(theMac, thePC)
// if you wanted to do the opposite just create a converter
// PC2Mac = getTextConverter(thePC, theMac)

outstring = Mac2PC.convert("Bj√rn, this text should be converted")
Mac2PC.clear
```

Notes:

You have to call Mac2PC.clear after every conversion to reset the encoding engine.
See also newer TextConverterMBS class.

8.0.82 How to convert ChartTime back to Xojo date?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We have this example code:
Example:

```
Function ChartTimeToDate(ChartTime as Double) As date
static diff as Double = 0.0

if diff = 0.0 then
dim d2 as Double = CDBaseChartMBS.chartTime(2015, 1, 1)
dim da as new date(2015, 1, 1)
dim ts as Double = da.TotalSeconds

diff = ts - d2
end if

dim d as new date
d.TotalSeconds = diff + ChartTime

Return d
End Function
```

Notes: As you see we calculate the difference in base date from Date and ChartTime and later use difference to convert.

8.0.83 How to convert line endings in text files?

Plugin Version: all, Platform: macOS.

Answer: You can simply read file with TextInputStream and write with new line endings using TextOutputStream class.
Example:

```
dim inputfile as FolderItem = SpecialFolder.Desktop.Child(”test.txt”)
dim outputfile as FolderItem = SpecialFolder.Desktop.Child(”output.txt”)
dim it as TextInputStream = TextInputStream.Open(inputfile)
dim ot as TextOutputStream = TextOutputStream.Create(outputfile)

ot.Delimiter = EndOfLine.Windows // new line ending
while not it.EOF
ot.WriteLine it.ReadLine
wend
```

Notes: TextInputStream will read any input line endings and with delimiter property in TextOutputStream you can easily define your new delimiter.

8.0.84 How to convert picture to string and back?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use this plugin functions:
Notes: JPEG:

JPEGStringToPictureMBS(buf as string) as picture
JPEGStringToPictureMBS(buf as string,allowdamaged as Boolean) as picture
PictureToJPEGStringMBS(pic as picture,quality as Integer) as string

PNG:

PictureToPNGStringMBS(pic as picture, gamma as single) as string
PictureToPNGStringMBS(pic as picture, mask as picture, gamma as single) as string
PictureToPNGStringMBS(pic as picture, gamma as single, Interlace as Boolean, FilterType as Integer) as string
PictureToPNGStringMBS(pic as picture, mask as picture, gamma as single, Interlace as Boolean, FilterType as Integer) as string
PNGStringToPictureMBS(data as string, gamma as single) as picture
PNGStringToPNGPictureMBS(data as string, gamma as single) as PNGpictureMBS

Tiff:

TIFFStringToPictureMBS(data as string) as picture
TIFFStringToTiffPictureMBS(data as string) as TiffPictureMBS

BMP:

BMPStringtoPictureMBS(data as string) as picture
Picture.BMPDataMBS(ResolutionValueDPI as Integer=72) as string

GIF:

GifStringToGifMBS(data as string) as GIFMBS
GifStringToPictureMBS(data as string) as Picture

8.0.85 How to copy an array?

Plugin Version: all, Platform: macOS.

Answer: You can use a function like this to copy an array:
Example:

```
Function CopyArray(a() as Double) as Double()
dim r() as Double
for each v as Double in a
r.Append v
next
Return r
End Function
```

Notes: If needed make several copies of this method with different data types, not just double.
For a deep copy of an array of objects, you need to change code to also make a copy of those objects.

8.0.86 How to copy an dictionary?

Plugin Version: all, Platform: macOS.

Answer: You can use a function like this to copy a dictionary:
Example:

```
Function CopyDictionary(d as Dictionary) As Dictionary
dim r as new Dictionary
for each key as Variant in d.keys
r.Value(key) = d.Value(key)
next
Return r
End Function
```

Notes: If needed make several copies of this method with different data types, not just double.
For a deep copy of an dictionary of objects, you need to change code to also make a copy of those objects.

8.0.87 How to copy parts of a movie to another one?

Plugin Version: all, Platforms: macOS, Windows.

Answer: The code below copies ten seconds of the snowman movie to the dummy movie starting at the 5th second.

Example:

```
dim f as FolderItem
dim md as EditableMovie
dim ms as EditableMovie

f=SpecialFolder.Desktop.Child("Our First Snowman.mov")
ms=f.OpenEditableMovie

ms.SelectionStartMBS=5
ms.SelectionLengthMBS=10

f=SpecialFolder.Desktop.Child("dummy.mov")
md=f.CreateMovie

msgbox str(md.AddMovieSelectionMBS(ms))
```

Notes: If result is not 0, the method fails.

8.0.88 How to create a birthday like calendar event?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
// start a connection to the calendar database
dim s as new CalCalendarStoreMBS

// needed for the error details
dim e as NSErrorMBS

dim r as CalRecurrenceRuleMBS = CalRecurrenceRuleMBS.initYearlyRecurrence(1, nil) // repeat every year without end

dim a as new CalAlarmMBS // add alarm
a.action = a.CalAlarmActionDisplay
a.relativeTrigger = -3600*24 // 24 Hours before

// create a new calendar
dim c as new CalEventMBS

dim d as new date(2011, 04, 20) // the date

dim calendars() as CalCalendarMBS = s.calendars
```

```
// set properties
c.Title="Test Birthday"
c.startDate=d
c.recurrenceRule = r
c.calendar=calendars(0) // add to first calendar
c.addAlarm(a)
c.endDate = d
c.isAllDay = true

// save event
call s.saveEvent(c,s.CalSpanAllEvents, e)
if e<>nil then
MsgBox e.localizedDescription
else
MsgBox "New event was created."
end if
```

Notes: This adds an event to iCal for the given date with alarm to remember you and repeats it every year.

8.0.89 How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

8.0.90 How to create a Mac picture clip file?

Plugin Version: all, Platform: Windows.

Answer: You can use code like this one.
Example:

```
dim f As FolderItem
dim p As Picture

f=SpecialFolder.Desktop.Child("Test.pictClipping")
if f=nil then Return

p=new Picture(300,200,32) 'Make a sample picture
p.Graphics.ForeColor=RGB(0,255,255)
p.Graphics.FillOval 0,0,99,99
```

```
p.Graphics.ForeColor=RGB(255,0,0)
p.Graphics.DrawOval 0,0,99,99

dim r As ResourceFork 'ResourceFork is needed for a clip file

// Please define a file type Any
r=f.CreateResourceFork(”Any”)

// get PICT data using plugin function
dim pictdata as string = p.PicHandleDataMBS
r.AddResource(pictdata,”PICT”,256,”Picture”)

dim m as new MemoryBlock(8)

m.LittleEndian = false
m.Int16Value(0) = 0
m.Int16Value(2) = 0
m.Int16Value(4) = p.Width
m.Int16Value(6) = p.Height

r.AddResource(m,”RECT”,256,””)

'Values taken from a sample file and irrelevant to the problem
dim data as string = DecodeBase64(”AQAAAAAAAAAAAAAAAACAFRDRVIAAAABAAAAAAAAAABUQ0lQAAAAA
r.AddResource(data,”drag”,128,””) 'ditto
r.Close
```

Notes: In general Apple has deprecated this, but a few application still support clippings.

8.0.91 How to create a PDF file in Xojo?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Check our DynaPDF plugin and the examples.
Notes: An alternative can be to use the CoreGraphics and Cocoa functions on Mac OS X.
For Windows, we can only suggest our DynaPDF plugin.

8.0.92 How to create EmailAttachment for PDF Data in memory?

Plugin Version: all, Platform: macOS.

Answer: You can use code like the one below:
Example:

```
Function EmailAttachmentFromPDFData(PDFData as string, filename as string) As EmailAttachment
dim a as new EmailAttachment

a.data = EncodeBase64(PDFData, 76)
a.ContentEncoding = "base64"
a.MIMEType = "application/pdf"
a.MacType = "PDF "
a.MacCreator = "prvw"
a.Name = filename

Return a
End Function
```

Notes: Compared to sample code from Xojo documentation, we set the mime type correct for PDF.
The MacType/MacCreator codes are deprecated, but you can still include them for older Mac email clients.
"prvw" is the creator code for Apple's preview app.

8.0.93 How to create PDF for image files?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use DynaPDF like this:
Example:

```
Function CreatePrintPDF(jpgFiles() as folderitem, pdfFile as FolderItem, PageWidth as Integer, PageHeight as Integer) As Boolean
// have files?
If pdfFile = Nil Then Return False
If jpgFiles = Nil Then Return False

If jpgFiles.Ubound <0 Then Return False

// new DynaPDF
Dim pdf As New MyDynapdfMBS

// page width/height in MilliMeter
Dim pdfWidth as Integer = PageWidth * 72 / 25.4
Dim pdfHeight as Integer = PageHeight * 72 / 25.4

// put your license here
Call pdf.SetLicenseKey "Starter"

// create pdf
Call pdf.CreateNewPDF pdfFile
```

```
// set a couple of options
Call pdf.SetPageCoords(MyDynaPDFMBS.kpcTopDown)
Call pdf.SetResolution(300)
Call pdf.SetUseTransparency(False)
Call pdf.SetSaveNewImageFormat(False)
Call pdf.SetGStateFlags(MyDynaPDFMBS.kgfUseImageColorSpace, False)
Call pdf.SetJPEGQuality(100)

// set page size
Call pdf.SetBBox(MyDynaPDFMBS.kpbMediaBox, 0, 0, pdfWidth, pdfHeight)
Call pdf.SetPageWidth(pdfWidth)
Call pdf.SetPageHeight(pdfHeight)

// append pages with one image per page
For i as Integer = 0 To jpgFiles.Ubound
Call pdf.Append
Call pdf.InsertImageEx(0, 0, pdfWidth, pdfHeight, jpgFiles(i), 1)
Call pdf.EndPage
Next

// close
Call pdf.CloseFile

Return True
End Function
```

Notes: This is to join image files in paper size to a new PDF.
e.g. scans in A4 into an A4 PDF.

8.0.94 How to CURL Options translate to Plugin Calls?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Below a few tips on how to translate command line CURL calls to plugin calls.
Notes: curl -vX PUT http://localhost:5984/appserials/78569238475/DocumentRegister.docx?rev=3-25634563456 –data-binary @DocumentRegister.docx -H ”Content-Type: application/msword”

- The option -v means verbose. You can use OptionVerbose and listen for messages in the DebugMessage event.
- The option -X PUT means we want to do a HTTP PUT Request. So set OptionPut to true. Also you will want to set OptionUpload to true as you upload data.
- We have the URL which you put into OptionURL property.

- The –data-binary option tells CURL to pass the given data. With the @ before the data, it is intrepreted as a file name, so the data is read from the given file. You'll need to open this file and pass data with the Read event as needed. (See CURLS ftp file upload example project)
- The last option -H specifies an additional header for the upload. Pas this additional header with the SetOptionHTTPHeader method.

curl -X PUT http://127.0.0.1:5984/appserials/f2f4e540bf8bb60f61cfcd4328001c59 -d ' { "type":"Product","description":"Application Serial","acronym":"AppSerial","dateAdded":"2011-03-21 14:57:36" } '

- Option -X PUT like above.
- Pass the URL again in OptionURL
- This time data is passed in command line for CURL. You'd put this data in the quotes into a string and make it available in the Read event. (See CURLS ftp upload example project)

8.0.95 How to delete file with ftp and curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can set post/pre quotes to have ftp commands executed before or after the download/upload.
Example:

```
dim d as CURLMBS // your curl object

// delete file
dim ws() As String
ws.Append "DELE Temp.txt"

d.SetOptionPostQuote(ws)
```

Notes: Use SetOptionPostQuote, SetOptionPreQuote or SetOptionQuote.
The ftp commands you pass here are native ftp commands and not the commands you use with ftp applications. To delete use DELE and the file path.

8.0.96 How to detect display resolution changed?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac OS X simply listen for display changed notifications.
Notes: Use the "Distribution Notification Center.rbp" example project as a base and use it to listen to notifications with the name "O3DeviceChanged".

8.0.97 How to detect retina?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use Window.BackingScaleFactorMBS to query the factor.
Example:

```
msgbox str(window1.BackingScaleFactorMBS)
```

8.0.98 How to disable force quit?

Plugin Version: all, Platform: macOS.

Answer:

Please visit this website and get the control panel for Mac OS 9 there:
http://www3.sk.sympatico.ca/tinyjohn/DFQ.html

For Mac OS X use the MBS Plugin with the SetSystemUIModeMBS method.
Notes:

Please use presentationOptions in NSApplicationMBS for Cocoa applications.

8.0.99 How to disable the error dialogs from Internet Explorer on javascript errors?

Plugin Version: all, Platform: Windows.

Answer: You can use this code in the htmlviewer open event:
Example:

```
if targetwin32 then
htmlviewer1._ole.Content.value("Silent") = True
end if
```

Notes: This disables the error dialogs from Internet Explorer.

8.0.100 How to display a PDF file in Xojo?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac OS X you can use CoreGraphics or PDFKit to display a PDF.
Notes: An alternative can be to load the PDF into a htmlviewer so the PDF plugin can display it.
On Windows you may need to use the Acrobat ActiveX control from Adobe or launch Acrobat Reader.

8.0.101 How to do a lottery in RB?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this function:
Example:

```
Sub Lotto(max as Integer,count as Integer,z() as Integer)
// Lotto count numbers of max put into the array z beginning at index 0
dim n(0) as Integer ' all the numbers
dim m as Integer ' the highest field in the current array
dim i,a,b,d as Integer ' working variables

'fill the array with the numbers
m=max-1
redim n(m)

for i=0 to m
n(i)=i+1
next

' unsort them by exchanging random ones
m=max*10
for i=1 to m
a=rnd*max
b=rnd*max

d=n(a)
n(a)=n(b)
n(b)=d
next

' get the first count to the dest array
m=count-1
redim z(m)
for i=0 to m
z(i)=n(i)
next

'sort the result
z.sort
End Sub
```

```
Sub Open()
// Test it

dim za(0) as Integer ' the array of the numbers

lotto 49,6,za ' 6 of 49 in Germany


' and display them
staticText1.text=str(za(0))+chr(13)+str(za(1))+chr(13)+str(za(2))+chr(13)+str(za(3))+chr(13)+str(za(4))+chr(13)+str(za
End Sub
```

8.0.102 How to do an asycron DNS lookup?

Plugin Version: all, Platform: Windows.

Answer: use CFHostMBS class (Mac OS X only).
Notes: Xojo internal functions and plugin DNS functions are sycronized.
You can use DNSLookupThreadMBS class for doing them asyncron.

8.0.103 How to draw a dushed pattern line?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:
Example:

```
// call like this: DrawDushedPatternLine g,0,0,width,height,10

Sub DrawDushedPatternLine(g as graphics,x1 as Integer,y1 as Integer,x2 as Integer,y2 as Integer, partlen
as Integer)
dim x,y,ox,oy as Double
dim dx,dy as Double
dim w,h,d as Double
dim b as Boolean

w=x2-x1
h=y2-y1

d=sqrt(w*w+h*h)

dx=w/d*partlen
dy=h/d*partlen
```

```
b=true
x=x1
while (x<x2) and (y<y2)
ox=x
oy=y

x=x+dx
y=y+dy

if b then
g.DrawLine ox,oy,x,y
end if

b=not b
wend


End Sub
```

Notes: It would be possible to add this to the plugin, but I think it's better if you do it in plain Xojo code, so it even works on Windows.

8.0.104 How to draw a nice antialiased line?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

This code can help you althrough it's not perfect.
You need to set lc to the current color you use.
Example:

```
Sub drawLine(xs as Integer, ys as Integer, xe as Integer, ye as Integer, face as RGBSurface, lineColor as color)
dim intX, intY, count, n, xDiff, yDiff as Integer
dim v, v1, floatX, floatY, xx, yy, xStep, yStep as Double
dim c as color

const st=1.0

xDiff=xe-xs
yDiff=ye-ys
count=max(abs(xDiff), abs(yDiff))
xStep=xDiff/count
yStep=yDiff/count
```

```
xx=xs
yy=ys
for n=1 to count
intX=xx
intY=yy
floatX=xx-intX
floatY=yy-intY

v=(1-floatX)*(1-floatY)*st
v1=1-v
c=face.pixel(intX, intY)
face.pixel(intX, intY)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)
v=floatX*(1-floatY)*st
v1=1-v
c=face.pixel(intX+1, intY)
face.pixel(intX+1, intY)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)
v=(1-floatX)*floatY*st
v1=1-v
c=face.pixel(intX, intY+1)
face.pixel(intX, intY+1)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)
v=floatX*floatY*st
v1=1-v
c=face.pixel(intX+1, intY+1)
face.pixel(intX+1, intY+1)=rgb(v*lineColor.red+v1*c.red, v*lineColor.green+v1*c.green, v*lineColor.blue+v1*c.blue)

xx=xx+xStep
yy=yy+yStep
next

End Sub
```

Notes:

PS: st should be 1 and face should be a RGBSurface or a Graphics object.

8.0.105 How to dump java class interface?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: In terminal you can use "javap -s <classname>" to display the class with the method names and parameters.
Notes: For example show ResultSet class: javap -s java.sql.ResultSet

8.0.106 How to duplicate a picture with mask or alpha channel?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this function:
Example:

```
Function Duplicate(extends p as Picture) As Picture
#if RBVersion >= 2011.04 then
if p.HasAlphaChannel then

// create nw picture and copy content:
dim q as new Picture(p.Width, p.Height)
q.Graphics.DrawPicture p,0,0

Return q

end if
#endif

// create new picture
dim q as new Picture(p.Width, p.Height, 32)

// get mask
dim oldMask as Picture = p.mask(false)
if oldMask = nil then
// no mask, so simple copy
q.Graphics.DrawPicture p,0,0
Return q
end if

// remove mask
p.mask = nil

// copy picture and mask
q.Graphics.DrawPicture p, 0, 0
q.mask.Graphics.DrawPicture oldMask,0,0

// restore mask
p.mask = oldmask

Return q
End Function
```

Notes: Simply copy it to a module and call it like this: q = p.duplicate.
The code above works with old Xojo versions because of the #if even if your RS version does not support alpha channel pictures. This way it's future proof.

8.0.107 How to enable assistive devices?

Plugin Version: all, Platform: macOS.

Answer: You can use AppleScript code like below:
Notes: tell application ”System Events”
activate

set UI elements enabled to true

return UI elements enabled
end tell

You can run this with AppleScriptMBS class.

8.0.108 How to encrypt a file with Blowfish?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:
Example:

```
dim fi as FolderItem = SpecialFolder.Desktop.Child(”test.xojo_binary_project”)
dim fo as FolderItem = SpecialFolder.Desktop.Child(”test.encrypted”)

// read input
dim bi as BinaryStream = BinaryStream.Open(fi)
dim si as string = bi.Read(bi.Length)
bi.Close

// encrypt
dim so as string = BlowfishMBS.Encrypt(”MyKey”,si)

// write output
dim bo as BinaryStream = BinaryStream.Create(fo)
bo.Write so
bo.Close
```

Notes: Of course you can decrypt same way, just use Decrypt function and of course swap files.

8.0.109 How to extract text from HTML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use both RemoveHTMLTagsMBS and DecodingFromHTMLMBS like this:
Example:

```
dim html as string = ”<p><B>Gr&uuml;&szlig;e</B></P>”
dim htmltext as string = RemoveHTMLTagsMBS(html)
dim text as string = DecodingFromHTMLMBS(htmltext)

MsgBox text // shows: Gr√º√üe
```

Notes: You can use it together with RemoveHTMLTagsMBS to remove html tags. What you get will be the text without tags.
DecodingFromHTMLMBS turns HTML escapes back to unicode characters. Like &auml; to √§.

8.0.110 How to find empty folders in a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this code:
Example:

```
dim folder as folderitem // your folder

dim c as Integer = folder.count
for i as Integer = 1 to c
dim item as folderitem = folder.trueitem(i)
if item = nil then
// ignore
elseif item.directory then
// folder
if item.count = 0 then
// found empty folder
end if
end if
next
```

8.0.111 How to find iTunes on a Mac OS X machine fast?

Plugin Version: all, Platform: macOS.

Answer: You can try Launch Services.
Example:

```
dim f as FolderItem

f=LaunchServicesFindApplicationForInfoMBS("hook","com.apple.iTunes","iTunes.app")

MsgBox f.NativePath
```

8.0.112 How to find network interface for a socket by it's name?

Plugin Version: all, Platform: macOS.

Answer: You can use our plugin to build a lookup table.
Example:

```
Function FindNetworkInterface(name as string) As NetworkInterface
name = name.trim

if name.len = 0 then Return nil

// search by IP/MAC
dim u as Integer = System.NetworkInterfaceCount-1
for i as Integer = 0 to u
dim n as NetworkInterface = System.GetNetworkInterface(i)
if n.IPAddress = name or n.MACAddress = name then
Return n
end if
next

// use MBS Plugin to build a mapping
dim interfaces() as NetworkInterfaceMBS = NetworkInterfaceMBS.AllInterfaces
dim map as new Dictionary

for each n as NetworkInterfaceMBS in interfaces
dim IPv4s() as string = n.IPv4s
dim IPv6s() as string = n.IPv6s

for each IPv4 as string in IPv4s
map.Value(IPv4) = n.Name
next
for each IPv6 as string in IPv6s
map.Value(IPv6) = n.Name
next
if n.MAC<>"" then
map.Value(n.MAC) = n.Name
```

```
end if
next

// now search interfaces by name, IPv4 or IPv6
for i as Integer = 0 to u
dim n as NetworkInterface = System.GetNetworkInterface(i)
if map.Lookup(n.IPAddress, "") = name then
Return n
end if

if map.Lookup(n.MACAddress, "") = name then
Return n
end if
next

End Function
```

Notes: The code above uses a lookup table build using NetworkInterfaceMBS class to find the network interface by name.

8.0.113 How to find version of Microsoft Word?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:
Example:

```
// find Word
dim f as FolderItem = LaunchServicesFindApplicationForInfoMBS("","com.microsoft.Word","")

// open bundle
dim c as new NSBundleMBS(f)

// read info
dim d as Dictionary = c.infoDictionary

// show version
MsgBox d.Lookup("CFBundleVersion","")
```

Notes: Older versions of Word can be found with creator code "MSWD".

8.0.114 How to fix CURL error 60/53 on connecting to server?

Plugin Version: all, Platform: macOS.

Answer: You probably connect with SSL and you have no valid certificate.
Example:

```
dim d as new CURLSMBS

// Disable SSL verification
d.OptionSSLVerifyHost = 0 // don't verify server
d.OptionSSLVerifyPeer = 0 // don't proofs certificate is authentic

// With SSL Verification:
dim cacert as FolderItem = Getfolderitem("cacert.pem")
d.OptionCAInfo = cacert.NativePath
d.OptionSSLVerifyHost = 2 // verify server
d.OptionSSLVerifyPeer = 1 // proofs certificate is authentic
```

Notes: You can either use the code above to disable the SSL verification and have no security.
Or you use the cacert file and enable the verification. Than you only get a connection if the server has a valid certificate.

see also:
http://curl.haxx.se/ca/

8.0.115 How to format double with n digits?

Plugin Version: all, Platform: macOS.

Answer: You can use the FormatMBS function for this.
Example:

```
dim d as Double = 123.4567890
listbox1.AddRow FormatMBS("%f", d)
listbox1.AddRow FormatMBS("%e", d)
listbox1.AddRow FormatMBS("%g", d)

listbox1.AddRow FormatMBS("%5.5f", d)
listbox1.AddRow FormatMBS("%5.5e", d)
listbox1.AddRow FormatMBS("%5.5g", d)

d = 0.000000123456
listbox1.AddRow FormatMBS("%f", d)
listbox1.AddRow FormatMBS("%e", d)
```

```
listbox1.AddRow FormatMBS("%g", d)

listbox1.AddRow FormatMBS("%5.5f", d)
listbox1.AddRow FormatMBS("%5.5e", d)
listbox1.AddRow FormatMBS("%5.5g", d)
```

Notes: see FormatMBS for details.
In general %f is normal style, %e is scientific and %g is whichever gives best result for given space.

8.0.116 How to get a time converted to user time zone in a web app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the WebSession.GMTOffset property.
Example:

```
Sub Open()
// current date on server
dim d as new date
dim s as string = d.LongTime

// adjust to client GMT offset
d.GMTOffset = d.GMTOffset + Session.GMTOffset

dim t as string = D.LongTime

MsgBox s+EndOfLine+t
End Sub
```

8.0.117 How to get an handle to the frontmost window on Windows?

Plugin Version: all, Platform: Windows.

Answer: This function returns a handle for the frontmost window:
Example:

```
Function GetForegroundWindowHandle() as Integer
#if targetwin32 then
declare function GetForegroundWindow Lib "user32.dll" as Integer
Return GetForegroundWindow()
#endif
End Function
```

8.0.118 How to get CFAbsoluteTime from date?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Use code like this:
Example:

```
dim d as new date
dim t as CFTimeZoneMBS = SystemCFTimeZoneMBS
dim g as new CFGregorianDateMBS
g.Day = d.Day
g.Month = d.Month
g.Year = d.Year
g.Minute = d.Minute
g.Hour = d.Hour
g.Second = d.Second

dim at as CFAbsoluteTimeMBS = g.AbsoluteTime(t)
dim x as Double = at.Value

MsgBox str(x)
```

Notes: As you see we need a timezone and put the date values in a gregorian date record.
Now we can query absolute time for the given timezone.

8.0.119 How to get client IP address on web app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the WebSession.RemoteAddress property.
Example:

```
Sub Open()
Title = Session.RemoteAddress
End Sub
```

8.0.120 How to get fonts to load in charts on Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use the SetFontSearchPath method in the CDBaseChartMBS class to specify where your fonts are.
Example:

```
if TargetLinux then
CDBaseChartMBS.SetFontSearchPath "/usr/share/fonts/truetype;/usr/share/fonts/truetype/msttcorefonts"
else
// on Mac and Windows we use system fonts.
end if

// also you can later switch default fonts:

dim Chart as CDBaseChartMBS // your chart

#If TargetARM And TargetLinux Then
// use specific fonts on Linux on Raspberry Pi
Call Chart.setDefaultFonts("/usr/share/fonts/truetype/piboto/PibotoLt-Regular.ttf","/usr/share/fonts/truetype/piboto/Pi
#EndIf
```

Notes: On macOS, iOS and Windows, the fonts are loaded from the system's font folder.

e.g. if you use ubuntu, you can install the ttf-mscorefonts-installer package and call this method with "/usr/share/fonts/truetype/msttcorefonts" as the path. No backslash on the end of a path, please.

8.0.121 How to get fonts to load in DynaPDF on Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use the AddFontSearchPath method in the DynaPDFMBS class to specify where your fonts are.
Example:

```
dim d as new DynaPDFMBS
if TargetLinux then
call d.AddFontSearchPath "/usr/share/fonts/truetype", true
else
// on Mac and Windows we use system fonts.
end if
```

Notes: On Mac OS X and Windows, the fonts are loaded from the system's font folder.

e.g. if you use ubuntu, you can install the ttf-mscorefonts-installer package and call this method with "/usr/share/fonts/truetype/msttcorefonts" as the path. No backslash on the end of a path, please.

8.0.122 How to get GMT time and back?

Plugin Version: all, Platform: macOS.

Answer: You can use the date class and the GMTOffset property.
Example:

```
// now
dim d as new date

// now in GMT
dim e as new date
e.GMTOffset = 0

// show
MsgBox str(d.TotalSeconds,"0.0")+" "+str(e.TotalSeconds, "0.0")

dim GMTTimeStamp as Double = e.TotalSeconds

// restore
dim f as new date

// add GMT offset here
f.TotalSeconds = GMTTimeStamp + f.GMTOffset*3600
// because here it's removed
f.GMTOffset = f.GMTOffset

MsgBox d.ShortTime+" ("+str(d.GMTOffset)+") "+str(d.TotalSeconds,"0.0")+EndOfLine+_
e.ShortTime+" ("+str(e.GMTOffset)+") "+str(e.TotalSeconds,"0.0")+EndOfLine+_
f.ShortTime+" ("+str(f.GMTOffset)+") "+str(f.TotalSeconds,"0.0")
```

Notes: It's sometimes a bit tricky with the date class as setting one property often changes the others.

8.0.123 How to get good crash reports?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Check this website from the webkit website:
Notes: http://webkit.org/quality/crashlogs.html

8.0.124 How to get list of all threads?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use the runtime module like in this function:
Example:

```
Function Threads() As Thread()
#pragma DisableBackgroundTasks
dim t() as Thread

Dim o as Runtime.ObjectIterator=Runtime.IterateObjects
While o.MoveNext
if o.Current isa Thread then
t.Append thread(o.current)
end if
Wend

Return t
End Function
```

Notes: This returns an array of all thread objects currently in memory.
The pragma is important here as it avoids thread switches which may cause a thread to be created or deleted.

8.0.125 How to get parameters from webpage URL in Xojo Web Edition?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the Webpage.ParametersReceived event.
Example:

```
Sub ParametersReceived(Variables As Dictionary)
for each key as Variant in Variables.keys
MsgBox key+" ->"+Variables.Value(key)
next
End Sub
```

Notes: The text encodings of this strings is not defined in Xojo 2010r5. Please use DefineEncoding.

8.0.126 How to get the color for disabled textcolor?

Plugin Version: all, Platform: macOS.

Answer: Ask the appearance manager:
Example:

```
Function GetThemeTextColor(inColor as Integer, inDepth as Integer, inColorDev as Boolean) As Color
declare function GetThemeTextColor lib "Carbon" (inColor as Integer, inDepth as Integer, inColorDev as Boolean, outColor as Ptr) as Integer

dim i as Integer
dim col as MemoryBlock

col = newMemoryBlock(6)

i = GetThemeTextColor(inColor, inDepth, inColorDev, col)

return RGB(col.UShort(0)\256, col.UShort(2)\256, col.UShort(4)\256)
End Function
```

Notes: The color for this is:

const kThemeTextColorDialogInactive = 2.

c = GetThemeTextColor(kThemeTextColorDialogInactive, Screen(0).Depth, true)

For Mac OS X you should use "CarbonLib" instead of "AppearanceLib" ...

8.0.127 How to get the current free stack space?

Plugin Version: all, Platform: macOS.

Answer: You can something like the code below:
Example:

```
Sub ShowStackSize()
dim threadid as Integer
dim size as Integer

declare function GetCurrentThread lib "Carbon" (byref threadid as Integer) as short
declare function ThreadCurrentStackSpace lib "Carbon" (threadid as Integer, byref size as Integer) as short

if GetCurrentThread(threadid)=0 then
if 0=ThreadCurrentStackSpace(threadid,size) then
MsgBox str(size)
end if
end if
```

```
End Sub
```

Notes: For Mac OS 9, use ”ThreadLib” instead of ”CarbonLib”. You can use #if if you like for that.

8.0.128 How to get the current timezone?

Plugin Version: all, Platforms: macOS, Windows.

Answer:

You can use the TimeZoneMBS class or the CFTimeZoneMBS class.
Or code like below:
Example:

```
Function GMTOffsetInMinutes() as Integer
// Returns the offset of the current time to GMT in minutes.
// supports Mac OS and Windows, but not Linux yet (let me know if
// you have code for that, please)
//
// Note that the offset is not always an even multiple of 60, but
// there are also half hour offsets, even one 5:45h offset

// This version by Thomas Tempelmann (rb@tempel.org) on 25 Nov 2005
// with a fix that should also make it work with future Intel Mac targets.
//
// Using code from various authors found on the RB NUG mailing list

dim result, bias, dayLightbias as Integer
dim info as memoryBlock
dim offset as Integer

#if targetMacOS then

Declare Sub ReadLocation lib "Carbon" (location As ptr)

info = NewMemoryBlock(12)
ReadLocation info
if false then
// bad, because it does not work on Intel Macs:
'offset = info.short(9) * 256 + info.byte(11)
else
offset = BitwiseAnd (info.long(8), &hFFFFFF)
end

offset = info.short(9) * 256 + info.byte(11)
offset = offset \60
return offset

#endif

#if targetWin32 then

Declare Function GetTimeZoneInformation Lib "Kernel32" ( tzInfoPointer as Ptr ) as Integer
// returns one of
// TIME_ZONE_ID_UNKNOWN 0
// – Note: e.g. New Delhi (GMT+5:30) and Newfoundland (-3:30) return this value 0
// TIME_ZONE_ID_STANDARD 1
// TIME_ZONE_ID_DAYLIGHT 2

info = new MemoryBlock(172)
result = GetTimeZoneInformation(info)

bias = info.Long(0)
// note: the original code I found in the NUG archives used Long(84) and switched to Long(0)
// only for result=1 and result=2, but my tests found that Long(0) is also the right value for result=0

if result = 2 then
daylightBias = info.long(168)
end if
offset = - (bias + dayLightbias)
return offset

#endif

End Function
```

8.0.129 How to get the current window title?

Plugin Version: all, Platform: macOS.

Answer: The code below returns the current window title for the frontmost window on Mac OS X if Accessibilty services are
Example:

```
Function CurrentWindowTitle() As string
// your application needs permissions for accessibility to make this work!

dim SystemWideElement,FocusedApplicationElement,FocusedWindowElement as AXUIElementMBS
dim FocusedApplication,FocusedWindow,Title as AXValueMBS
dim s as String
dim cs as CFStringMBS
```

```
SystemWideElement=AccessibilityMBS.SystemWideAXUIElement
if SystemWideElement<>nil then
FocusedApplication=SystemWideElement.AttributeValue(AccessibilityMBS.kAXFocusedApplicationAttribute)
if FocusedApplication.Type=AccessibilityMBS.kAXUIElementMBSTypeID then
FocusedApplicationElement=new AXUIElementMBS
FocusedApplicationElement.Handle=FocusedApplication.Handle
FocusedApplicationElement.RetainObject

FocusedWindow=FocusedApplicationElement.AttributeValue(AccessibilityMBS.kAXFocusedWindowAttribute)

if FocusedWindow<>nil and AccessibilityMBS.kAXUIElementMBSTypeID=FocusedWindow.Type then

FocusedWindowElement=new AXUIElementMBS
FocusedWindowElement.Handle=FocusedWindow.Handle
FocusedWindowElement.RetainObject

Title=FocusedWindowElement.AttributeValue(AccessibilityMBS.kAXTitleAttribute)
if Title<>nil and Title.Type=kCFStringMBSTypeID then
cs=new CFStringMBS
cs.handle=Title.Handle
cs.RetainObject
Return cs.str
end if
end if
end if
end if
End Function
```

8.0.130 How to get the cursor blink interval time?

Plugin Version: all, Platform: macOS.

Answer: On Mac OS you can use GetCaretTime from the toolbox.
Example:

```
declare function GetCaretTime lib "Carbon" () as Integer

MsgBox str(GetCaretTime())+" ticks"
```

Notes: 60 ticks make one second.

8.0.131 How to get the list of the current selected files in the Finder?

Plugin Version: all, Platform: macOS.

Answer:

Use the AppleScript like this one:

```
tell application ”finder”
return selection
end tell
```

Which translates into this AppleEvent:

```
Process(”Finder”).SendAE ”core,getd,’—-’:obj { form:prop, want:type(prop), seld:type(sele), from:’null’() }
”
```

and as Xojo code it looks like this:
Example:

```
dim ae as appleevent
dim o1 as appleeventObjectSpecifier
dim f as folderItem
dim aList as appleeventdescList
dim i as Integer
dim dateiname as string

// setup the AppleEvent
o1=getpropertyObjectDescriptor( nil, ”sele”)
ae= newappleEvent(”core”, ”getd”, ”MACS”)
ae.objectSpecifierParam(”—-”)=o1

// send it
if ae.send then
// got the list
alist=ae.replyDescList

// now show the list of filename into an editfield:

for i=1 to alist.count
f=alist.folderItemItem(i)

dateiname=f.name
// editfield1 with property ”mulitline=true”!
editfield1.text=editfield1.text + dateiname + chr(13)
next
```

```
end if
```

8.0.132 How to get the Mac OS system version?

Plugin Version: all, Platform: macOS.

Answer: The following code queries the value and displays the version number:
Example:

```
dim first as Integer
dim second as Integer
dim third as Integer
dim l as Integer

if System.Gestalt(”sysv”,l) then

Third=Bitwiseand(l,15)
second=Bitwiseand(l\16,15)
first=Bitwiseand(l\256,15)+10*Bitwiseand(l\256\16,15)
end if

if First>=10 then
msgbox ”Mac OS X ”+str(First)+”.”+str(Second)+”.”+str(third)
else
msgbox ”Mac OS ”+str(First)+”.”+str(Second)+”.”+str(third)
end if
```

8.0.133 How to get the Mac OS Version using System.Gestalt?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
Dim s As String
Dim b As Boolean
Dim i, resp as Integer

// Systemversion
b = System.Gestalt(”sysv”, resp)
If b then
s = Hex(resp)
```

```
For i =Len(s)-1 DownTo 1
s=Left(s,i)+"."+Mid(s,i+1)
Next
MsgBox "Systemversion: Mac OS " + s
end if
```

Notes: The MBS Plugin has a SystemInformationMBS.OSVersionString function for this.

8.0.134 How to get the screensize excluding the task bar?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Notes: Use the Screen class with the available* properties.

8.0.135 How to get the size of the frontmost window on Windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Notes: Make yourself a class for the WindowRect with four properties:

```
Bottom as Integer
Left as Integer
Right as Integer
Top as Integer
```

Add the following method to your class:

```
Sub GetWindowRect(windowhandle as Integer)
dim err as Integer
dim mem as memoryBlock
#if targetwin32 then
Declare Function GetWindowRect Lib "user32.dll" (hwnd as Integer, ipRect As Ptr) as Integer

mem = newmemoryBlock(16)
err = GetWindowRect(windowhandle, mem)
Left = mem.long(0)
Top = mem.Long(4)
Right = mem.Long(8)
Bottom = mem.Long(12)
```

```
#endif
End Sub
```

Good to use for the MDI Master Window!

8.0.136 How to get the source code of a HTMLViewer?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
// for Windows:

msgbox HTMLViewer1.IEHTMLTextMBS

// for MacOS with WebKit 2.x:

msgbox HTMLViewer1.WKWebViewMBS.HTMLText
```

8.0.137 How to get Xojo apps running Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You need to install some requuire packages.
Notes: You need CUPS as well as GTK packages. On 64 bit systems also the ia32-libs package.

Please note that you need a x86 compatible Linux. So no PPC, Power, ARM or other CPUs.

8.0.138 How to handle really huge images with GraphicsMagick or ImageMagick?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Sometimes it may be better to use an extra application to process images.
Notes: A typical 32 bit app made with Xojo can use around 1.8 GB on Windows and 3 GB on Mac OS X. Some images may be huge, so that processing them causes several copies of the image to be in memory. With a 500 MB image in memory, doing a scale or rotation may require a temp image. So with source, temp and dest images with each 500 MB plus your normal app memory usage, you may hit the limit of Windows with 1.8 GB.

In that case it may be worth running a tool like gm in the shell class. gm is the command line version of GraphicsMagick. There you can run the 64 bit version which is not limited in memory like your own application. Also you can monitor progress and keep your app responsive.

8.0.139 How to handle tab key for editable cells in listbox?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this function:
Example:

```
Function HandleTabInList(list as listbox, row as Integer, column as Integer, key as String) As Boolean
// Handle tab character in Listbox.CellKeyDown event

Select case asc(key)
case 9
if Keyboard.AsyncShiftKey then
// back

// look for column left
for i as Integer = column-1 downto 0
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next

// not found, so look in row before
row = row - 1
if row >= 0 then
for i as Integer = list.ColumnCount-1 downto 0
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next
end if
else
// forward

// look for column right
for i as Integer = column+1 to list.ColumnCount-1
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next
```

```
// not found, so look in row below
row = row + 1
if row <list.ListCount then
for i as Integer = 0 to list.ColumnCount-1
if list.ColumnType(i) >= list.TypeEditable then
list.EditCell(row, i)
Return true
end if
next
end if
end if
end Select
End Function
```

Notes: You call it from CellKeyDown event like this:

```
EventHandler Function CellKeyDown(row as Integer, column as Integer, key as String) As Boolean
if HandleTabInList(me, row, column, key) then Return true
End EventHandler
```

As you see in the code, we handle tab and shift + tab for moving back and forward. Also we wrap to previous/next row if needed. Feel free to extend this to wrap from last to first row or create a new row for editing.

8.0.140 How to hard link MapKit framework?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Our MapKit classes weak link the framework. If you need hard linking it for the App Store, you can add this method to a class:
Example:

```
Sub ReferenceMapKit()
// just put this in window or app class

#if TargetMachO and Target64Bit then
Declare sub testing Lib "MapKit" Selector "test" (id as ptr)
testing(nil)
#endif

End Sub
```

Notes: No need to call the method.
Just having it in a window or app, will cause the compiler to hard link the framework.

8.0.141 How to have a PDF downloaded to the user in a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use a WebHTMLViewer control and load the PDF file with the PDF plugin from the browser.
Example:

```
dim CurrentFile as WebFile // a property of the WebPage

// define the PDF file
CurrentFile = new WebFile
CurrentFile.Filename = "test.pdf"
CurrentFile.MIMEType = "application/pdf"
CurrentFile.Data = "some pdf data" // MyDynaPDF.GetBuffer
CurrentFile.ForceDownload = true

// start the download
showurl(CurrentFile.url)
```

Notes: See our Create PDF example for the Xojo Web Edition.

8.0.142 How to hide all applications except mine?

Platform: macOS.

Answer: The code below will on Mac OS hide all applications except your one:
Example:

```
dim p as new ProcessMBS

p.GetFirstProcess
do
if not p.FrontProcess then
p.Visible=false
end if
loop until not p.GetNextProcess
```

8.0.143 How to hide script errors in HTMLViewer on Windows?

Plugin Version: all, Platform: Windows.

Answer: Set Internet Explorer to silent mode with code like this:
Example:

```
htmlviewer1._ole.Content.value("Silent") = True
```

Notes: Simply put this code in the open event of your htmlviewer control (using me instead of htmlviewer1).

8.0.144 How to hide the grid/background/border in ChartDirector?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: If you want to hide something in a chart, simply assign the kTransparent constant as color.

8.0.145 How to hide the mouse cursor on Mac?

Plugin Version: all, Platform: macOS.

Answer: Try this declare:
Example:

```
Declare Sub HideCursor Lib "Carbon" () Inline68K("A852")

HideCursor
```

Notes: The MBS Plugin has this function and supports it on Windows, too.

8.0.146 How to insert image to NSTextView or TextArea?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With NSTextViewMBS you can use this code to insert file:
Example:

```
// insert a file to textview

Public Sub InsertFile(textview as NSTextViewMBS, f as FolderItem)
// read to file
```

```
dim b as BinaryStream = BinaryStream.Open(f)
dim s as string = b.Read(b.Length)

// build wrapper
dim fileWrapper as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(s)
fileWrapper.preferredFilename = f.name

// make attachment
dim fileAttachment as new NSTextAttachmentMBS(fileWrapper)
dim attributedString as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttachment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub
```

Notes: For TextArea you can query the underlaying NSTextViewMBS object via TextArea.NSTextViewMBS method.

8.0.147 How to jump to an anchor in a htmlviewer?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You can use javascript to change the current window's location.
Example:

```
// load website
htmlviewer1.LoadURL "http://www.monkeybreadsoftware.net/addressbook-abpersonmbs.shtml"

// later jump to anchor named "16":

if TargetWin32 then
call HTMLViewer1.IERunJavaScriptMBS "window.location = ""#16"""
end if
```

8.0.148 How to keep a movieplayer unclickable?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: To keep the user away from clicking on a playing Movie you can just drop a Canvas in front of the Movieplayer and take the clicks there.
Example:

```
Function Canvas1.MouseDown(X as Integer, Y as Integer) as boolean
return true // take it and do nothing
End Function
```

8.0.149 How to keep my web app from using 100% CPU time?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Linux and MacOS you can use renice command in the terminal. On Windows use the task manager to reduce priority.
Notes: If you launch your app with nohup on Linux or Mac OS X like this from the terminal or a script:

```
nohup /webapps/MyApp/MyApp &
```

you can simply have a second line saying this:

```
renice 20 $ !
```

which tells the system to lower priority to lowest value for the latest background process.

8.0.150 How to kill a process by name?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can kill a process (or application) by name if you loop over all the processes and kill the one you need.
Example:

```
dim p as new ProcessMBS
p.GetfirstProcess ' get first
do
if p.name = "TextEdit" then
call p.KillProcess
Return
end if
loop until not p.GetNextProcess
```

Notes: You may want to check the result of killProcess function. Not every user is allowed to kill every application.

8.0.151 How to know how many CPUs are present?

Plugin Version: all, Platform: macOS.

Answer: Try this function:
Example:

```
Function GetCPUCount() as Integer
Declare Function MPProcessors Lib "Carbon" () as Integer

Return MPProcessors()
End Function
```

Notes: Your app will than need that library to launch on Classic. To avoid this the MBS plugin checks if this library is available and return 1 if it's not available.

8.0.152 How to know the calling function?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: On Mac you can use a helper function like this this code:
Example:

```
Public Function CallingFunction() as string
// Query name of calling function of a function

#Pragma BreakOnExceptions false

try

// raise a dummy exception
dim r as new NilObjectException
raise r

catch x as NilObjectException

// get stack
dim stack() as string = x.Stack

// pick function name and return
dim name as string = stack(2)
Return name

end try
End Function
```

Notes: You need to include function names in your application.

8.0.153 How to launch an app using it's creator code?

Plugin Version: all, Platform: macOS.

Answer: Send an AppleEvent "oapp" with the creator code to the Finder ("MACS"):
Example:

```
Dim a as AppleEvent
dim creator as string

creator = "MSIE" ' here the Internet Explorer

a = NewAppleEvent("aevt", "odoc", "MACS")
a.Timeout = -1

a.ObjectSpecifierParam("—-") = GetUniqueIDObjectDescriptor("appf", nil, creator)

if not a.send then
msgBox "An error has occured"
else

end if
```

8.0.154 How to launch disc utility?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use this code:
Example:

```
dim f as FolderItem = LaunchServicesFindApplicationForInfoMBS("","com.apple.DiskUtility","")

if f<>Nil then
f.Launch
end if
```

Notes: This works even if people renamed the disc utility or moved it to another folder.

8.0.155 How to make a lot of changes to a REAL SQL Database faster?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You may try to embed your changes to the database between two transaction calls.
Example:

```
dim db as Database // some database

db.SQLExecute "BEGIN TRANSACTION"
// Do some Stuff
db.SQLExecute "END TRANSACTION"
```

Notes: This can increase speed by some factors.

8.0.156 How to make a NSImage object for my retina enabled app?

Plugin Version: all, Platform: macOS.

Answer: You can use code like this:
Example:

```
Function NewRetinaImage(pic as Picture, mask as Picture = nil) As NSImageMBS
// first make a NSImageMBS from it
dim n as new NSImageMBS(pic, mask)

// now set to half the size, so we have 2x pixels for the image
n.size = new NSSizeMBS(n.width/2, n.height/2)

// and return
Return n
End Function
```

Notes: The thing to do is to have 2x the pixels, but assign a size to the image which gives it the right size in points.
You can pass the NSImageMBS from here to NSMenuItemMBS. For Retina displays, the full resolution is used. For others it will be reduced.

8.0.157 How to make a window borderless on Windows?

Plugin Version: all, Platform: Windows.

Answer: Try this declares:
Example:

```
// Sets window to borderless popup type, and sets its initial dimensions.
// Call this method, then Win32SetBorderlessPos, and then RB's Show
// method. Use RB Frame type 7 (Global Floating Window).

Const SWP_NOMOVE = &H2
Const SWP_FRAMECHANGED = &H20
Const HWND_TOPMOST = -1
Const GWL_STYLE = -16
Const WS_POPUPWINDOW = &H80880000

Dim styleFlags as Integer

#If TargetWin32 Then

Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA" (hwnd as Integer, nIndex as Integer, dwNewLong as Integer) as Integer
Declare Function SetWindowPos Lib "user32" (hwnd as Integer, hWndInstertAfter as Integer, x as Integer, y as Integer, cx as Integer, cy as Integer, flags as Integer) as Integer

styleFlags = SetWindowLong( w.WinHWND, GWL_STYLE, WS_POPUPWINDOW )
styleFlags = BitwiseOr( SWP_FRAMECHANGED, SWP_NOMOVE )
styleFlags = SetWindowPos( w.WinHWND, HWND_TOPMOST, 0, 0, wd, ht, styleFlags )

#EndIf
```

8.0.158 How to make an alias using AppleEvents?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
Sub MakeAlias(folder as folderitem, target as folderitem, aliasname as string)
dim ev as AppleEvent
dim myResult as boolean
dim properties as AppleEventRecord

ev = NewAppleEvent("core","crel","MACS")
ev.MacTypeParam("kocl") = "alis"
ev.FolderItemParam("to ") = target
ev.FolderItemParam("insh") = folder

properties=new AppleEventRecord
```

```
properties.StringParam("pnam")=aliasname

ev.RecordParam("prdt")=properties

myResult = ev.send
// true on success, false on error
End Sub
```

Notes: Call it like this:
MakeAlias SpecialFolder.Desktop, SpecialFolder.Desktop.Child("Gif Copy.rb"), "test.rb alias"

Seems to not work on Mac OS X 10.6

8.0.159 How to make AppleScripts much faster?

Plugin Version: all, Platform: macOS.

Answer: use "ignoring application responses" like in this example:
Notes: on run { fn,fpx,fpy }
ignoring application responses
tell app "Finder" to set the position of folder fn to fpx,fpy
end ignoring
end run

8.0.160 How to make double clicks on a canvas?

Plugin Version: all, Platform: macOS.

Answer:

Update: Newer Xojo versions support DoubleClick event, so you don't need this code.

Here's my tip from the tips list on how to add a double-click event to the Canvas control. The technique could easily be used for a window or any Rectcontrol:

Because of its built-in drawing methods, the Canvas control is often used to create custom interface controls. But while the Canvas control has event handlers for most mouse events, it doesn't have an event handler for DoubleClick events. Fortunately, you can add a double-click event handler to a Canvas control easily. Basically, you're going to create a new class based on Canvas and add a double-click event to that. You can then use the new class anytime you need a Canvas with a double-click event.

To create a new Canvas class with a DoubleClick event handler, do this:

1. Add a new class to your project.
2. Set the Super property of the new class to ”Canvas”.
3. Change the name of this new class to ”DoubleClickCanvas”.

A double-click occurs when two clicks occur within the users double-click time (set in the Mouse control panel on both Macintosh and Windows) and within five pixels of each other. So, you’ll need a few properties to store when and where the last click occurred.

4. Add a new property with the following declaration and mark it as private: lastClickTicks as Integer
5. Add a new property with the following declaration and mark it as private: lastClickX as Integer
6. Add a new property with the following declaration and mark it as private: lastClickY as Integer

Since the Canvas control doesn’t have a DoubleClick event, you will need to add one.

7. Add a new event to your class by choosing New Event from the Edit menu and enter ”DoubleClick” as the event name.

Double-clicks occur on MouseUp. In order for the mouseUp event to fire, you must return True in the MouseDown event.

8. In the MouseDown event, add the following code:

```
Return True
```

In the MouseUp event, you will need to determine what the users double-click time is. This value is represented on both the Mac and Windows in ticks. A tick is 1/60th of a second. Since there isn’t a built-in function for this, you’ll need to make a toolbox call. The mouseUp event code below makes the appropriate toolbox call for both Macintosh and Windows. It then compares the time of the users last click to the time of the current click and compares the location of the users last click to the location of the current click.

9. Add the following code to the MouseUp event:

```
dim doubleClickTime, currentClickTicks as Integer

#if targetMacOS then
Declare Function GetDblTime Lib ”Carbon” () as Integer
doubleClickTime = GetDblTime()
#endif

#if targetWin32 then
Declare Function GetDoubleClickTime Lib ”User32.DLL” () as Integer
```

```
doubleClickTime = GetDoubleClickTime()/60 // convert to ticks from milliseconds
#endif

currentClickTicks = ticks
//if the two clicks happened close enough together in time
if (currentClickTicks - lastClickTicks) <= doubleClickTime then
//if the two clicks occured close enough together in space
if abs(X - lastClickX) <= 5 and abs(Y - LastClickY) <= 5 then
DoubleClick //a double click has occured so call the event
end if
end if
lastClickTicks = currentClickTicks
lastClickX = X
lastClickY = Y
```

10. Now to test out your new DoubleClickCanvas, drag the class from the Project window to a window in your project to create an instance of it.

11. Double-click on the canvas you just added to your window to open the Code Editor. Notice that the canvas has a DoubleClick event handler. In this event handler, add the following code:
BEEP

8.0.161 How to make my Mac not sleeping?

Plugin Version: all, Platform: macOS.

Answer: Just inform the Mac OS about some system activity with code like this:
Example:

```
Sub UpdateSystemActivity()

#if TargetCarbon
declare function myUpdateSystemActivity lib "Carbon" alias "UpdateSystemActivity" (activity as Integer)
as short

const OverallAct = 0 // Delays idle sleep by small amount */
const UsrActivity = 1 // Delays idle sleep and dimming by timeout time */
const NetActivity = 2 // Delays idle sleep and power cycling by small amount */
const HDActivity = 3 // Delays hard drive spindown and idle sleep by small amount */
const IdleActivity = 4 // Delays idle sleep by timeout time */

dim e as Integer

e=myUpdateSystemActivity(UsrActivity)
```

```
// you may react on an error if e is not 0 after the call.

#endif
End Sub
```

Notes: You may use another constant if you prefer some different behavior.
Call it maybe every second.

8.0.162 How to make my own registration code scheme?

Plugin Version: all, Platform: Windows.

Answer: There are excellent articles about how to make a registratin code scheme, but you can also simply use our RegistrationEngineMBS class.
Notes: If you need a license text, why not use the one from Xojo as a starting point?

8.0.163 How to make small controls on Mac OS X?

Plugin Version: all, Platform: macOS.

Answer: You can try this code on Mac OS X:
Example:

```
'/*
'* Use the control's default drawing variant. This does not apply to
'* Scroll Bars, for which Normal is Large.
'*/
const kControlSizeNormal = 0

'/*
'* Use the control's small drawing variant. Currently supported by
'* the Check Box, Combo Box, Radio Button, Scroll Bar, Slider and Tab
'* controls.
'*/
const kControlSizeSmall = 1

'/*
'* Use the control's small drawing variant. Currently supported by
'* the Indeterminate Progress Bar, Progress Bar and Round Button
'* controls.
'*/
const kControlSizeLarge = 2
```

```
'/*
'* Control drawing variant determined by the control's bounds. This
'* ControlSize is only available with Scroll Bars to support their
'* legacy behavior of drawing differently within different bounds.
'*/
const kControlSizeAuto = &hFFFF

const kControlSizeTag = "size"

declare function SetControlData lib "Carbon" (controlhandle as Integer, part as short, tagname as OSType, size as Integer, data as ptr) as short

dim m as MemoryBlock

m=NewMemoryBlock(2)
m.UShort(0)=kControlSizeSmall

Title=str(SetControlData(CheckBox1.Handle, 0, kControlSizeTag, 2, m))
```

8.0.164 How to mark my Mac app as background only?

Plugin Version: all, Platform: macOS.

Answer: You can run a build script on each build with this code:
Example:

```
Dim App As String = CurrentBuildLocation + "/" + CurrentBuildAppName + ".app"
Call DoShellCommand("/usr/bin/defaults write " + App + "/Contents/Info ""NSUIElement"" YES")
```

Notes: This will set the NSUIElement flag to YES.

8.0.165 How to move a file or folder to trash?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like below:
Example:

```
Function MoveToTrash(f as FolderItem) As Boolean
#if TargetMacOS then
dim r as FolderItem
dim e as Integer = MacFileOperationMBS.MoveObjectToTrashSync(f, r, MacFileOperationMBS.kFSFileOperationDefaultOptions)
```

```
if e = 0 then
Return true // Ok
end if

#elseif TargetWin32 then
dim w as new WindowsFileCopyMBS

dim flags as Integer = w.FileOperationAllowUndo + w.FileOperationNoErrorUI + w.FileOperationSilent
+ w.FileOperationNoConfirmation
if w.FileOperationDelete(f, flags) then
Return true // OK
end if

flags = w.FileOperationNoErrorUI + w.FileOperationSilent + w.FileOperationNoConfirmation
if w.FileOperationDelete(f, flags) then
Return true // OK
end if
#else
// Target not supported
break
Return false
#endif
End Function
```

Notes: If you want to move a file to trash, you could use f.movefileto f.trashfolder, but that will overwrite existing files in the trash. You can use our MacFileOperationMBS class to move a file on Mac to the trash. And it uses the same code as the Finder, so files are renamed when the same name is already in use in the trash:

On Windows we use WindowsFileCopyMBS class.
Requires Mac OS X 10.5.

8.0.166 How to move an application to the front using the creator code?

Plugin Version: all, Platform: macOS.

Answer: This makes SimpleText (Code ttxt) to the frontmost application:
Example:

```
dim a as appleevent

a=newappleEvent(”misc”,”actv”,”ttxt”)
```

```
if a.send then
end if
```

Notes: (Code is Mac only)

8.0.167 How to move file with ftp and curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can set post/pre quotes to have ftp commands executed before or after the download/upload.
Example:

```
dim d as CURLMBS // your curl object

// rename/move file
dim ws() As String
ws.Append "RNFR Temp.txt"
ws.append "RNTO MyFile.txt"

d.SetOptionPostQuote(ws)
```

Notes: Use SetOptionPostQuote, SetOptionPreQuote or SetOptionQuote.
The ftp commands you pass here are native ftp commands and not the commands you use with ftp applications. So rename is two commands. First RNFR to tell where to rename from and second RNTO with the new file name. To delete use DELE and the file path.

8.0.168 How to normalize string on Mac?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like below:
Example:

```
Function Normalize(t as string) As string
const kCFStringNormalizationFormD = 0 // Canonical Decomposition
const kCFStringNormalizationFormKD = 1 // Compatibility Decomposition
const kCFStringNormalizationFormC = 2 // Canonical Decomposition followed by Canonical Composition
const kCFStringNormalizationFormKC = 3 // Compatibility Decomposition followed by Canonical Composition

dim s as CFStringMBS = NewCFStringMBS(t)
dim m as CFMutableStringMBS = s.Normalize(kCFStringNormalizationFormD)
```

```
Return m.str
End Function
```

Notes: This uses Apple's CFString functions to normalize unicode variants.

8.0.169 How to obscure the mouse cursor on Mac?

Plugin Version: all, Platform: macOS.

Answer: Try this declare:
Example:

```
Declare Sub ObscureCursor Lib "Carbon" ()

ObscureCursor
```

Notes: The MBS Plugin has this function, but it's not supported for Windows.

8.0.170 How to open icon file on Mac?

Plugin Version: all, Platform: macOS.

Answer: Use the NSImageMBS class like this:
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.ico")
dim n as new NSImageMBS(f)

window1.Backdrop = n.CopyPictureWithMask
```

8.0.171 How to open PDF in acrobat reader?

Plugin Version: all, Platform: macOS.

Answer: Try this code:
Example:

```
dim pdf as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
```

```
// open PDF in Acrobat Reader on Mac:

// find app
dim bundleID as string = "com.adobe.Reader"
dim app as FolderItem = LaunchServicesFindApplicationForInfoMBS("", bundleID, "")

if app<>nil then

// launch app with parameters

dim docs() as FolderItem
docs.Append pdf

dim param as new LaunchServicesLaunchParameterMBS
param.Defaults = true
param.Application = app

dim x as FolderItem = LaunchServicesOpenXMBS(docs, param)

// on failure, simply launch it
if x = nil then
pdf.Launch(true)
end if

else
pdf.Launch(true)
end if
```

Notes: On Windows, simply use pdf.launch or WindowsShellExecuteMBS.

8.0.172 How to open printer preferences on Mac?

Plugin Version: all, Platform: macOS.

Answer: You can use our OpenMacOSXPreferencesPaneMBS function like this:
Example:

```
dim e as Integer = OpenMacOSXPreferencesPaneMBS("PrintAndFax")
if 0 = e then
MsgBox "OK"
elseif e = -43 then
MsgBox "File not found."
else
MsgBox "Error: "+str(e)
end if
```

8.0.173 How to open special characters panel on Mac?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We have functions for that in Cocoa and Carbon.
Example:

```
dim a as new NSApplicationMBS
a.orderFrontCharacterPalette
```

Notes: For Cocoa, you can use orderFrontCharacterPalette method in NSApplicationMBS class.

Or simply for Carbon and Cocoa the ShowCharacterPaletteMBS method.

8.0.174 How to optimize picture loading in Web Edition?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the WebPicture class.
Notes: Take your picture and create a WebPicture object. Store this WebPicture in a property of the WebPage, Session or app (as global as possible). On the first time you use this picture on an user session, the browser will load it. Second time you use it, the browser will most likely pick it from the cache.
Having pictures in App or some module reuses the same picture for all sessions which reduces memory footprint.
This does not work well with pictures you change very often or use only for one webpage on one user.

If you like to see an example, check our Map example.

8.0.175 How to parse XML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:
Example:

```
dim s as string = "<test><test /></test>"

try
```

```
dim x as new XmlDocument(s)
MsgBox "OK"
catch xe as XmlException
MsgBox "invalid XML"
end try
```

Notes: If you got an exception, you have a parse error.

8.0.176 How to play audio in a web app?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use the HTML5 audio tag and control it with javscript.
Notes: This is just another example app I made today. It plays a christmas song. The audio file is provided by the application to the server, so no external web server is needed and this application can run stand alone. To compile and run you need Xojo 2010r5.

In the open event we search the audio files and open them as binarystreams. We create the two webfile objects. Those webfiles are part of the app class, so we have them globally. There we set the data with the content of our streams. We also define file names and mime types. They are needed so browser know what we have here:

```
audioFileM4V = new WebFile
audioFileM4V.Data = bM.Read(BM.Length)
audioFileM4V.Filename = "music.m4a"
audioFileM4V.MIMEType = "audio/m4a"
```

```
audioFileOGG = new WebFile
audioFileOGG.Data = bO.Read(BO.Length)
audioFileOGG.Filename = "music.ogg"
audioFileOGG.MIMEType = "audio/ogg"
```

Next in the open event of the webpage we have a PageSource control. The location is set to be before content. In the open event we define the html code for this. First we pick the URLs for the audio files. Than we build the html to use the audio tag. As you see, we give it an ID for later use and have it preload automatically. If you add an autoplay tag, you can have the audio play right away. Inside the audio tag we have two sources so we provide audio for both Firefox (OGG) and Safari (MPEG4). Finally we have a text to display if HTML5 audio tag is not supported.

You can set the source in the EditSource event:

```
dim urlo as string = app.audioFileOGG.URL
dim urlm as string = app.audioFileM4V.URL
me.Source = "<audio id=""mymusic"" preload=""auto""><source src="""+urlo+""" type=""audio/ogg"" /><source src="""+urlm+""" type=""audio/mpeg"" />Your browser does not support the audio element.</audio>"
```

Next in the Play button we execute code to play the audio. This is a short javascript code which searches in the html document for the element with the ID "mymusic" which is the ID of our audio tag above. Once we got the object, we call it's play method to start playback.

```
me.ExecuteJavaScript("document.getElementById('mymusic').play();")
```

same for pause:

```
me.ExecuteJavaScript("document.getElementById('mymusic').pause();")
```

and finally for changing volume:

```
me.ExecuteJavaScript("document.getElementById('mymusic').volume="+str(me.Value/100.0)+";")
```

8.0.177 How to pretty print xml?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the XML Transform method with the right XLS.
Notes: Learn more here:
http://docs.xojo.com/index.php/XMLDocument.Transform

8.0.178 How to print to PDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: This code below shows how to redirect printing to a PDF file on Mac OS X.
Example:

```
// get Xojo printer setup
dim p as new PrinterSetup

// now put it into NSPrintInfo to manipulate
dim n as new NSPrintInfoMBS
n.SetupString = p.SetupString
```

```
// change destination to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
n.SetSaveDestination(f)

// move back
p.SetupString = n.SetupString

// and print as usual
dim g as Graphics = OpenPrinter(p)
g.DrawString "Hello World", 20, 20
```

Notes: And you can use normal graphics class for that.

8.0.179 How to query Spotlight's Last Open Date for a file?

Plugin Version: all, Platform: macOS.

Answer: You can use a MDItemMBS objec to query this value:
Example:

```
Function LastOpenedDate(Extends F As FolderItem, DefaultOtherDates As Boolean = True) As Date
#If TargetMacOS Then
Dim xMDItem as New MDItemMBS(F)
Dim xDate as Variant

If xMDItem <>Nil Then
xDate = xMDItem.GetAttribute(xMDItem.kMDItemLastUsedDate).DateValue
If xDate IsA Date Then Return xDate
Else
If xDate <>Nil Then Break
End If
#EndIf

If DefaultOtherDates Then
If F.ModificationDate <>Nil Then Return F.ModificationDate
If F.CreationDate <>Nil Then Return F.CreationDate
End If
End Function
```

Notes: Thanks for Josh Hoggan for this example code.

8.0.180 How to quit windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
#if targetwin32 then
dim i1,i2,r as Integer
declare function ExitWindowsEx lib "user32" (uFlags as Integer, dwReserved as Integer) as Integer
i1 = 2
i2 = 0
r = ExitWindowsEx(i1,i2)
if r<>0 then
' Error()
end if

#endif
```

Notes: uFlags parameters:

'4 = EWX_Force
'0 = EWX_Logoff
'2 = EWX_Reboot
'1 = EWX_shutdown, should shut down computer

Also check the ExitWindowsMBS method.

8.0.181 How to read a CSV file correctly?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With all the rules for quotes and delimiters, you can simply use the SplitCommaSeparatedValuesMBS method in our plugins like this:
Example:

```
dim f as FolderItem = SpecialFolder.Desktop.Child("test.csv")
dim t as TextInputStream = f.OpenAsTextFile

while not t.EOF
dim s as string = t.ReadLine(encodings.ASCII)

dim items() as string = SplitCommaSeparatedValuesMBS(s, ";", """")
```

```
List.AddRow ""
dim u as Integer = UBound(items)
for i as Integer = 0 to u
List.Cell(List.LastIndex,i) = items(i)
next

wend
```

Notes: Please make sure you choose the right text encoding.

8.0.182 How to read the command line on windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
#if targetwin32 then
dim line as string
Dim mem as MemoryBlock

Declare Function GetCommandLineA Lib "kernel32" () As Ptr

mem=GetCommandLineA()
s=mem.cstring(0)

#endif
```

Notes: Newer Xojo versions have a system.commandline property.

8.0.183 How to render PDF pages with PDF Kit?

Plugin Version: all, Platform: Windows.

Answer: Try this code:
Example:

```
// choose a file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")

// open it as PDF Document
dim sourceFile as New PDFDocumentMBS(f)
```

```
if sourceFile.handle <>0 then // it is a PDF file

// get upper bound of pages
dim c as Integer = sourceFile.pageCount-1

// from first to last page
for n as Integer = 0 to c

// pick that page
dim page as PDFPageMBS = sourceFile.pageAtIndex(n)

// render to image
dim p as NSImageMBS = page.Render

// and convert to RB picture and display
Backdrop = p.CopyPictureWithMask

next

end if
```

Notes: PDFKit works only on Mac OS X.

8.0.184 How to restart a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:
Example:

```
dim ae as appleevent
ae=newappleEvent("FNDR","rest","MACS")
if not ae.send then
msgBox "The computer couldn't be restarted."
end if
```

8.0.185 How to resume ftp upload with curl plugin?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: CURL supports that and you simply need to set the right options.

Notes: First of course OptionUpload must be true. Second OptionFTPAppend must be true so the OptionResumeFrom is used. Store there (or in OptionResumeFromLarge) your start value.
Don't forget to implement the read event and return data there as requested.

8.0.186 How to rotate a PDF page with CoreGraphics?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: This code opens a PDF and draws the first page into a new PDF with 90¬∞ rotation.
Example:

```
// Rotate a PDF page

// our files
dim sourcefile as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
dim destfile as FolderItem = SpecialFolder.Desktop.Child("rotated.pdf")

// open PDF
dim pdf as CGPDFDocumentMBS = sourcefile.OpenAsCGPDFDocumentMBS

// query media size of first page
dim r as CGRectMBS = pdf.MediaBox(1)

// create new PDF
dim c as CGContextMBS = destfile.NewCGPDFDocumentMBS(r,"title","Author","Creator")

// create rotated rectangle
dim nr as new CGRectMBS(0,0,r.Height,r.Width)

// create new page
c.BeginPage nr
c.SaveGState

const pi = 3.14159265

// rotate by 90¬∞
c.RotateCTM pi*1.5

// fix origin
c.TranslateCTM -r.width,0

// draw PDF
c.DrawCGPDFDocument pdf,r,1

// cleanup
c.RestoreGState
c.EndPage
```

```
c = nil

// show in PDF viewer
destfile.Launch
```

Notes: This code is Mac only as it needs CoreGraphics.

8.0.187 How to rotate image with CoreImage?

Plugin Version: all, Platform: macOS.

Answer: Use the code like the one below:
Example:

```
// Rotate image with CoreImage

// load image
dim f as FolderItem = SpecialFolder.Desktop.Child(”test.png”)
dim image as new CIImageMBS(f)

// rotate 45 degree
dim n as new NSAffineTransformMBS
n.rotateByDegrees(45)

dim TransformFilter as new CIFilterAffineTransformMBS
TransformFilter.inputImage = image
TransformFilter.inputTransform = n

// get result
dim resultImage as CIImageMBS = TransformFilter.outputImage

// for saving to file
dim outputImage as NSImageMBS = resultImage.RenderNSImage(false)

f = SpecialFolder.Desktop.Child(”output.png”)
dim b as BinaryStream = BinaryStream.Create(f, true)
b.Write outputImage.PNGRepresentation

// as Xojo picture object for display
dim pic as Picture = outputImage.CopyPictureWithMask

Backdrop = pic
```

8.0.188 How to run a 32 bit application on a 64 bit Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Install 32 bit compatibility libraries.
Notes: The package is called ia32-libs for ubuntu (and others).
Some applications need to be run on a 32 bit system as they need some hardware related libraries. Like libUSB or libHID for USB devices.

8.0.189 How to save HTMLViewer to PDF with landscape orientation?

Plugin Version: all, Platform: macOS.

Answer: You can use NSPrintInfoMBS to change the options for PrintToPDFFile function.
Example:

```
// make it landscape
dim n as NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
n.orientation = n.NSLandscapeOrientation

// save html to file
dim f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
call HTMLViewer1.PrintToPDFFileMBS(f,10,30,10,30)
```

Notes: You may want to reset options later.
This code is only for Mac OS X.

8.0.190 How to save RTFD?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: With NSTextViewMBS you can use this code to save to RTFD:
Example:

```
// save text as RTFD including image attachments
dim f as FolderItem = GetSaveFolderItem(FileTypes1.ApplicationRtfd, "test.rtfd")

if f = nil then Return

dim a as NSAttributedStringMBS = textView.textStorage
dim w as NSFileWrapperMBS = a.RTFDFileWrapperFromRange(0, a.length, DocumentAttributes)

dim e as NSErrorMBS
if w.writeToFile(f, e) then
```

```
else
MsgBox e.LocalizedDescription
end if
```

Notes: For TextArea you can query the underlaying NSTextViewMBS object via TextArea.NSTextViewMBS method.

8.0.191 How to save RTFD?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: How to load PDF to htmlviewer on desktop?
Example:

```
Public Sub LoadPDFData(viewer as HTMLViewer, PDFData as string)
Dim base64string As String = EncodeBase64(PDFData)

// remove line endings to make it a big line
base64string = ReplaceLineEndings(base64string, "")

// build data URL
// https://en.wikipedia.org/wiki/Data_URI_scheme
Dim dataURL As String = "data:application/pdf;base64," + base64string

// show in webviewer
HTMLViewer1.LoadURL(dataURL)

// may not work everywhere due to URL length limit
// for Web projects, use WebFile instead!
End Sub
```

Notes: This avoids a temporary file, which may also work.
For Web Apps, please use WebFile.

8.0.192 How to scale a picture proportionally with mask?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: For a proportional scaling, we calculate the new picture size relative to the target maximum size.
Example:

```
Function ProportinalScaledWithMask(extends pic as Picture, Width as Integer, Height as Integer) As Picture
// Calculate scale factor

dim faktor as Double = min( Height / Pic.Height, Width / Pic.Width)

// Calculate new size
dim w as Integer = Pic.Width * faktor
dim h as Integer = Pic.Height * faktor

// create new picture
dim NewPic as new Picture(w,h,32)

// check if we have a mask and clear it
dim m as picture = pic.mask(False)
pic.mask = nil

// draw picture in the new size
NewPic.Graphics.DrawPicture Pic, 0, 0, w, h, 0, 0, Pic.Width, Pic.Height

if m <>nil then
// restore mask and scale it
pic.mask = m
NewPic.mask.Graphics.DrawPicture m, 0, 0, w, h, 0, 0, Pic.Width, Pic.Height
end if


// return result
Return NewPic
End Function
```

Notes: This version handles mask. As you see we actually have to remove mask in order to copy the picture part correctly.

8.0.193 How to scale a picture proportionally?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: For a proportional scaling, we calculate the new picture size relative to the target maximum size.
Example:

```
Function ProportionalScaled(extends pic as Picture, Width as Integer, Height as Integer) As Picture
// Calculate scale factor

dim faktor as Double = min( Height / Pic.Height, Width / Pic.Width)
```

```
// Calculate new size
dim w as Integer = Pic.Width * faktor
dim h as Integer = Pic.Height * faktor

// create new picture
dim NewPic as new Picture(w,h,32)

// draw picture in the new size
NewPic.Graphics.DrawPicture Pic, 0, 0, w, h, 0, 0, Pic.Width, Pic.Height

// return result
Return NewPic
End Function
```

Notes: This does not handle mask, but you can scale the mask the same way and assign it to the new picture.
(see other FAQ entry with mask)

8.0.194 How to scale/resize a CIImageMBS?

Plugin Version: all, Platform: Windows.

Answer: Use the CIFilterLanczosScaleTransform filter to scale down a picture to a specific size.
Example:

```
Dim pic As Picture = LogoMBS(500)
Dim image As CIImageMBS = CIImageMBS.imageWithPicture(pic)

Dim filter As New CIFilterLanczosScaleTransformMBS

Const targetWidth = 600.0
Const targetHeight = 400.0

Dim scale As Double = targetHeight / image.Extent.Height
Dim aspect As Double = targetWidth / (image.Extent.Width * scale)

filter.inputImage = image
filter.inputScale = scale
filter.inputAspectRatio = aspect

Dim result As Picture = filter.outputImage.RenderPicture

Backdrop = result
```

Notes: This is same code as our scaleTo convenience method.

8.0.195 How to scale/resize a picture?

Plugin Version: all, Platform: Windows.

Answer: There are several ways to scale or resize a picture. The easiest way may be the ScaleMBS function in the Picture class.
Example:

```
dim Original,Scaled as Picture

Original=LogoMBS(500)
Scaled=Original.ScaleMBS(100,100,true)
```

Notes: The plugin ways:
- GraphicsMagick can scale/resize.
- CoreImage scale filter may result in the fastest and best images on Mac OS X 10.4.
- NSImageMBS can scale, but is Mac OS X only.
- CGImageMBS can scale, but is Mac OS X only.
- CIImageMBS can scale, but is Mac OS X only.
- QuickTime Graphics exporter and importer can be connected to scale. (this was used more often a few years ago)
- ImageMagick can scale very nice and crossplatform. But the ImageMagick libraries are big.
- The picture.ScaleMBS function is self written and results in equal output on Mac, Windows and Linux without any additional libraries installed.
- Picture.ScalingMBS does crossplatform scaling with several modes.

with pure Xojo:
- make a new picture and draw the old one with new size inside.

8.0.196 How to search with regex and use unicode codepoints?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can specify unicode characters in search string with backslash x and digits.
Example:

```
dim r as RegExMbs
dim s as string
dim c as Integer
```

```
s="123 √§√√º ABC 456"

r=new RegExMBS
if r.Compile(".√.") then
c=r.Execute(s,0)
MsgBox str(c)+" "+str(r.Offset(0))+" "+str(r.Offset(1))
// shows: 1 4 10
// 1 for ubound of the offset array
// 4 for 4 bytes before the matched pattern
// 10 for the 10 bytes before the end of the matched pattern
end if

r=new RegExMBS
if r.Compile(".\xF6.") then // finds √ using Unicode codepoint
c=r.Execute(s,0)
MsgBox str(c)+" "+str(r.Offset(0))+" "+str(r.Offset(1))
// shows: 1 4 10
// 1 for ubound of the offset array
// 4 for 4 bytes before the matched pattern
// 10 for the 10 bytes before the end of the matched pattern
end if
```

8.0.197 How to see if a file is invisible for Mac OS X?

Plugin Version: all, Platform: macOS.

Answer: Try this function:
Example:

```
Function Invisible(F As FolderItem) As Boolean
Dim TIS As TextInputStream
Dim S,All As String
Dim I as Integer
dim g as folderitem

If Left(F.Name,1)="." or not f.visible Then
Return True
End If

g=F.Parent.Child(".hidden")
If g.Exists Then
TIS=g.OpenAsTextFile
if tis<>Nil then
All=TIS.ReadAll
For I=1 to CountFields(All,Chr(11))
S=NthField(All, Chr(11), I)
```

```
If S=F.name Then
Return True
End If
Next
end if
End if
End Function
```

8.0.198 How to set cache size for SQLite or REALSQLDatabase?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You use the pragma cache_size command on the database.
Example:

```
// set cache size to 20000 pages which is about 20 MB for default page size
dim db as REALSQLDatabase
db.SQLExecute "PRAGMA cache_size = 20000"
```

Notes: Default cache size is 2000 pages which is not much.
You get best performance if whole database fits in memory.
At least you should try to have a cache big enough so you can do queries in memory.
You only need to call this pragma command once after you opened the database.

8.0.199 How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:
Example:

```
window1.ModifiedMBS=true
```

8.0.200 How to show a PDF file to the user in a Web Application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use a WebHTMLViewer control and load the
Example:

```
dim CurrentFile as WebFile // a property of the WebPage

// define the PDF file
CurrentFile = new WebFile
CurrentFile.Filename = "test.pdf"
CurrentFile.MIMEType = "application/pdf"
CurrentFile.Data = "some pdf data" // MyDynaPDF.GetBuffer

// load into html viewer
HTMLViewer1.URL = CurrentFile.URL
```

Notes: See our Create PDF example for the Xojo Web Edition.

8.0.201 How to show Keyboard Viewer programmatically?

Platform: macOS.

Answer: Use Xojo or AppleScript to launch the KeyboardViewerServer.app.
Example:

```
dim a as new AppleScriptMBS
dim text as string
dim lines(-1) as string

lines.append "set theApplication to ""KeyboardViewerServer"""
lines.append "set thePath to ""/System/Library/Components/KeyboardViewer.component/Contents/SharedSupport/KeyboardViewerServer.app"""
lines.append ""
lines.append "set POSIXPath to ((POSIX file thePath) as string)"
lines.append "tell application ""System Events"" to set isRunning to 0 <(count (application processes whose name is theApplication))"
lines.append "if isRunning then tell application POSIXPath to quit"
lines.append "delay 0.15"
lines.append ""
lines.append "ignoring application responses"
lines.append " tell application POSIXPath to run"
lines.append "end ignoring"

text=join(lines,EndOfLine.macintosh)

a.Compile text
a.Execute
```

Notes: AppleScript code:

```
set theApplication to ”KeyboardViewerServer”
set thePath to ”/System/Library/Components/KeyboardViewer.component/Contents/SharedSupport/KeyboardViewerServer.app”

set POSIXPath to ((POSIX file thePath) as string)
tell application ”System Events” to set isRunning to 0 <(count (application processes whose name is theApplication))
if isRunning then tell application POSIXPath to quit
delay 0.15

ignoring application responses
tell application POSIXPath to run
end ignoring
```

8.0.202 How to show the mouse cursor on Mac?

Plugin Version: all, Platform: macOS.

Answer: Try this declare:
Example:

```
Declare Sub ShowCursor Lib ”Carbon” ()

ShowCursor
```

Notes: The MBS Plugin has this function and supports it on Windows, too.

8.0.203 How to shutdown a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:
Example:

```
dim ae as appleevent
ae=newappleEvent(”FNDR”,”shut”,”MACS”)
if not ae.send then
msgBox ”The computer couldn’t be shutdown.”
end if
```

Notes: Or toolbox call (Attention: This method will stop the computer immediataly: No document asked to be saved, all applications quitting without knowing).

```
Declare Sub ShutDwnPower Lib ”Carbon” ()
ShutDwnPower
```

8.0.204 How to sleep a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:
Example:

```
dim ae as appleevent
ae=newappleEvent(”FNDR”,”slep”,”MACS”)
if not ae.send then
msgBox ”The computer doesn’t want to sleep.”
end if
```

8.0.205 How to speed up rasterizer for displaying PDFs with DynaPDF?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Here a few speed tips:
Notes:

- Use the DynaPDFRasterizerMBS function instead of our render functions.
- Reuse DynaPDFRasterizerMBS as long as the target picture size doesn’t change.
- Import only the PDF pages you want to display.
- Let DynaPDF do zooming, rotating or other effects instead of you change it.

8.0.206 How to use PDFLib in my RB application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The PDFlib plugin was discontinued in favor of our DynaPDF plugin.
Notes: If you need help to move, please contact us.

8.0.207 How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.
Example:

```
msgbox "This String contains ""quotes""."
```

8.0.208 How to use Sybase in Web App?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use our MBS Xojo SQL Plugin to connect to a Sybase Database in your web application.
Notes: If you see db.Connect giving the error message "cs_ctx_alloc ->CS_MEM_ERROR", than some things are not setup right for Sybase.
The Apache process may not have all the SYBASE environment variables being set when the CGI was launched.

Adding these lines to /etc/httpd/conf/httpd.conf stopped the faux memory errors for us:

```
SetEnv LD_LIBRARY_PATH /opt/sybase/OCS-15_0/lib:/opt/sybase/OCS-15_0/lib3p64:/opt/sybase/OCS-15_0/lib3p:
SetEnv SYBROOT /opt/sybase
SetEnv SYBASE_OCS /opt/sybase
SetEnv SYBASE /opt/sybase
```

8.0.209 How to use the Application Support folder?

Plugin Version: all, Platform: macOS.

Answer:

I was saving a registration code for an app to the Preferencefolder.
People on the list have suggested that it would be better in
the ApplicationSupportFolder. How do I save the file called CWWPrefs
into that folder using MBS?

I have checked for examples and the docs but can't see how to apply it

```
//f = SpecialFolder.Preferences.child("CWWPrefs")
f = ApplicationSupportFolderMBS(-32768)
```

Example:

```
dim folder,file as FolderItem

folder = createApplicationSupportFolderMBS(-32763)

if folder=nil then
// Some very old Mac OS Versions may not support it
// or the plugin may fail for any reason
folder=SpecialFolder.Preferences
end if

file=folder.Child("CWWPrefs")

MsgBox file.NativePath
```

Notes:

You may not be able to write there with a normal user account!

8.0.210 How to use the IOPMCopyScheduledPowerEvents function in Xojo?

Plugin Version: all, Platform: macOS.

Answer: You can use the following code which does this using the SoftDeclareMBS class.
Example:

```
Sub Open()
dim c as CFDateMBS
dim t as CFAbsoluteTimeMBS

// get current date
c=NewCFDateMBS

// in absolute time (seconds since x)
t=c.AbsoluteTime

// add 600 seconds (= 10 Minutes)
t.Value=t.Value+600

// Make a Date from it
c=t.Date

// Schedule the event
// 0 on success
// E00002C1 for missing root rights
```

```
Title=hex(schedulePowerEvent(c, "wake"))

// Just for information, display the scheduled stuff
CFShowMBS CopyScheduledPowerEvents
End Sub

Function CopyScheduledPowerEvents() As cfarrayMBS
dim s as SoftDeclareMBS
dim m as MemoryBlock

s=new SoftDeclareMBS

if s.LoadLibrary("IOKit.framework") then
if s.LoadFunction("IOPMCopyScheduledPowerEvents") then
if s.CallFunction(0,nil) then
Return NewCFArrayMBSHandle(s.Result,true)
else
MsgBox "Failed to Call IOPMCopyScheduledPowerEvents."
end if
else
MsgBox "Failed to load IOPMCopyScheduledPowerEvents."
end if
else
MsgBox "Failed to load IOKit."
end if

Return nil
End Function

Function SchedulePowerEvent(time_to_wake as CFDateMBS, Type as CFStringMBS) as Integer
dim s as SoftDeclareMBS
dim m as MemoryBlock

'/*
'* Types of power event
'* These are potential arguments to IOPMSchedulePowerEvent().
'* These are all potential values of the kIOPMPowerEventTypeKey in the CFDictionaries
'* returned by IOPMCopyScheduledPowerEvents().
'*/
'/*!
'@define kIOPMAutoWake
'@abstract Value for scheduled wake from sleep.
'*/
'#define kIOPMAutoWake "wake"
'
'/*!
'@define kIOPMAutoPowerOn
'@abstract Value for scheduled power on from off state.
```

```
'*/
'#define kIOPMAutoPowerOn "poweron"
'
'/*!
'@define kIOPMAutoWakeOrPowerOn
'@abstract Value for scheduled wake from sleep, or power on. The system will either wake OR
'power on, whichever is necessary.
'*/
'
'#define kIOPMAutoWakeOrPowerOn "wakepoweron"
'/*!
'@define kIOPMAutoSleep
'@abstract Value for scheduled sleep.
'*/
'
'#define kIOPMAutoSleep "sleep"
'/*!
'@define kIOPMAutoShutdown
'@abstract Value for scheduled shutdown.
'*/
'
'#define kIOPMAutoShutdown "shutdown"

s=new SoftDeclareMBS

if s.LoadLibrary("IOKit.framework") then
if s.LoadFunction("IOPMSchedulePowerEvent") then

m=NewMemoryBlock(12)
m.Long(0)=time_to_wake.handle
m.Long(4)=0 // nil
m.Long(8)=type.Handle

if s.CallFunction(3,m) then
Return s.Result
end if
end if
end if

End Function
```

Notes: Requires Mac OS X and to execute root rights.

8.0.211 How to validate a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use this function below which uses a regular expression to verify that the string is a valid UUID/GUID:
Example:

```
Function IsGUID(guid as string) As Boolean
dim r as new RegEx

r.SearchPattern = ”^(\{ { 0,1 } ( [ 0-9a-fA-F ] ) { 8 } -( [ 0-9a-fA-F ] ) { 4 } -( [ 0-9a-fA-F ] ) { 4 }
-( [ 0-9a-fA-F ] ) { 4 } -( [ 0-9a-fA-F ] ) { 12 } \} { 0,1 } )$ ”

Return r.Search(guid)<>nil
End Function
```

Notes: Simply parsing the GUID with CFUUIDMBS does not give the same result as CFUUIDMBS will also take a string like ”DDDD”.

8.0.212 How to walk a folder hierarchie non recursively?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use code like this one:
Example:

```
Sub Walk(folder as FolderItem)
dim folders() as FolderItem

folders.Append folder

while UBound(folders)>=0

dim currentFolder as FolderItem = folders.pop

dim c as Integer = currentFolder.Count
for i as Integer = 1 to c
dim item as FolderItem = currentFolder.TrueItem(i)

if item = Nil then
// no permission
elseif item.Visible then // only visible

if item.Directory then
folders.Append item
```

```
else
// work with file here
end if

end if

next

wend
End Sub
```

Notes: As you see we go with a long loop which runs until we don't have more folders to process.
We ignore items we can't access due to permission limits.
And we only work visible items.
If you like, check folderitem.isBundleMBS on item to handle packages and applications better on Mac OS X.

8.0.213 I got this error: PropVal, QDPictMBS.Name (property value), Type mismatch error. Expected CGDataProviderMBS, but got Variant, Name:QDPictMBS

Plugin Version: all, Platform: macOS.

Answer: The plugins MacOSX and MacOSXCF belong together. If you use one part, please also install the other part.
Notes: We splitted the plugin because the Xojo IDE on Windows crashed on compilation.

8.0.214 I registered the MBS Plugins in my application, but later the registration dialog is shown.

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: There are two main reasons.
Notes: 1. you may use the plugin before registering them. This is often the case if you register in a window open event and use the plugin in a control open event.
On the console on Mac OS X or Windows, you may see a message like this "MBS Plugins were used by the application before the RegisterMBSPlugin function was called. Please fix this in your code!".

2. you may have mixed different plugin versions which are not compatible.
In this case you can see a message "Internal plugin registration error." on the console on Mac OS X. Newer plugins may show a message dialog reporting this. Older version simply think they are not registered.

If the installer just merges old and new applications, users may have libraries of older and newer plugin versions in the libs folder. If your application loads the wrong version, the registration fails.

If you use remote debugging, make sure you clear the tempory files there, too. Otherwise you may have old DLLs on your hard disc which may disturb your application.

You can run into issues if you use your registration code on different places of your app. Please register only once in app.open (or app Constructor). If you have several codes, simply call them one after the other.

Also check that you only call RegisterMBSPlugin with valid serial number. If you later call RegisterMBSPlugin with Demo like in example code above, you remove the license.

Next check if you can clear the Xojo caches and that helps. This includes the Xojo Scratch folder and the Plugins & Project caches. Simply locate those folders and delete them. For Windows look in hidden AppData folder in your user folder. For Mac, please check
textasciitilde /Library/Caches and your temp folders.

Finally make sure you use the right serial number. Not an older one or a misspelled one.

8.0.215 I want to accept Drag & Drop from iTunes

Plugin Version: all, Platform: macOS.

Answer: You need to accept AcceptMacDataDrop "itun" and Handle the DropObject.
Example:

```
Sub Open()
window1.AcceptMacDataDrop "itun"
End Sub

Sub DropObject(obj As DragItem)
dim s as string
dim f as folderItem
dim d as CFDictionaryMBS
dim o as CFObjectMBS
dim key as CFStringMBS
dim dl as CFDictionaryListMBS
dim i,c as Integer
dim u as CFURLMBS
dim file as FolderItem

if obj.MacDataAvailable("itun") then
s = obj.MacData("itun")
```

```
// Parse XML
o=NewCFObjectMBSFromXML(NewCFBinaryDataMBSStr(s))

// Make dictionary
if o isa CFDictionaryMBS then
d=CFDictionaryMBS(o)

// get Tracks Dictionary
key=NewCFStringMBS("Tracks")
o=d.Value(key)

if o isa CFDictionaryMBS then
d=CFDictionaryMBS(o)
dl=d.List

// Walk over all entries in the Tracks dictionary
c=dl.Count-1
for i=0 to c
o=dl.Value(i)

if o isa CFDictionaryMBS then
d=CFDictionaryMBS(o)

key=NewCFStringMBS("Location")
o=d.Value(key)
if o isa CFStringMBS then
u=NewCFURLMBSCFStringMBS(CFStringMBS(o),nil)

file=u.file
if file<>nil then
MsgBox file.NativePath
end if
end if
end if
next
end if
end if
end if
End Sub
```

Notes: The code above inside a window on Xojo 5.5 with MBS Plugin 5.3 will do it nice and show the paths.

8.0.216 I'm drawing into a listbox but don't see something.

Plugin Version: all.

Answer: If you draw this in a listbox cellbackground, you need to draw on the correct position
Example:

```
Function CellBackgroundPaint(g As Graphics, row as Integer, column as Integer) As Boolean
dim f as FolderItem
f=SpecialFolder.Desktop
f.DrawWideIconMBS(g,listbox1.left,listbox1.top+row*20,16)
Return true
End Function
```

Notes: Try this in a listbox. The Graphics object there has a cliping and an offset which the plugin doesn't know about.

8.0.217 I'm searching for a method or so to move a window from position x.y to somewhere else on the screen.

Platform: macOS.

Answer:

The code I produced in RB isn't smooth enough. Is there a call in MBS, if not, can it be done? The speed of it has to be like the show of a DrawerWindow.

Try the declare below for Carbon. With WindowLib it will work on Mac OS 8.5 and newer.
Notes:

See Window.Transition functions.

8.0.218 If I use one of your plug-ins under windows, would this then impose the use of dll after compilation or my would my compiled soft still be a stand-alone single file software?

Platforms: macOS, Linux, Windows.

Answer: Stand alone.
Notes: Xojo compiles all used plugins into the application binary.
Some plugin parts need external dlls but you will find that in the documentation. (e.g. pdflib for some classes)

8.0.219 Is the fn key on a powerbook keyboard down?

Plugin Version: all, Platform: macOS.

Answer: I am unable to figure out how or if it is possible to detect if the fn key is down on a powerbook keyboard. Is it possible?
Example:

```
' Window.Open Event of a blank project:

dim i as Integer

for i=0 to 127
if keyboard.asynckeydown(i) then
title=str(i) // found
return
end if
next
title="" // not found
```

Notes: This test application shows the keycode (decimal) 63 for the fn key.

8.0.220 Is there a case sensitive Dictionary?

Plugin Version: all.

Answer: The MBS Plugin has several classes which can work as a replacement.
Notes: First you could use VariantToVariantHashMapMBS or VariantToVariantOrderedMapMBS.
If you know that all keys are Strings or Integers only, you can use the specialized classes which are a little bit faster due to avoiding variants:

IntegerToIntegerHashMapMBS class
IntegerToIntegerOrderedMapMBS class
IntegerToStringHashMapMBS class
IntegerToStringOrderedMapMBS class
IntegerToVariantHashMapMBS class
IntegerToVariantOrderedMapMBS class
StringToStringHashMapMBS class
StringToStringOrderedMapMBS class
StringToVariantHashMapMBS class
StringToVariantOrderedMapMBS class

8.0.221 Is there a way to use the MBS plugin to get only the visible item and folder count on a volume?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use the DirectorySizeMBS class for this as in the example below:
Example:

```
dim d as DirectorySizeMBS

d=new DirectorySizeMBS

// volume(1) as my boot volume is very full
if d.update(volume(1),true,0) then
MsgBox str(d.VisibleItemCount)+" visible items, "+str(d.HiddenItemCount)+" invisible items."
end if
```

Notes: Complete Question: Is there a way to use the MBS plugin to get only the visible item and folder count on a volume? The FileCount and FolderCount properties of
VolumeInformationMBS seem to provide the total # of items including
invisible items such as .DS_Store and more importantly .Trashes which
is causing me a great amount of difficulty during a recursive scan of a
volume. I've got a progress bar which uses the total of the filecount
and foldercount properties as the maximum value, but my routine needs
to filter out all invisible items, as it is creating a catalog of a
volume for archiving purposes. Any thoughts how I could get accurate
number.

8.0.222 Is there an easy way I can launch the Displays preferences panel?

Plugin Version: all, Platform: macOS.

Answer: Use the code below:
Example:

```
dim error as Integer

error=OpenMacOSXPreferencesPaneMBS("Displays")
if error<>0 then
MsgBox "Failed to launch QuickTime System Preferences panel."
end if
```

8.0.223 List of Windows Error codes?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We have a list of windows error codes on our website.
Notes: http://www.monkeybreadsoftware.de/xojo/winerror.shtml

8.0.224 Midi latency on Windows problem?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The issue is system related, not a problem with RB or the plugin.
Notes: Two things will adversely affect the timing:

(1) latency of the software synthesizer output driver. The default Windows wavetable synthesizer has considerable latency. I don't know how many milliseconds, but it is noticeable.

(2) latency of the digital audio output driver. Different systems have different drivers for different audio hardware. My Dell laptop has a minimum 15ms latency in the audio driver.

These two things put together were causing a very sluggish MIDI response. I was able to verify these as the culprits by routing MIDI directly out of RB into a sample player, which only introduces the latency of (2) and does not include latency of (1).

I don't know how widely known are these facts, if not then you may want to add this information to the documentation, since Windows programmers using the MIDI plugin may not know those problems, and might mistakenly blame your plugin, as I did :) Sorry about that!

(From Aaron Andrew Hunt)

8.0.225 My Xojo Web App does not launch. Why?

Plugin Version: all, Platform: macOS.

Answer: Here is a list of checks to do for linux apache installations with Xojo or Xojo Web applications:
Notes: Just a list of checks to do for linux apache installations:

- You have 64bit linux? Than you need 32 bit compatibility libraries.
- The folder of your app is writable? Set permissions to 777.
- The cgi script is executable? Set permissions to 755.

- The app file itself is executable? Set permissions to 755.
- You uploaded cgi file as text, so it has unix line endings? (this often gives error ”Premature end of script headers” in apache log)
- You uploaded config.cfg file and made it writable? Set permissions to 666.
- Your apache allows execution of cgi scripts? You enabled cgi for apache and uncommented addhandler command for CGI on a new apache installation?
- You uploaded the app file and libraries as binary files? Upload as text breaks them.
- You did upload the libs folder?
- You don’t have code in app.open, session.open and other events which crashes app right at launch?
- You don”t have a print command in your app.open event? (see feedback case 23817)
- You allowed htaccess file to overwrite permissions?

8.0.226 SQLDatabase not initialized error?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Before you can use SQLDatabaseMBS, it must be initialized.
Example:

```
dim d as new SQLDatabaseMBS
```

Notes: This happens normally when you use ”new SQLDatabaseMBS”.
But if you just have a SQLConnectionMBS and get a recordset there, the initialization may not have happend, yet.
So please simply add a line ”dim d as new SQLDatabaseMBS” to your app.open code after registration, so the plugin part can initialize and late provide recordsets.

8.0.227 Textconverter returns only the first x characters. Why?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Some older Xojo versions limit the Textconverter to around 1024 characters in input and output.
This should be fixed with RB5.
Notes:

Xojo seems not to support Textconverters at all on Windows.

8.0.228 The type translation between CoreFoundation/Foundation and Xojo data types.

Plugin Version: all, Platform: macOS.

Answer: The plugin does conversion between Cocoa/Carbon data types and native Xojo data types. The following list help you knowing what the current plugins support:
Notes: Cocoa NSObject to Variant:

nil ->nil
NSDictionary ->Dictionary
NSData ->MemoryBlock
NSString ->String
NSAttributedString ->NSAttributedStringMBS
NSDate ->Date
NSNumber ->double/integer/Int64/UInt64/UInt32/Boolean
NSURL ->String
NSValue with NSRect ->NSRectMBS
NSValue with NSPoint ->NSPointMBS
NSValue with NSSize ->NSSizeMBS
NSValue with NSRange ->NSRangeMBS
NSValue with QTTime ->QTTimeMBS
NSValue with QTTimeRange ->QTTimeRangeMBS
NSArray ->Array of Variant
QuartzFilter ->QuartzFilterMBS

- ->*MBS

Variant to Cocoa NSObject:

nil ->nil
Dictionary ->NSDictionary
Boolean ->NSNumber
Integer ->NSNumber
Color ->NSColor
Int64 ->NSNumber
Single ->NSNumber
Double ->NSNumber
Date ->NSDate
MemoryBlock ->NSData
String ->NSString
NSImageMBS ->NSImage
NSAttributedStringMBS ->NSAttributedString
NSColorMBS ->NSColor
NSRectMBS ->NSValue with NSRect
NSSizeMBS ->NSValue with NSSize

NSPointMBS ->NSValue with NSPoint
NSRangeMBS ->NSValue with NSRange
NSBurnMBS ->NSBurn
NSViewMBS ->NSView
NSFontMBS ->NSFont
NSParagraphStyleMBS ->NSParagraphStyle
NSAttributedStringMBS ->NSAttributedString
WebPolicyDelegateMBS ->WebPolicyDelegate
WebUIDelegateMBS ->WebUIDelegate
WebFrameLoadDelegateMBS ->WebFrameLoadDelegate
WebResourceLoadDelegateMBS ->WebResourceLoadDelegate
NSIndexSetMBS ->NSIndexSet
QTTimeMBS ->QTTime
QTTimeRangeMBS ->QTTimeRange
Array of Variant ->NSArray
Array of String ->NSArray
CFStringMBS ->NSString
CFNumberMBS ->NSNumber
CFDataMBS ->NSData
CFURLMBS ->NSURL
CFArrayMBS ->NSArray
CFDictionaryMBS ->NSDictionary
CFBinaryDataMBS ->NSDate

Carbon CFTypeRef to Variant:

CFDictionaryRef ->Dictionary
CFStringRef ->String
CFDataRef ->String
CFURL ->String
CFNumber ->Integer/Double/Int64
CFArray ->Array
CFDate ->date
nil ->nil
CGColorSpace ->CGColorSpaceMBS
CGColor ->CGColorMBS
CGImage ->CGImageMBS
CF* ->CF*MBS

Variant to Carbon CFTypeRef:

Dictionary ->CFDictionaryRef
Boolean ->CFBooleanRef
Color ->CFNumberRef
Integer ->CFNumberRef

Int64 ->CFNumberRef
Single ->CFNumberRef
Double ->CFNumberRef
String ->CFStringRef
Color ->CGColorRef
Date ->CFDateRef
nil ->nil
Memoryblock ->CFDataRef
FolderItem ->CFURLRef
Dictionary ->CFDictionaryRef
Array of Variant/String/Date/Double/Single/Int64/Integer ->CFArray
CGRectMBS ->CGRect as CFDataRef
CGSizeMBS ->CGSize as CFDataRef
CGPointMBS ->CGPoint as CFDataRef
CGColorMBS ->CGColor
CGColorSpaceMBS ->CGColorSpace
CGImageMBS ->CGImage
CGDataConsumerMBS ->CGDataConsumer
CGDataProviderMBS ->CGDataProvider
CF*MBS ->CF*

Strings without encodings should be put into dictionaries as memoryblocks.

8.0.229 Uploaded my web app with FTP, but it does not run on the server!

Plugin Version: all, Platform: Windows.

Answer: If you see errors like a simple ”Segmentation Fault” on Linux or some other wired errors, you may want to check your FTP upload mode. It must be binary for web apps. ASCII mode corrupts the application.

8.0.230 What classes to use for hotkeys?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use CarbonHotKeyMBS class on Mac and WindowsKeyFilterMBS on Windows.
Notes: CarbonHotKeyMBS will also work fine in Cocoa apps.

8.0.231 What do I need for Linux to get picture functions working?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: In order to get our plugins working on Linux systems without GUI, the plugin loads graphics

libraries dynamically.
Notes: To get it working, the plugin tries to load gtk with this paths:

- libgtk-x11-2.0.so"
- libgtk-x11-2.0.so.0"
- /usr/lib/libgtk-x11-2.0.so"
- /usr/lib32/libgtk-x11-2.0.so"
- /usr/lib/libgtk-x11-2.0.so.0"
- /usr/lib32/libgtk-x11-2.0.so.0"

gdk is loaded with this paths:

- libgdk-x11-2.0.so"
- libgdk-x11-2.0.so.0"
- /usr/lib/libgdk-x11-2.0.so"
- /usr/lib32/libgdk-x11-2.0.so"
- /usr/lib/libgdk-x11-2.0.so.0"
- /usr/lib32/libgdk-x11-2.0.so.0"

For the paths without explicit path, the system will search in /lib, /usr/lib and all directories in the LD_LIBRARY_PATH environment variable.

8.0.232 What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

8.0.233 What font is used as a 'small font' in typical Mac OS X apps?

Plugin Version: all, Platform: macOS.

Answer:

Xojo 4.5 has a constant "SmallSystem" to use for a font name.
For older versions try this code:
Example:

```
Sub GetThemeFont(fontType as Integer, ByRef fontName as String, ByRef fontSize as Integer, ByRef fontStyle as Integer)
dim err as Integer
dim theFont, theFontSize, theFontStyle as MemoryBlock

const smSystemScript = -1

Declare Function GetThemeFont Lib "Carbon" (inFontID as Integer, inScript as Integer, outFontName as Ptr, outFontSize as Ptr, outStyle as Ptr) as Integer

theFont = NewMemoryBlock(256) //Str255
theFontSize = NewMemoryBlock(2) //SInt16
theFontStyle = NewMemoryBlock(1) //Style

err = GetThemeFont(fontType, smSystemScript, theFont, theFontSize, theFontStyle)

if err = 0 then
fontName = theFont.PString(0)
fontSize = theFontSize.UShort(0)
fontStyle = theFontStyle.Byte(0)
else
fontName = ""
fontSize = 0
fontStyle = 0
end if
End Sub
```

8.0.234 What is last plugin version to run on Mac OS X 10.4?

Plugin Version: all, Platform: Windows.

Answer: Last Version with 10.4 support is version 15.4.
Notes: With version 15.4 you can build applications for OS X 10.4 and newer.
For Version 16.0 we disabled 10.4 and moved minimum to 10.5. We may be able to enable it again to build a version of 16.x, but may need to charge for this by hour.

8.0.235 What is last plugin version to run on PPC?

Plugin Version: all, Platform: Windows.

Answer: Last Version with PPC is 15.4.
Notes: With version 15.4 you can build PPC applications for OS X 10.4 and newer.
For Version 16.0 we disabled PPC. We may be able to enable it again to build a PPC version of 16.x, but may need to charge for this by hour.

8.0.236 What is last version of the plugins for macOS 32-bit?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use version 23.0 or older.
Notes: We stopped including 32-bit code for macOS in version 23.1.
Please us older versions if you use an old Xojo.

Xojo 2017r3 and newer load our 64-bit plugins.

8.0.237 What is the difference between Timer and WebTimer?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Time is server side and WebTimer client side.
Notes: Timer is the normal timer class in Xojo. It runs on the server. On the side the WebTimer runs on the client. It triggers a request to the server to perform the action. So a WebTimer is good to keep the connection running and the website updated regularly. A timer on the server is good to make regular jobs like starting a database backup every 24 hours.

8.0.238 What is the list of Excel functions?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Below a list of function names known by LibXL.
Notes: LibXL parses the functions and writes tokens to the excel file. So even if Excel can do more functions, we can only accept the ones known by LibXL.

ABS, ABSREF, ACOS, ACOSH, ACTIVE.CELL, ADD.BAR, ADD.COMMAND, ADD.MENU, ADD.TOOLBAR, ADDRESS, AND, APP.TITLE, AREAS, ARGUMENT, ASC, ASIN, ASINH, ATAN, ATAN2, ATANH, AVEDEV, AVERAGE, AVERAGEA, BAHTTEXT, BETADIST, BETAINV, BINOMDIST, BREAK, CALL, CALLER, CANCEL.KEY, CEILING, CELL, CHAR, CHECK.COMMAND, CHIDIST, CHIINV, CHITEST, CHOOSE, CLEAN, CODE, COLUMN, COLUMNS, COMBIN, CONCATENATE, CONFIDENCE, CORREL, COS, COSH, COUNT, COUNTA, COUNTBLANK, COUNTIF, COVAR, CREATE.OBJECT, CRITBINOM, CUSTOM.REPEAT, CUSTOM.UNDO, DATE, DATEDIF, DATESTRING, DATEVALUE, DAVERAGE, DAY, DAYS360, DB, DBCS, DCOUNT, DCOUNTA, DDB, DEGREES, DELETE.BAR, DELETE.COMMAND, DELETE.MENU, DELETE.TOOLBAR, DEREF, DEVSQ, DGET, DIALOG.BOX, DIRECTORY, DMAX, DMIN, DOCUMENTS, DOLLAR, DPRODUCT, DSTDEV, DSTDEVP, DSUM, DVAR, DVARP, ECHO, ELSE, ELSE.IF, ENABLE.COMMAND, ENABLE.TOOL, END.IF, ERROR, ERROR.TYPE, EVALUATE, EVEN, EXACT, EXEC, EXECUTE, EXP, EXPONDIST, FACT, FALSE, FCLOSE, FDIST, FILES, FIND, FINDB, FINV, FISHER, FISHERINV, FIXED, FLOOR, FOPEN, FOR, FOR.CELL, FORECAST, FORMULA.CONVERT, FPOS, FREAD, FREADLN, FREQUENCY, FSIZE, FTEST, FV, FWRITE, FWRITELN, GAMMADIST, GAMMAINV, GAMMALN, GEOMEAN, GET.BAR, GET.CELL, GET.CHART.ITEM, GET.DEF, GET.DOCUMENT, GET.FORMULA, GET.LINK.INFO, GET.MOVIE, GET.NAME, GET.NOTE, GET.OBJECT, GET.PIVOT.FIELD, GET.PIVOT.ITEM, GET.PIVOT.TABLE, GET.TOOL, GET.TOOLBAR, GET.WINDOW, GET.WORKBOOK, GET.WORKSPACE, GETPIVOTDATA, GOTO, GROUP, GROWTH, HALT, HARMEAN, HELP, HLOOKUP, HOUR, HYPERLINK, HYPGEOMDIST, IF, INDEX, INDIRECT, INFO, INITIATE, INPUT, INT, INTERCEPT, IPMT, IRR, ISBLANK, ISERR, ISERROR, ISLOGICAL, ISNA, ISNONTEXT, ISNUMBER, ISPMT, ISREF, ISTEXT, ISTHAIDIGIT, KURT, LARGE, LAST.ERROR, LEFT, LEFTB, LEN, LENB, LINEST, LINKS, LN, LOG, LOG10, LOGEST, LOGINV, LOGNORMDIST, LOOKUP, LOWER, MATCH, MAX, MAXA, MDETERM, MEDIAN, MID, MIDB, MIN, MINA, MINUTE, MINVERSE, MIRR, MMULT, MOD, MODE, MONTH, MOVIE.COMMAND, N, NA, NAMES, NEGBINOMDIST, NEXT, NORMDIST, NORMINV, NORMSDIST, NORMSINV, NOT, NOTE, NOW, NPER, NPV, NUMBERSTRING, ODD, OFFSET, OPEN.DIALOG, OPTIONS.LISTS.GET, OR, PAUSE, PEARSON, PERCENTILE, PERCENTRANK, PERMUT, PHONETIC, PI, PIVOT.ADD.DATA, PMT, POISSON, POKE, POWER, PPMT, PRESS.TOOL, PROB, PRODUCT, PROPER, PV, QUARTILE, RADIANS, RAND, RANK, RATE, REFTEXT, REGISTER, REGISTER.ID, RELREF, RENAME.COMMAND, REPLACE, REPLACEB, REPT, REQUEST, RESET.TOOLBAR, RESTART, RESULT, RESUME, RETURN, RIGHT, RIGHTB, ROMAN, ROUND, ROUNDBAHTDOWN, ROUNDBAHTUP, ROUNDDOWN, ROUNDUP, ROW, ROWS, RSQ, RTD, SAVE.DIALOG, SAVE.TOOLBAR, SCENARIO.GET, SEARCH, SEARCHB, SECOND, SELECTION, SERIES, SET.NAME, SET.VALUE, SHOW.BAR, SIGN, SIN, SINH, SKEW, SLN, SLOPE, SMALL, SPELLING.CHECK, SQRT, STANDARDIZE, STDEV, STDEVA, STDEVP, STDEVPA, STEP, STEYX, SUBSTITUTE, SUBTOTAL, SUM, SUMIF, SUMPRODUCT, SUMSQ, SUMX2MY2, SUMX2PY2, SUMXMY2, SYD, T, TAN, TANH, TDIST, TERMINATE, TEXT, TEXT.BOX, TEXTREF, THAIDAYOFWEEK, THAIDIGIT, THAIMONTHOFYEAR, THAINUMSOUND, THAINUMSTRING, THAISTRINGLENGTH, THAIYEAR, TIME, TIMEVALUE, TINV, TODAY, TRANSPOSE, TREND, TRIM, TRIMMEAN, TRUE, TRUNC, TTEST, TYPE, UNREGISTER, UPPER, USDOLLAR, USERDEFINED, VALUE, VAR, VARA, VARP, VARPA, VDB, VIEW.GET, VLOOKUP, VOLATILE, WEEKDAY, WEIBULL, WHILE, WINDOW.TITLE, WINDOWS, YEAR and ZTEST.

8.0.239 What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

8.0.240 What to do on Xojo reporting a conflict?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

I get an error like ”This item conflicts with another item of the same name” when using one of the plugin functions.

Xojo just wants to tell you that you dropped something in the plugins folder what is not a plugin.
Notes:

Some users dropped the examples, the documentation or other files into the plugins folder. Don't do it.

8.0.241 What to do with a NSImageCacheException?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You need to add exception handlers for NSExceptionMBS in order to catch this exception.
Notes: You may also add code to write the stack of the exception into a log file for later locating the error source.

A NSImage has several image representations in memory. So basicly you pass in the base image and for whatever size an image is needed, the NSImage class will create a cache image representation of the requested size so on the next query it can use that cache for the same requested size.

8.0.242 What to do with MySQL Error 2014?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can get this error on MySQL if you have a recordset open while you create another one.

8.0.243 What to do with SQL Plugin reporting Malformed string as error?

Plugin Version: all, Platform: macOS.

Answer: Please make sure the table and/or database fields have a text encoding set.
Notes: For Firebird our plugin tries to use UTF-8 encoding if possible and to correctly convert between various tables, the tables and their fields need to have a text encoding defined.
e.g. if the text field in the table is windows-1252 and the other ISO 8859-5, then the Firebird database can convert them to UTF-8 and deliver texts to the plugin.
If encoding is set to none, it may get confused for non-ascii text.

8.0.244 Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

8.0.245 Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

8.0.246 Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

8.0.247 Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

8.0.248 Where is GetObjectClassNameMBS?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use this replacement method:
Example:

```
Function GetObjectClassNameMBS(o as Object) As string
dim t as Introspection.TypeInfo = Introspection.GetType(o)
Return t.FullName
End Function
```

Notes: GetObjectClassNameMBS was removed from the plugins.

8.0.249 Where is NetworkAvailableMBS?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: We removed NetworkAvailableMBS some versions ago. It was not working right and basicly it's not useful. If you want to check whether you have a network, than do a DNS resolve:
Example:

```
// two independend domain names
const domain1 = "www.google.com"
const domain2 = "www.macsw.de"

// resolve IPs
dim ip1 as string = DNSNameToAddressMBS(Domain1)
dim ip2 as string = DNSNameToAddressMBS(Domain2)

// if we got IPs and not the same IPs (error/login pages)
if len(ip1)=0 or len(ip2)=0 or ip1=ip2 then
MsgBox "no connection"
else
MsgBox "have connection"
end if
```

Notes: This way you can detect whether you got something from DNS. And you can make sure that a DNS redirection to a login page won't catch you.

8.0.250 Where is StringHeight function in DynaPDF?

Plugin Version: all, Platform: Windows.

Answer: Use the function GetFTextHeight or GetFTextHeightEx.
Notes: Be aware that GetFTextHeight works with format commands and you may want to escape your text if you don't use them.

8.0.251 Where is XLSDocumentMBS class?

Plugin Version: all, Platform: macOS.

Answer: This class has been removed in favor of XLBookMBS class.
Notes: This classes have been removed XLSCellMBS, XLSDocumentMBS, XLSFormatRecordMBS, XLSMergedCellsMBS, XLSRowMBS and XLSSheetMBS.

8.0.252 Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:
http://www.wotsit.org

8.0.253 Where to register creator code for my application?

Plugin Version: all, Platform: macOS.

Answer:

Register at Apple:
http://developer.apple.com/dev/cftype/information.html

8.0.254 Which Mac OS X frameworks are 64bit only?

Plugin Version: all, Platform: macOS.

Answer: Some frameworks from Mac OS X do not support 32 bit applications, so we can't provide plugins for Xojo until 64bit target is available.
Notes: For Mac OS X 10.8:

- Accounts
- EventKit
- GLKit
- Social

and in 10.9:

- Accounts
- AVKit
- EventKit
- GameController
- GLKit
- MapKit
- MediaLibrary
- Social
- SpriteKit

In general Apple makes all new frameworks being 64 bit only.

8.0.255 Which plugins are 64bit only?

Plugin Version: all, Platform: macOS.

Answer: Some of our plugins work only in 64 bit modes as operation systems do not provide 32 bit code.
Notes: This effects currently: EventKit, Accounts, Social frameworks from Apple and our matching plugins.

8.0.256 Why application doesn't launch because of a missing ddraw.dll!?

Plugin Version: all, Platform: Windows.

Answer: Some RB versions require that you install DirectX from Microsoft on your Windows.

8.0.257 Why application doesn't launch because of a missing shlwapi.dll!?

Plugin Version: all, Platform: Windows.

Answer: Some RB versions require that you install the Internet Explorer from Microsoft on your Windows.
Notes: This bug is for several older Windows 95 editions.

8.0.258 Why do I hear a beep on keydown?

Plugin Version: all, Platform: Windows.

Answer: When the user presses a key, RB goes through all keydown event handlers till on returns true.
Notes: If no keydown event handler returns true for the key, a beep is performed.

8.0.259 Why does folderitem.item return nil?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Because Xojo fails to make a folderitem for you. Reason may be an alias file which can't be resolved or simply that you don't have enough access rights to read the folder content.
Notes: A more rarely reason is that the directory changed and the file with the given index or name does no longer exist.

8.0.260 Why doesn't showurl work?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

There are three main reasons:

1. showurl is not supported by Xojo in 68k applications.
2. there is now application defined for the protocol (e.g. http) in the Internet Control panel.
3. You don't have Internet Config installed.

You can use the InternetConfigMBS class to check for this stuff.

8.0.261 Why don't the picture functions not work on Linux?

Plugin Version: all, Platform: macOS.

Answer: Please make sure libcairo is installed.
Notes: For accessing pictures on Linux, the MBS Plugin relays on the cairo library.
Please install the package if you don't have it already.
Our plugin looks for library called libcairo.so or libcairo.so.2.

8.0.262 Why have I no values in my chart?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You have no data points visible, there may be several reasons:
Notes: For example one of the data values may be infinite or invalid.
Or the scaling may be out of range, so you simply see nothing.

8.0.263 Will application size increase with using plugins?

Plugin Version: all, Platform: Windows.

Answer: All plugins used by your application will be included in the application.
Notes: If you use no plugins, your application will not change size.
And if you use one class from the plugins, your application size will increase by a few kilobytes.
The documentation of the plugins include a list of all plugin parts and their sizes for the different platforms.

8.0.264 XLS: Custom format string guidelines

Plugin Version: all, Platform: macOS.

Answer: You have to download the source code and compile a static version of the library.
Notes: Up to four sections of format codes can be specified. The format codes, separated by semicolons, define the formats for positive numbers, negative numbers, zero values, and text, in that order. If only two sections are specified, the first is used for positive numbers and zeros, and the second is used for negative numbers. If only one section is specified, it is used for all numbers. Four sections example:

#,###.00_); [Red] (#,###.00);0.00;"sales "@

The following table describes the different symbols that are available for use in custom number formats.

Specify colors

To set the text color for a section of the format, type the name of one of the following eight colors in square brackets in the section. The color code must be the first item in the section.

Instead of using the name of the color, the color index can be used, like this [Color3] for Red. Valid numeric indexes for color range from 1 to 56, which reference by index to the legacy color palette.
Specify conditions

To set number formats that will be applied only if a number meets a specified condition, enclose the condition in square brackets. The condition consists of a comparison operator and a value. Comparison operators include: = Equal to; >Greater than; <Less than; >= Greater than or equal to, <= Less than or equal to, and <>Not equal to. For example, the following format displays numbers that are less than or equal to 100 in a red font and numbers that are greater than 100 in a blue font.
[Red] [<=100] ; [Blue] [>100]
If the cell value does not meet any of the criteria, then pound signs ("#") are displayed across the width of the cell.
Dates and times

Examples

8.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

8.0.266 Xojo or my RB application itself crashes on launch on Mac OS Classic. Why?

Plugin Version: all.

Answer:

You may check if the application has enough memory to be loaded.
RB should have on Mac OS Classic more than 20 MB of RAM.
I prefered to use 50 MB and for an application a 10 MB partition is a good way to start.

Parameter	Description
x	The x value of the data point. For an enumerated x-axis (see Axis.setLabels on what is an enumerated axis), the first data point is 0, and the nth data point is (n-1).
xLabel	The bottom x-axis label of the data point.
x2Label	The top x-axis label of the data point.
value	The value of the data point.
accValue	The sum of values of all data points that are in the same x position and same data group as the current data point, and with data set number less than or equal to the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
totalValue	The sum of values of all data points that are in the same x position and same data group as the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
percent	The percentage of the data point based on the total value of all data points that are in the same x position and same data group as the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
accPercent	The accumulated percentage of the data point based on the total value of all data points that are in the same x position and same data group as the current data point. This is useful for stacked charts, such as stacked bar chart and stacked area chart.
gpercent	The percentage of the data point based on the total value of all data points in a layer.
dataSet	The data set number to which the data point belongs. The first data set is 0. The nth data set is (n-1).
dataSetName	The name of the data set to which the data point belongs.
dataItem	The data point number within the data set. The first data point is 0. The nth data point is (n-1).
dataGroup	The data group number to which the data point belongs. The first data group is 0. The nth data group is (n-1).
dataGroupName	The name of the data group to which the data point belongs.
layerId	The layer number to which the data point belongs. The first layer is 0. The nth layer is (n-1).
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using Layer.addExtraField, Layer.addExtraField2, BaseChart.addExtraField or BaseChart.addExtraField2.

diFieldN	Same as fieldN. See above.
dsFieldN	Similar to fieldN, except that dsFieldN means the extra field is indexed by data set number. The Pth data set corresponds to the Pth element of the extra field.
dsdiFieldN	Similar to fieldN, except that dsdiFieldN means the extra fields are indexed by both the data set number and data point number. The Pth data item of the Qth data set corresponds to the Pth element of the (N + Q)th extra field.

Parameter	Description
zx	The symbol scale in the x dimension. Applicable for layers with symbol scales set by LineLayer.setSymbolScale.
zy	The symbol scale in the y dimension. Applicable for layers with symbol scales set by LineLayer.setSymbolScale.
z	The symbol scale without distinguishing the dimension to use. Applicable for layers with symbol scales set by LineLayer.setSymbolScale.

Parameter	Description
slope	The slope of the trend line.
intercept	The y-intercept of the trend line.
corr	The correlation coefficient in linear regression analysis.
stderr	The standard error in linear regression analysis.

Parameter	Description
top	The value of the top edge of the box-whisker symbol.
bottom	The value of the bottom edge of the box-whisker symbol.
max	The value of the maximum mark of the box-whisker symbol.
min	The value of the minimum mark of the box-whisker symbol.
med	The value of the median mark of the box-whisker symbol.

Parameter	Description
high	The high value.
low	The low value.
open	The open value.
close	The close value.

Parameter	Description
dir	The direction of the vector.
len	The length of the vector.

Parameter	Description
radius	The radial value of the data point.
value	Same as { radius } . See above.
angle	The angular value of the data point.
x	Same as { angle } . See above.
label	The angular label of the data point.
xLabel	Same as { label } . See above.
name	The name of the layer to which the data point belongs.
dataSetName	Same as { name } . See above.
i	The data point number. The first data point is 0. The nth data point is (n-1).
dataItem	Same as { i } . See above.
z	The symbol scale. Applicable for layers with symbol scales set by PolarLayer.setSymbolScale.
fieldN	The (N + 1)th extra field. For example, { field0 } means the first extra field. An extra field is an array of custom elements added using Layer.addExtraField, Layer.addExtraField2, BaseChart.addExtraField or BaseChart.addExtraField2.

diFieldN	Same as fieldN. See above.
dsFieldN	Similar to fieldN, except that dsFieldN means the extra field is indexed by layer index. The Pth layer corresponds to the Pth element of the extra field.
dsdiFieldN	Similar to fieldN, except that dsdiFieldN means the extra fields are indexed by both the data set number and data point number. The Pth data item of the Qth layer corresponds to the Pth element of the (N + Q)th extra field.

Parameter	Description
dir	The direction of the vector.
len	The length of the vector.

Parameter	Description
value	The axis value at the tick position.
label	The axis label at the tick position.

Parameter	Description
[param]	The name of the parameter
[a]	If this field a number, it specifies the number of decimal places (digits to the right of the decimal point).

[b] The thousand separator. Should be a non-alphanumeric character (not 0-9, A-Z, a-z). Use '
textasciitilde ' for no thousand separator. The default is '
textasciitilde ', which can be modified using BaseChart.setNumberFormat.

[c] The decimal point character. The default is '.', which can be modified using BaseChart.setNumberFormat.

[d] The negative sign character. Use '
textasciitilde ' for no negative sign character. The default is '-', which can be modified using BaseChart.setNumberFormat.

Parameter	Description
yyyy	The year in 4 digits (e.g. 2002)
yyy	The year showing only the least significant 3 digits (e.g. 002 for the year 2002)
yy	The year showing only the least significant 2 digits (e.g. 02 for the year 2002)
y	The year showing only the least significant 1 digits (e.g. 2 for the year 2002)
mmm	The month formatted as its name. The default is to use the first 3 characters of the english month name (Jan, Feb, Mar ...). The names can be configured using BaseChart.setMonthNames.
mm	The month formatted as 2 digits from 01 - 12, adding leading zero if necessary.
m	The month formatted using the minimum number of digits from 1 - 12.
MMM	The first 3 characters of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
MM	The first 2 characters of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
M	The first character of the month name converted to upper case. The names can be configured using BaseChart.setMonthNames.
dd	The day of month formatted as 2 digits from 01 - 31, adding leading zero if necessary.
d	The day of month formatted using the minimum number of digits from 1 - 31.
w	The name of the day of week. The default is to use the first 3 characters of the english day of week name (Sun, Mon, Tue ...). The names can be configured using BaseChart.setWeekDayNames.
hh	The hour of day formatted as 2 digits, adding leading zero if necessary. The 2 digits will be 00 - 23 if the 'a' option (see below) is not specified, otherwise it will be 01 - 12.
h	The hour of day formatted using the minimum number of digits. The digits will be 0 - 23 if the 'a' option (see below) is not specified, otherwise it will be 01 - 12.
nn	The minute formatted as 2 digits from 00 - 59, adding leading zero if necessary.
n	The minute formatted using the minimum number of digits from 00 - 59.
ss	The second formatted as 2 digits from 00 - 59, adding leading zero if necessary.
s	The second formatted using the minimum number of digits from 00 - 59.
a	Display either 'am' or 'pm', depending on whether the time is in the morning or afternoon. The text 'am' and 'pm' can be modified using BaseChart.setAMPM.

Shape Id	Value	Description
SquareShape	1	Square shape. See (1, 1) above.
DiamondShape	2	Diamond shape. See (2, 1) above.
TriangleShape	3	Triangle shape pointing upwards. See (3, 1) above.
RightTriangleShape	4	Triangle shape pointing rightwards. See (4, 1) above.
LeftTriangleShape	5	Triangle shape pointing leftwards. See (5, 1) above.
InvertedTriangleShape	6	Triangle shape pointing downwards. See (1, 2) above.
CircleShape	7	Circle shape. See (2, 2) above.
StarShape	[Method]	Star shapes of various points. See (2, 3), (2, 4), (2, 5), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5) above for stars with 3 to 10 points.
PolygonShape	[Method]	Polygon shapes symmetrical about a vertical axis with a vertex at the top center position. See (4, 1), (4, 3), (4, 5), (5, 1) for polygons of 5 to 8 sides.
Polygon2Shape	[Method]	Polygon shapes symmetrical about a vertical axis but without any vertex at the top center position. See (4, 2), (4, 4) for polygons of 5 and 6 sides.
CrossShape	[Method]	'+' shapes. See (5, 2), (5, 3), (5, 4), (5, 5), (6, 1), (6, 2), (6. 3) for '+' shape with arm width of 0.1 - 0.7.
Cross2Shape	[Method]	'X' shapes. See (6, 4), (6, 5), (7, 1), (7, 2), (7, 3), (7, 4), (7, 5) for 'X' shapes with arm width of 0.1 - 0.7.

langEnglish	0	Roman script
langFrench	1	Roman script
langGerman	2	Roman script
langItalian	3	Roman script
langDutch	4	Roman script
langSwedish	5	Roman script
langSpanish	6	Roman script
langDanish	7	Roman script
langPortuguese	8	Roman script
langNorwegian	9	Roman script
langHebrew	10	Hebrew script
langJapanese	11	Japanese script
langArabic	12	Arabic script
langFinnish	13	Roman script
langGreek	14	Greek script using smRoman script code
langIcelandic	15	modified smRoman/Icelandic script
langMaltese	16	Roman script
langTurkish	17	modified smRoman/Turkish script
langCroatian	18	modified smRoman/Croatian script
langTradChinese	19	Chinese (Mandarin) in traditional characters
langUrdu	20	Arabic script
langHindi	21	Devanagari script
langThai	22	Thai script
langKorean	23	Korean script

Nan	Meaning
1	Invalid square root (negative number, usually)
2	Invalid addition (indeterminate such as infinity + (-infinity))
4	Invalid division (indeterminate such as 0/0)
8	Invalid multiplication (indeterminate such as 0*infinity)
9	Invalid modulo such as (a mod 0)
17	Try to convert invalid string to a number like val(”x7”)
33	Invalid argument in a trig function
34	Invalid argument in an inverse trig function
36	Invalid argument in a log function
37	Invalid argument in Pow function
38	Invalid argument in toolbox financial function
40	Invalid argument in hyperbolic function
42	Invalid argument in a gamma function

Symbol	Description and result
0	Digit placeholder. For example, if the value 8.9 is to be displayed as 8.90, use the format #.00
#	Digit placeholder. This symbol follows the same rules as the 0 symbol. However, the application shall not display extra zeros when the number typed has fewer digits on either side of the decimal than there are # symbols in the format. For example, if the custom format is #.##, and 8.9 is in the cell, the number 8.9 is displayed.
?	Digit placeholder. This symbol follows the same rules as the 0 symbol. However, the application shall put a space for insignificant zeros on either side of the decimal point so that decimal points are aligned in the column. For example, the custom format 0.0? aligns the decimal points for the numbers 8.9 and 88.99 in a column.
. (period)	Decimal point.
%	Percentage. If the cell contains a number between 0 and 1, and the custom format 0% is used, the application shall multiply the number by 100 and adds the percentage symbol in the cell.
, (comma)	Thousands separator. The application shall separate thousands by commas if the format contains a comma that is enclosed by number signs (#) or by zeros. A comma that follows a placeholder scales the number by one thousand. For example, if the format is #.0,, and the cell value is 12,200,000 then the number 12.2 is displayed.
E- E+ e- e+	Scientific format. The application shall display a number to the right of the ”E” symbol that corresponds to the number of places that the decimal point was moved. For example, if the format is 0.00E+00, and the value 12,200,000 is in the cell, the number 1.22E+07 is displayed. If the number format is #0.0E+0, then the number 12.2E+6 is displayed.
$ -+/():space	Displays the symbol. If it is desired to display a character that differs from one of these symbols, precede the character with a backslash (\). Alternatively, enclose the character in quotation marks. For example, if the number format is (000), and the value 12 is in the cell, the number (012) is displayed.
\	Display the next character in the format. The application shall not display the backslash. For example, if the number format is 0\!, and the value 3 is in the cell, the value 3! is displayed.
*	Repeat the next character in the format enough times to fill the column to its current width. There shall not be more than one asterisk in one section of the format. If more than one asterisk appears in one section of the format, all but the last asterisk shall be ignored. For example, if the number format is 0*x, and the value 3 is in the cell, the value 3xxxxxx is displayed. The number of x characters that are displayed in the cell varies based on the width of the column.
_ (underline)	Skip the width of the next character. This is useful for lining up negative and positive values in different cells of the same column. For example, the number format _(0.0_);(0.0) aligns the numbers 2.3 and -4.5 in the column even though the negative number is enclosed by parentheses.
”text”	Display whatever text is inside the quotation marks. For example, the format 0.00 ”dollars” displays 1.23 dollars when the value 1.23 is in the cell.
@	Text placeholder. If text is typed in the cell, the text from the cell is placed in the format where the at symbol (@) appears. For example, if the number format is ”Bob ”@” Smith” (including quotation marks), and the value ”John” is in the cell, the value Bob John Smith is displayed.

[Black] [Green] [White] [Blue] [Magenta] [Yellow] [Cyan] [Red]

To display	As	Use this code
Months	1-12	m
Months	01-12	mm
Months	Jan-Dec	mmm
Months	January-December	mmmm
Months	J-D	mmmmm
Days	1-31	d
Days	01-31	dd
Days	Sun-Sat	ddd
Days	Sunday-Saturday	dddd
Years	00-99	yy
Years	1900-9999	yyyy
Hours	0-23	h
Hours	00-23	hh
Minutes	0-59	m
Minutes	00-59	mm
Seconds	0-59	s
Seconds	00-59	ss
Time	4 AM	h AM/PM
Time	4:36 PM	h:mm AM/PM
Time	4:36:03 P	h:mm:ss A/P
Time	4:36:03.75	h:mm:ss.00
Elapsed time	1:02	[h] :mm
Elapsed time	62:16	[mm] :ss
Elapsed time	3735.80	[ss] .00

To display	As	Use this code
1234.59	1234.6	####.#
8.9	8.900	#.000
.631	0.6	0.#
12	12.0	#.0#
1234.568	1234.57	#.0#
44.398	44.398	???.???
102.65	102.65	???.???
2.8	2.8	???.???
5.25	5 1/4	# ??/??
5.3	5 3/10	# ??/??
12000	12,000	#,###
12000	12	#,
12400000	12.4	0.0,,